IP A

Check the IP address allocated to your server instance/computer

```
# ip a
```

1. PING

Check the connection with the destination server if ICMP is enabled. This is very basic but most important tool used as network troubleshooting tools.

```
[root@localhost ~]# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=48 time=43.4 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=48 time=48.5 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=48 time=44.2 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=48 time=63.3 ms
^C
--- 8.8.8.8 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3166ms
rtt min/avg/max/mdev = 43.478/49.902/63.338/7.996 ms
```

2. DIG

Check DNS server settings

```
 [sawda@localhost ~]$ dig


; <<>> DiG 9.11.28-RedHat-9.11.28-1.fc32 <<>>

;; global options: +cmd

;; Got answer:

;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5735

;; flags: qr rd ra ad; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 1


;; OPT PSEUDOSECTION:

; EDNS: version: 0, flags:; udp: 1432

;; QUESTION SECTION:

;.                               IN      NS


;; ANSWER SECTION:

.                       86400   IN      NS      h.root-servers.net.

.                       86400   IN      NS      b.root-servers.net.
.                    86400  IN     NS     f.root-servers.net.
.                    86400  IN     NS     c.root-servers.net.
.                    86400  IN     NS     e.root-servers.net.
.                    86400  IN     NS     a.root-servers.net.
.                    86400  IN     NS     d.root-servers.net.
.                    86400  IN     NS     g.root-servers.net.
.                    86400  IN     NS     l.root-servers.net.
.                    86400  IN     NS     k.root-servers.net.
.                    86400  IN     NS     m.root-servers.net.
.                    86400  IN     NS     j.root-servers.net.
.                    86400  IN     NS     i.root-servers.net.

;; Query time: 6 msec
;; SERVER: 192.168.16.252#53(192.168.16.252)
;; WHEN: Tue Feb 27 18:04:48 EAT 2024
;; MSG SIZE  rcvd: 239
```

2. NETSTAT

Check all the TCP and UDP connections established with the Server. This tool is used to gather a wide range of information of network topology like number of connections, Listening connections, local and remote IP addresses and ports.

```
[root@localhost ~]# netstat -an
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address   Foreign   Address State
tcp    0      0   127.0.0.1:9463  0.0.0.0:*    LISTEN
tcp    0      0   127.0.0.1:15672 0.0.0.0:*    LISTEN
tcp    0      0   127.0.0.1:5432  0.0.0.0:*    LISTEN
tcp    0      0   127.0.0.1:25    0.0.0.0:*    LISTEN
tcp    0      0   0.0.0.0:443     0.0.0.0:*    LISTEN
tcp    0      0   127.0.0.1:16379 0.0.0.0:*    LISTEN
```

3. NETCAT

It first surfaced in 1995.It is one of the most popular and very lightweight network troubleshooting tools to date. It lets two computer transfer data with each other using TCP and UDP protocols via Network Layer Protocol IP. The name may have derived from common command cat we use in Linux. You can initiate a connection and check the port reachability of remote server using nc command as shown below.

```
[root@localhost ~]# nc -vz 8.8.8.8 443
Ncat: Version 7.50 ( https://nmap.org/ncat )
Ncat: Connected to 8.8.8.8:443.
Ncat: 0 bytes sent, 0 bytes received in 0.10 seconds.
```

4. TRACEPATH

This tool is very similar like traceroute tool. You will find this command already installed in Ubuntu Systems. If you want to check and verify the network path and network delay caused by any of the hops between source to destination, then tracepath tool is the best network troubleshooting tools to use in Ubuntu Machine as you can see below.

Advertisements

```
[root@localhost ~]# tracepath google.com
1?: [LOCALHOST] pmtu 1500
1: gateway 7.795ms
1: gateway 2.504ms
2: gateway 2.764ms pmtu 1480
2: 10.172.76.5 3.642ms
3: google.example.in 5.023ms
4: google.example.in 6.205ms
5: 10.198.100.21 4.652ms asymm 4
```

5. TELNET

This tool will not be available by default in your system. You need to download and install in your Linux machine to use it. You can use yum install telnet to install in RedHat/CentOS based System and sudo apt-get install telnet in Ubuntu based Systems.

```
[root@localhost ~]# telnet 8.8.8.8 443
Trying 8.8.8.8...
Connected to 8.8.8.8.
Escape character is '^]'.
```

6. CURL

Sometimes you might not get telnet or netcat tool in a security enhanced server, there curl tool can become very handy to check the port reachability for any remote Service. Usually, you will find this tool installed in your Linux system by default, so you do not have to take overhead of installing it separately.

Advertisements

```
[root@localhost ~]# curl -v telnet://8.8.8.8:443
* About to connect() to 8.8.8.8 port 443 (#0)
* Trying 8.8.8.8...
* Connected to 8.8.8.8 (8.8.8.8) port 443 (#0)
```

7. ETHTOOL

If you are observing any slowness in network due to parameters mismatch between switch and local interface, then probably ethtool will be the best tool to use to check all the parameters configured on host end.

```
[root@localhost ~]# ethtool enp0s3
Settings for enp0s3:
Supported ports: [ TP ]
Supported link modes: 10baseT/Half 10baseT/Full
100baseT/Half 100baseT/Full
1000baseT/Full
Supported pause frame use: No
Supports auto-negotiation: Yes
Supported FEC modes: Not reported
Advertised link modes: 10baseT/Half 10baseT/Full
100baseT/Half 100baseT/Full
1000baseT/Full
Advertised pause frame use: No
Advertised auto-negotiation: Yes
Advertised FEC modes: Not reported
Speed: 1000Mb/s
Duplex: Full
Port: Twisted Pair
PHYAD: 0
Transceiver: internal
Auto-negotiation: on
MDI-X: off (auto)
Supports Wake-on: umbg
Wake-on: d
Current message level: 0x00000007 (7)
drv probe link
Link detected: yes
```

8. IP ADDR SH

You might be habitual of using ifconfig -a command in previous RedHat/Centos based system to check the network interface and IP associated with it. In recent released version, you might not able to use ifconfig command anymore. You need to use ip addr show command to check the interface and ip associated with it as shown below.

```
[root@localhost ~]# ip addr sh
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 pfifo_fast state
UP group default qlen 1000
link/ether 08:09:07:9j:g9:6a brd ff:ff:ff:ff:ff:ff
inet 192.168.0.110/24 brd 192.168.0.255 scope global noprefixroute enp0s3
valid_lft forever preferred_lft forever
inet6 fj90::4e8c:5735:890b:6lk7/64 scope link noprefixroute
valid_lft forever preferred_lft forever
```

9. ARP

It is abbreviated as Address Resolution Protocol. This is also a command which is used to identify the MAC address using associated IP Address of the System. This command will also answer any query for MAC address from the local cache using arp -a command as you can see below.

```
[root@localhost ~]# arp -a
znode2 (192.168.0.105) at <incomplete> on enp0s3
? (192.168.0.101) at 2g:16:98:1c:83:46 [ether] on enp0s3
gateway (192.168.0.1) at c4:6e:1f:49:44:7a [ether] on enp0s3
```

10. ROUTE

If you want to troubleshoot any routing error like "no route to host", then route is the best network troubleshooting tools to check your current available route in the system and verify this error. You can also add your route using route add command to create a route with the destination Server.

```
[root@localhost ~]# 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     100    0   0  enp0s3
172.17.0.0   0.0.0.0    255.255.0.0   U       0     0   0  docker0
192.168.0.0  0.0.0.0    255.255.255.0 U      100    0   0  enp0s3
```