Assignment – 1: Network Diagnostic Commands

Name – Avik Samanta

Roll No. - 204101016

Submission Date – 24/01/2021

Q:1

a. We can specify number of ECHO_REQUESTES using –**c** option with *ping* command.

Ex. : **$ ping -c 5 www.google.com**

This will send **5** ECHO_REQUEST packets to www.google.com.

b. We can specify the time interval using **-i** option with *ping* command.

Ex. : **$ ping -i 2 www.google.com**

This will send ECHO_REQUESTs to www.google.com in every 2 seconds.

c. We can send ECHO_REQUEST packets one after another without waiting for reply using the option **-l** with *ping* command.

Ex. : **$ ping -l 2 www.google.com**

This will preload 2 ECHO_REQUEST packets without waiting for reply. But for a normal user (not a super user), maximum limit of preload is **3**.

d. We can specify the packet size using **-s** option with *ping* command.

Ex. : **$ ping -s 32 www.google.com**

This will send ECHO_REQUEST packets of size 32 bytes. If the Packet Size is set to 32 bytes, the total packet size will be 40 bytes [including 8 bytes of ICMP header] [and there is also 20 bytes of IPv4 header - resulting in total of 60 bytes].

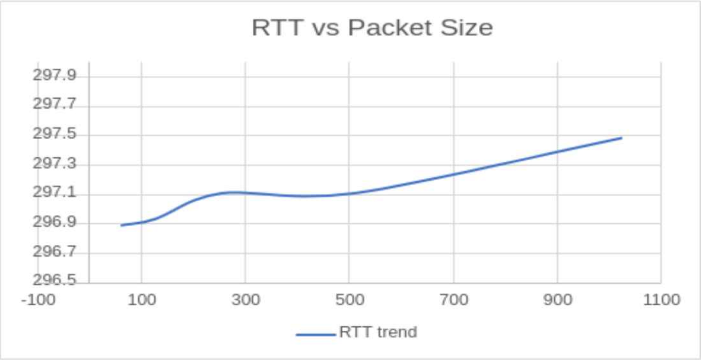
Q:2

My public IP - 202.142.99.128, located at Kolkata, India

The hosts that are chosen for the experiment :-

IP address(Domain Name)	**Location**	**6:30 IST**	**15:30 IST**	**23:30 IST**
163.53.78.110(flipkart.com)	Bangalore, India	213.705ms	212.812ms	212.376ms
23.58.50.171(myntra.com)	Kolkata, India	240.994ms	241.803ms	241.814ms
52.85.128.12(amazon.com)	Washington, USA	232.807ms	234.368ms	234.665ms
142.250.182.196(google.com)	California, USA	293.160ms	296.754ms	293.880ms
81.17.18.194(marksandspensor.com)	Zurich, Switzerland	97.842ms	102.921ms	97.679ms
203.205.219.58(qq.com)	Shenzhen, Hong Kong	246.122ms	246.165ms	246.134ms

- The first three columns give us the details of the host we have chosen.
- The 4th column indicates the ping Avg. RTT (at 6:30 IST) for each host. And while Experimenting we found out that, 0% packets were lost for each of the hosts.
- The 5th column indicates the ping Avg. RTT (at 15:30 IST) for each host. And while Experimenting we found out that, 0% packets were lost for all of the hosts, except for 81.17.18.194 (marksandspensor.com), where almost 50% packets were lost.
- The 6th column indicates the ping Avg. RTT (at 23:30 IST) for each host. And while Experimenting we found out that, 0% packets were lost for each of the hosts.
- We can see the geographical distance barely effecting the avg. RTT time. Even sometimes, hosts which are far away, having less RTT, that for the hosts with less geographical distance. So, geographical distance is weakly correlated with the avg. RTTs.
- In my case, I did not come across any packet loss during ping ECHO_REQUESTs at the night time or at the morning. But I got some loss of packets during working hours of the day [15:00 IST], for one of the hosts.
- Normally that can be because of two reasons, Firewall may have blocked the packets for some reason, or the server may be experiencing heavy traffic or may even be down.



