IP, Net-mask and some manipulation

Create a Setup so that you can ping Google but not able to ping Facebook from same system.

When you here this, first thing that comes in mind is that firewall will do the work. But, today we shall see, how we can achieve this set-up by using routing table, IP, and net-mask.

First of all, we shall ping to google and facebook, and we see that we are successfully able to ping both of them.

ping www.google.com

ping www.facebook.com

```
[root@localhost ~]# ping  www.google.com
PING www.google.com (172.217.166.164) 56(84) bytes of data.
64 bytes from bom07s20-in-f4.1e100.net (172.217.166.164): icmp_seq=1 ttl=112 tim
e=39.2 ms
64 bytes from bom07s20-in-f4.1e100.net (172.217.166.164): icmp_seq=2 ttl=112 tim
e=37.5 ms
^C
--- www.google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 37.507/38.350/39.193/0.843 ms
[root@localhost ~]# ping  www.facebook.com
PING star-mini.c10r.facebook.com (157.240.198.35) 56(84) bytes of data.
64 bytes from edge-star-mini-shv-01-del1.facebook.com (157.240.198.35): icmp_seq
=1 ttl=54 time=46.0 ms
64 bytes from edge-star-mini-shv-01-del1.facebook.com (157.240.198.35): icmp_seq
=2 ttl=54 time=40.10 ms
64 bytes from edge-star-mini-shv-01-del1.facebook.com (157.240.198.35): icmp_seq
=3 ttl=54 time=39.7 ms
^C
--- star-mini.c10r.facebook.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 6ms
rtt min/avg/max/mdev = 39.734/42.243/46.000/2.706 ms
[root@localhost ~]#
```

Let's have a look at our routing table using the command **route –n**

```
[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
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
[root@localhost ~]# route del -net 0.0.0.0
[root@localhost ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
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
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
```

Whatever I have put under the red box, we shall talk about it in a moment. Before understanding what it means, we shall use route **del –net 0.0.0.0** and check the routing table again using route –n.

Now, we won't see the red color content in the routing table.

After doing this, try pinging google again by ping 172.217.166.164, you won't be able to establish a connection.

```
[root@localhost ~]# ping 172.217.166.164
connect: Network is unreachable
```

What is the problem exactly?

Let's have a look at the routing table to understand it in detail.

```
[root@localhost ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
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
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
```

From the first snapshot of the document, we can see that the IP of google is 172.217.166.164, however, in the routing table, we don't have that range. In simple words, this means that the IP of google is not in the range of our routing table; hence, we cannot establish a connection with google.

How can we get rid of this problem?

Well, now we are going to add the IP of google.com in our routing table. The command for that is: **route add –net 172.217.166.0 netmask 255.255.255.0 gw 192.168.0.1 enp0s3**.

This implies, that, now, we have added all the IPs from 172.217.166.0 to 172.217.166.255 to our routing table. Now, the IP of google, i.e., 172.217.166.164 is in the range.

```
[root@localhost ~]# route add -net 172.217.166.0  netmask 255.255.255.0
    gw 192.168.0.1 enp0s3
[root@localhost ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
172.17.0.0      0.0.0.0         255.255.0.0     U     0      0        0 docker0
172.217.166.0   192.168.0.1     255.255.255.0   UG    0      0        0 enp0s3
192.168.0.0     0.0.0.0         255.255.255.0   U     100    0        0 enp0s3
192.168.122.0   0.0.0.0         255.255.255.0   U     0      0        0 virbr0
```

Now, if we try pinging, we can see that the connection is established!

```
[root@localhost ~]# ping 172.217.166.164
PING 172.217.166.164 (172.217.166.164) 56(84) bytes of data.
64 bytes from 172.217.166.164: icmp_seq=1 ttl=112 time=42.7 ms
64 bytes from 172.217.166.164: icmp_seq=2 ttl=112 time=40.2 ms
64 bytes from 172.217.166.164: icmp_seq=3 ttl=112 time=32.8 ms
64 bytes from 172.217.166.164: icmp_seq=4 ttl=112 time=34.7 ms
```

Again, from the first snapshot, we can see the IP of facebook which is 157.240.198.35. If we try to ping it, there will be a network error. This error is neither due to bad internet connection nor facebook is down. It is because; the IP of facebook is not in the range of our routing table.

```
[root@localhost ~]# ping  157.240.198.35
connect: Network is unreachable
[root@localhost ~]#
```

To get rid of this issue, we can use the same steps to add Facebook's IP in the routing table. But, to be able to ping all the IPs, be it google, facebook, twitter, or our local system, we shall use the following command.

route add –net 0.0.0.0 netmask 0.0.0.0 gw 192.168.0.1 enp0s3

This is the same IP that we have deleted from our routing table to demonstrate this small use-case of IP.

It means that now, our routing table has all the IP ranging from 0.0.0.0 to 255.255.255.255 in it, and hence we can ping to any network.

```
[root@localhost ~]#
[root@localhost ~]# route add  -net 0.0.0.0  netmask 0.0.0.0   gw 192.168.0.1
      enp0s3
```