– The data I got (pinging the host 142.250.182.196), using different packet sizes - [64 bytes - 296.887 ms, 128 bytes - 296.930 ms, 256 bytes - 297.106 ms, 512 bytes - 297.106 ms, 1024 bytes - 297.481 ms, 2048 bytes – 100% packets lost]. I plotted that, which clearly shows, avg. RTT increases with the packet size (almost linearly).

Q:3

a. **ifconfig command output explanation :-**

1. Interfaces :-
   - Enp2s0 (Ethernet interface)
   - Lo (loopback interface)
   - Wlo1 (Wireless network interface)
2. Flags :-
   - UP – Device is functioning
   - RUNNING – Interface is ready to accept data
   - BROADCAST – Device can send traffic to all the hosts in the link
   - MULTICAST – Device support multicasting
3. MTU - Maximum Transmission Unit, Maximum allowed frame size. Devices by default have MTU set as 1500. But loopback device normally has larger MTU than others (in this case 64k).
4. IP :-
   - inet – machine’s IP (IPv4) address
   - netmask – Network mask
   - broadcast – Broadcast address of the network
   - inet6 – machine’s IP (IPv6) address
   - prefixlen – Length of the network portion of the address (prefix)
5. Ethernet :-
   - ether – This denotes the hardware or MAC address, different for each Ethernet card.
   - txqueuelen – Length of the transmission queue
6. Packets :-
   - RX packets – no of packets received (and total size)
   - RX errors – it shows how many packets are dropped or overrun. Those having 0 values, means no any packet is dropped or overrun. If those have values greater than zero, either the ethernet device is failing or there is congestion in the network.
   - TX packets – no of packets transmitted (and total size)
   - TX errors – it is same as RX error, but for transmitted packets.

b. **ifconfig options :-**

- **$ ifconfig –a**

This shows all the interfaces, even if they are down. As shown above.

- **$ ifconfig –s**

What it does is that, instead of showing details of the device, it shows a short list, brief description of all the interfaces.

- Up - **$ ifconfig <interface> up**

This activates the driver for the given <interface>

- Down - **$ ifconfig <interface> down**

This deactivates the driver for the given <interface>

- **$ ifconfig <interface> <address>**

This assigns the given IP address to the interface

- **$ ifconfig <interface> mtu <bytes>**

This assigns the MTU value for the Network interface

- Add - **$ ifconfig <interface> add <address/prefixlength>**

This is used to add the given IPv6 address to the <interface>

- Delete - **$ ifconfig <interface> del <address/prefixlength>**

This is used to remove an IPv6 address from the <interface>

- ARP - **$ ifconfig <interface> [-]arp**

This is used to enable/disable ARP protocol in the <interface>

- Promiscuous- **$ ifconfig <interface> [-]promisc**

This is used to enable/disable ARP promiscuous mode (all the packets will be received by the network) in the <interface>

- All Multicast - **$ ifconfig <interface> [-]allmulti**

This is used to enable/disable all multicast mode (all the multicast packets will be received) in the <interface>

c. route command output Explanation :-

1. This command is used to work with kernel routing table.
2. Destination – the destination network or the destination host. Default is set to 0.0.0.0 (non-routable meta-address which is used to denote invalid, unknown or non-applicable target). The link-local is a special address, used for communication between two hostes, which are on the same network segment or broadcasting domain.
3. Gateway – the gateway address of the network.
4. Genmask – the network mask of the destination network. The mask is 0.0.0.0 for default route.
5. Flags – there can be different kinds of flags, i.e. -
   - U – the router is up
   - H – target is host
   - G – use gateway
   - ! - reject route
6. Metric – the distance to the target (counted as number Of hops)
7. Ref – number of references to the route
8. Use – Count of look-ups for the route
9. Iface – interface for which packets for the route will be sent

d. Route command options :-

★ **-n**

Routing table in numeric form. Output looks like -

$ route -n

```
Kernel IP routing table

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

0.0.0.0         192.168.0.1     0.0.0.0         UG    600    0        0 wlo1

169.254.0.0     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
```

★ **-Cn**

Operates on the kernel’s routing cache. To route the packets faster, Kernel maintains this routing cache information. The above command will print the cache information. The output looks like -

```
$ route –Cn

Kernel IP routing cache

Source          Destination     Gateway         Flags Metric Ref    Use Iface
```

★ **del**

This removes any given network (-net), or given host (-host), or the default gateway (default gw) route from the network. Output example (result) -

```
$ sudo route del default

$ route -n

Kernel IP routing table

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

169.254.0.0     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
```

★ **add**

This adds any network (-net), or any host (-host), or any default gateway (default gw) route to the network. Output example (result) -

```
$ sudo route add default gw 192.168.0.1

$ route -n

Kernel IP routing table

Destination     Gateway         Genmask         Flags Metric Ref    Use Iface

0.0.0.0         192.168.0.1     0.0.0.0         UG    0      0        0 wlo1

169.254.0.0     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
```

★ **reject**

This is used to block or reject routing to a particular host. In the following example we will try to reject routing to www.esquire.com (199.232.252.155). The output looks like -

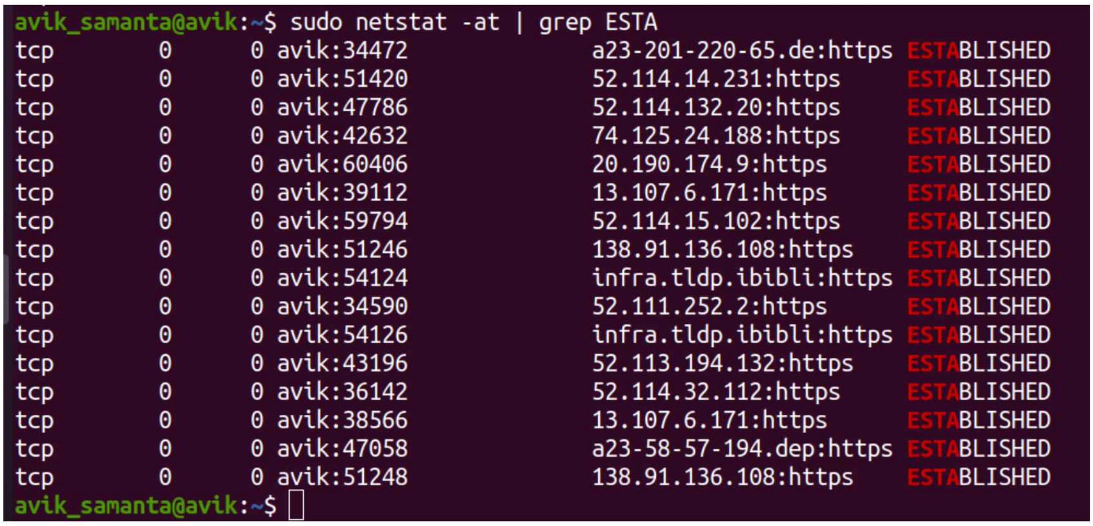
```
$ sudo route add -host 199.232.252.155 reject

$ ping -c 5 www.esquire.com

ping: connect: No route to host
```

Q:4

a. netstat command is a networking tool used for network statistics, troubleshooting, configuration, network connections, routing tables, masquerade connections, multicast memberships etc.

b. To show all the ESTABLISHED TCP connections, use **–at** option -

```
$ sudo netstat –at | grep ESTA
```



```
avik_samanta@avik:~$ sudo netstat -at | grep ESTA
tcp        0      0 avik:34472              a23-201-220-65.de:https ESTABLISHED
tcp        0      0 avik:51420              52.114.14.231:https     ESTABLISHED
tcp        0      0 avik:47786              52.114.132.20:https     ESTABLISHED
tcp        0      0 avik:42632              74.125.24.188:https     ESTABLISHED
tcp        0      0 avik:60406              20.190.174.9:https      ESTABLISHED
tcp        0      0 avik:39112              13.107.6.171:https      ESTABLISHED
tcp        0      0 avik:59794              52.114.15.102:https     ESTABLISHED
tcp        0      0 avik:51246              138.91.136.108:https    ESTABLISHED
tcp        0      0 avik:54124              infra.tldp.ibibli:https ESTABLISHED
tcp        0      0 avik:34590              52.111.252.2:https      ESTABLISHED
tcp        0      0 avik:54126              infra.tldp.ibibli:https ESTABLISHED
tcp        0      0 avik:43196              52.113.194.132:https    ESTABLISHED
tcp        0      0 avik:36142              52.114.32.112:https     ESTABLISHED
tcp        0      0 avik:38566              13.107.6.171:https      ESTABLISHED
tcp        0      0 avik:47058              a23-58-57-194.dep:https ESTABLISHED
tcp        0      0 avik:51248              138.91.136.108:https    ESTABLISHED
avik_samanta@avik:~$ 
```

- First column represents the internet protocol
- 2nd and 3rd column indicate the Recv-Q and Send-Q columns tell us how much data is in the queue for that socket, waiting to be read (Recv-Q) or sent (Send-Q).
- 4th column denotes the Local IP address (my computer) and the port no that is being used. (the local IP address is provided by the DHCP)
- The 5th column denotes the foreign address. It is actually the IP address and port no of a remote device to which the socket is connected.
- The last column indicates the state of the connection (ESTABLISHED/LISTENING/TIME-WAIT/ACKNOLEDGEMENT/SYN-SENT)

c. $ netstat -r

- This gives us the kernel routing information.
- Destination – the destination network or the destination host. Default is set to 0.0.0.0 (non-routable meta-address which is used to denote invalid, unknown or non-applicable target). The link-local is a special address, used for communication between two hostes, which are on the same network segment or broadcasting domain.
- Gateway – the gateway address of the network.
- Genmask – the network mask of the destination network. The mask is 0.0.0.0 for default route.
- Flags – there can be different kinds of flags, i.e. -
    - U – the router is up
    - H – target is host
    - G – use gateway
    - ! - reject route
- Metric – the distance to the target (counted as number Of hops)
- Ref – number of references to the route
- Use – Count of look-ups for the route
- mss – default maximum segment size for TCP over the route
- window - Default window size for TCP connections over this route
- irtt - initial round trip time (RTT), The kernel uses this to guess about the best TCP protocol parameters without waiting on answers.
- Iface – interface for which packets for the route will be sent

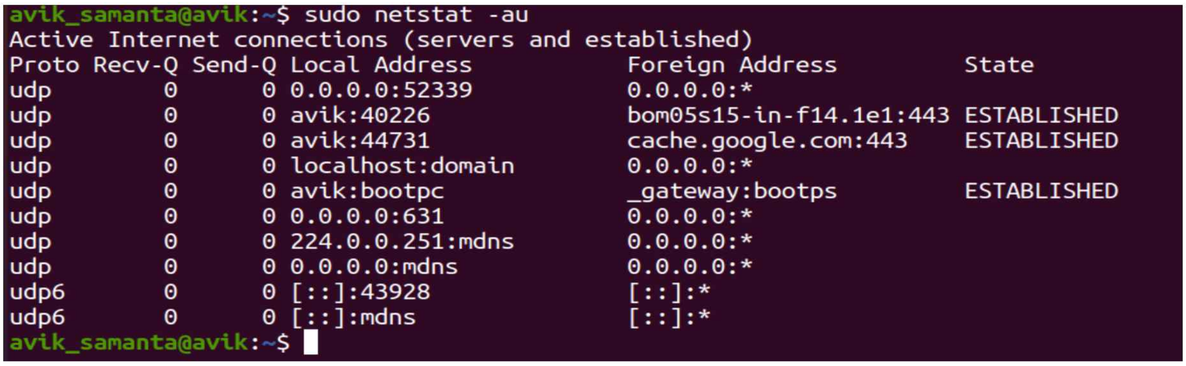
d. We get the status of all the interfaces, using **-i** option with netstat command

$ netstat -i

```
Kernel Interface table

Iface      MTU    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg

enp2s0    1500        0      0      0 0             0      0      0      0 BMU

lo       65536     4214      0      0 0          4214      0      0      0 LRU

wlo1      1500   895489      0      0 0        336678      0      0      0 BMRU
```

So, there are three network interfaces in my computer (one Ethernet interface, one loopback interface, and one wireless network interface)

e. To show the statistics of all UDP connections, use **-au** option -



```
avik_samanta@avik:~$ sudo netstat -au
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 0.0.0.0:52339           0.0.0.0:*
udp        0      0 avik:40226              bom05s15-in-f14.1e1:443 ESTABLISHED
udp        0      0 avik:44731              cache.google.com:443    ESTABLISHED
udp        0      0 localhost:domain        0.0.0.0:*
udp        0      0 avik:bootpc             _gateway:bootps         ESTABLISHED
udp        0      0 0.0.0.0:631             0.0.0.0:*
udp        0      0 224.0.0.251:mdns        0.0.0.0:*
udp        0      0 0.0.0.0:mdns            0.0.0.0:*
udp6       0      0 [::]:43928              [::]:*
udp6       0      0 [::]:mdns               [::]:*
avik_samanta@avik:~$
```

f. The loopback device is a special, virtual network interface that your computer uses to communicate with itself. It is used mainly for diagnostics and troubleshooting, and to connect to servers running on the local machine.

Q:5

Traceroute command is used to find out what path a packet takes to reach the destination host.

a. I used the traceroute command with ICMP (**$ traceroute <hostname> -I**) packets

IP Address	hopCount (15:30 IST)	hopCount (22:30 IST)	hopCount (7:30 IST)
163.53.78.110	14	14	14
23.58.50.171	11	11	11
52.85.128.12	18	18	18
142.250.182.196	11	11	11
81.17.18.194	15	15	17
203.205.219.58	17	17	17

- 192.168.0.1, 172.21.227.1, 10.10.245.25, * * *, * * *, 103.225.178.121, 172.31.1.66 - These hops were common at the beginning of the trace routing for each of the hosts.
- 115.113.172.1, 172.25.75.226 were common hops after those 7 hops for 2nd and 3rd hosts in the list.
- 182.73.243.165 was another common hop for the 5th and 6th hosts after in the list the 1st 7 common hops.

b. Same host may have different route at different time of the day, because of the traffic experienced at each hop and routing and congestion differences.

c. There can be hosts for which traceroute does not find complete route. This may happen due to several reasons. There may be no route from source to destination. It may happen like hop or destination firewall or ACL is configured not to accept or forward ICMP packet requests. It also may happen the server is down or experiencing heavy traffic.

d. As ping sends ICMP echo-requests and expect ICMP echo-reply, and if some remote host is configured not to accept or forward ICMP packets (either by default or to handle heavy traffic), then the ping request will be blocked, but still the traceroute can find the host using standard network routing TCP/UDP.

Q:6

a. To show the ARP table of the device, we use **arp** command

$ arp

```
Address                  HWtype  HWaddress           Flags Mask            Iface
```

```
_gateway                 ether   d8:07:b6:99:3f:80   C                      wlo1
```

- Address – denotes the IP address of the hosts
- HWtype – denotes the type of the device (interface)
- HWaddress – indicates the hardware address or MAC address
- Flags - [C : the entries are dynamically learned by the protocol, M : the entries are manually created, P : Publish, it tells the hosts to respond to the packets which are ARP request and ARP response]
- Mask – network mask (normally no entries in these cases)
- Iface – what network interface (ethernet or wireless LAN) the host is using

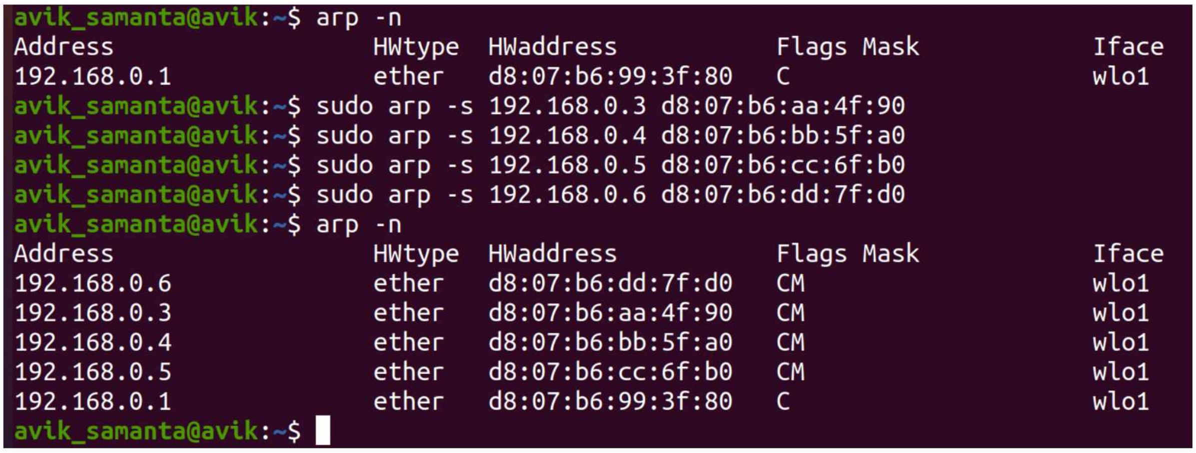
b.

- Adding new entries to the ARP table, using **-s** option (static hosts)

  $ sudo -s <hostname> <hw_address>

- Deleting/removing entries from the ARP table, using **-d** option

  $ sudo -s <hostname>



```
avik_samanta@avik:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.1              ether   d8:07:b6:99:3f:80   C                     wlo1
avik_samanta@avik:~$ sudo arp -s 192.168.0.3 d8:07:b6:aa:4f:90
avik_samanta@avik:~$ sudo arp -s 192.168.0.4 d8:07:b6:bb:5f:a0
avik_samanta@avik:~$ sudo arp -s 192.168.0.5 d8:07:b6:cc:6f:b0
avik_samanta@avik:~$ sudo arp -s 192.168.0.6 d8:07:b6:dd:7f:d0
avik_samanta@avik:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.168.0.6              ether   d8:07:b6:dd:7f:d0   CM                    wlo1
192.168.0.3              ether   d8:07:b6:aa:4f:90   CM                    wlo1
192.168.0.4              ether   d8:07:b6:bb:5f:a0   CM                    wlo1
192.168.0.5              ether   d8:07:b6:cc:6f:b0   CM                    wlo1
192.168.0.1              ether   d8:07:b6:99:3f:80   C                     wlo1
avik_samanta@avik:~$
```

c. There are couple of parameters which determine how long the kernel ARP cache entries will last and when they are deleted from the cache -

- Type of entry – if static, means they are configured manually, they stay there permanently. Whereas dynamic, means configured automatically by ARP protocols, will last for a specific amount of time.
- Timeout – in every system there will be a specific timeout (300 seconds for my system) for the dynamic entries (in kernel ARP cache). If they are not used for that amount of time, then those entries will be removed from the cache.

Trial and Error method to discover the ARP cache entries -

- First take some timeout value, for example 60 mins. Then make the system clock 60 mins faster.
- If the ARP cache is cleared, then check some smaller value for the timeout, i.e. 30 mins or so.
- But if the ARP cache is not cleared, check some larger values for the timeout, i.e. 120 mins or so.

d. If two IP addresses map to same Ethernet address, then there can be two cases -

- If they are in different LANs, then nothing bad will happen, as Ethernet or Physical or MAC address never leaves the network, which the NIC is immediately connected to.
- But if they are connected to the same LAN, then they will confuse switches, they will try to respond to the same traffic. One will get the requests of the other one and vice versa, as processing requests then will completely depend on the traffic and who got the last request.