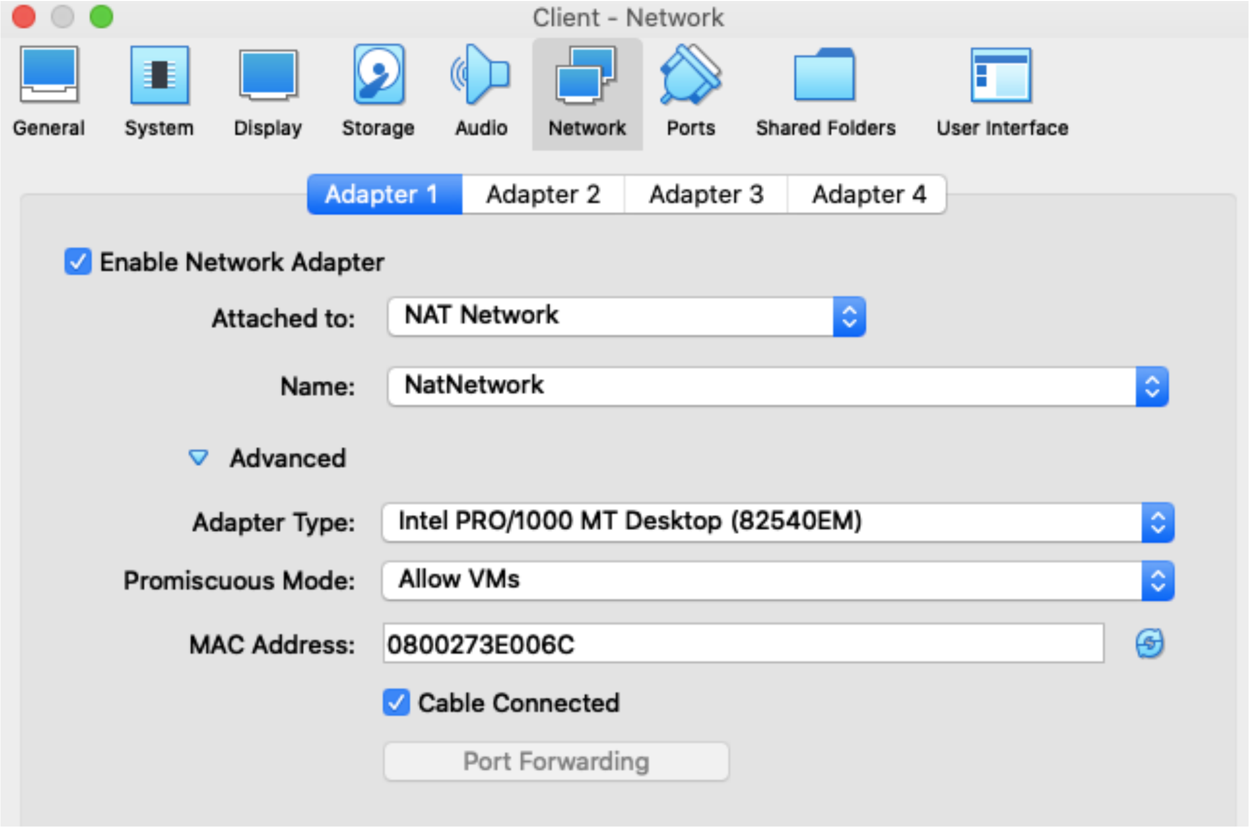
Roozah Khan

Final Project

Firewall Evasion lab

Task 1

We created client and server VMs and we enabled the network adapter called NAT network so the VMs can communicate with each other as well as reach out to the internet all in the same local network. We created a "Client" VM and "Server clone" VM.



☑ Enable Network Adapter

Attached to: NAT Network

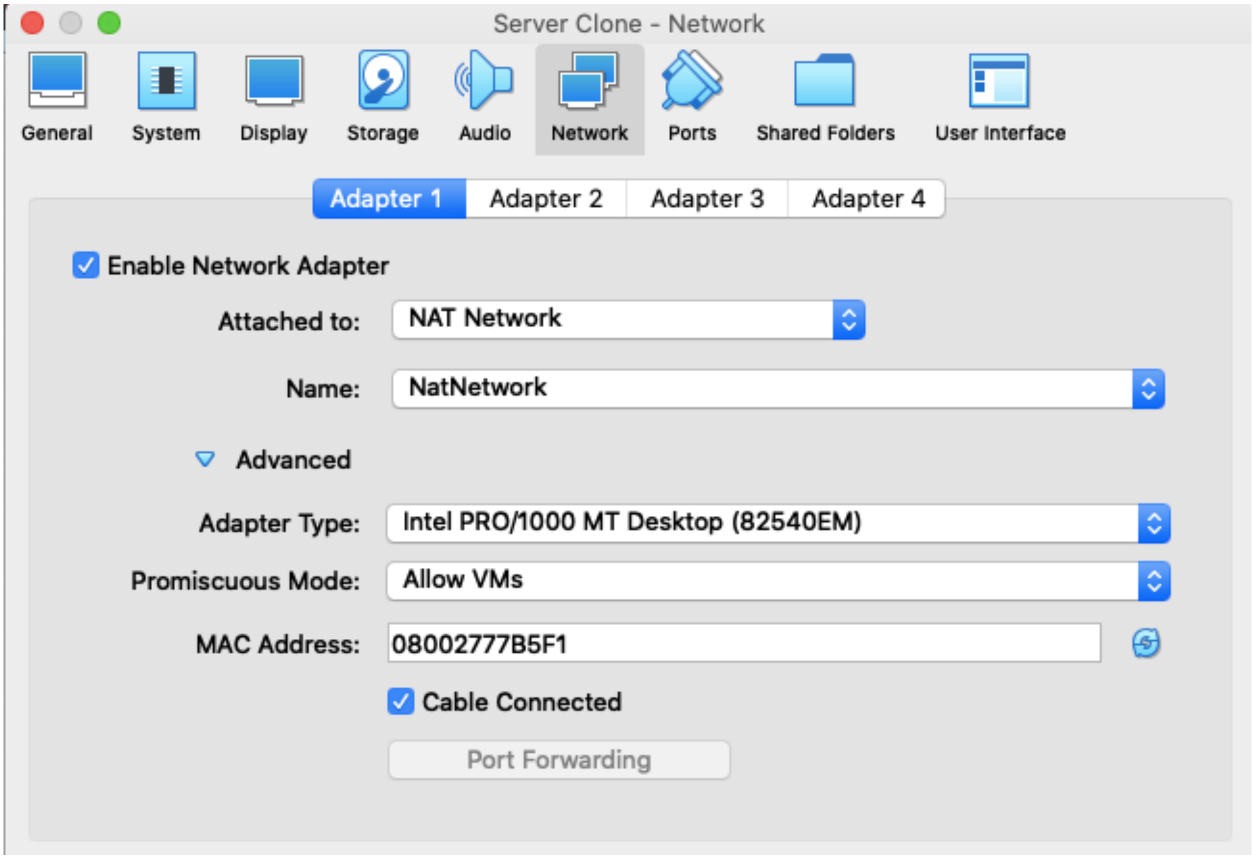
Name: NatNetwork

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Allow VMs

MAC Address: 0800273E006C

☑ Cable Connected



☑ Enable Network Adapter

Attached to: NAT Network

Name: NatNetwork

Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Allow VMs

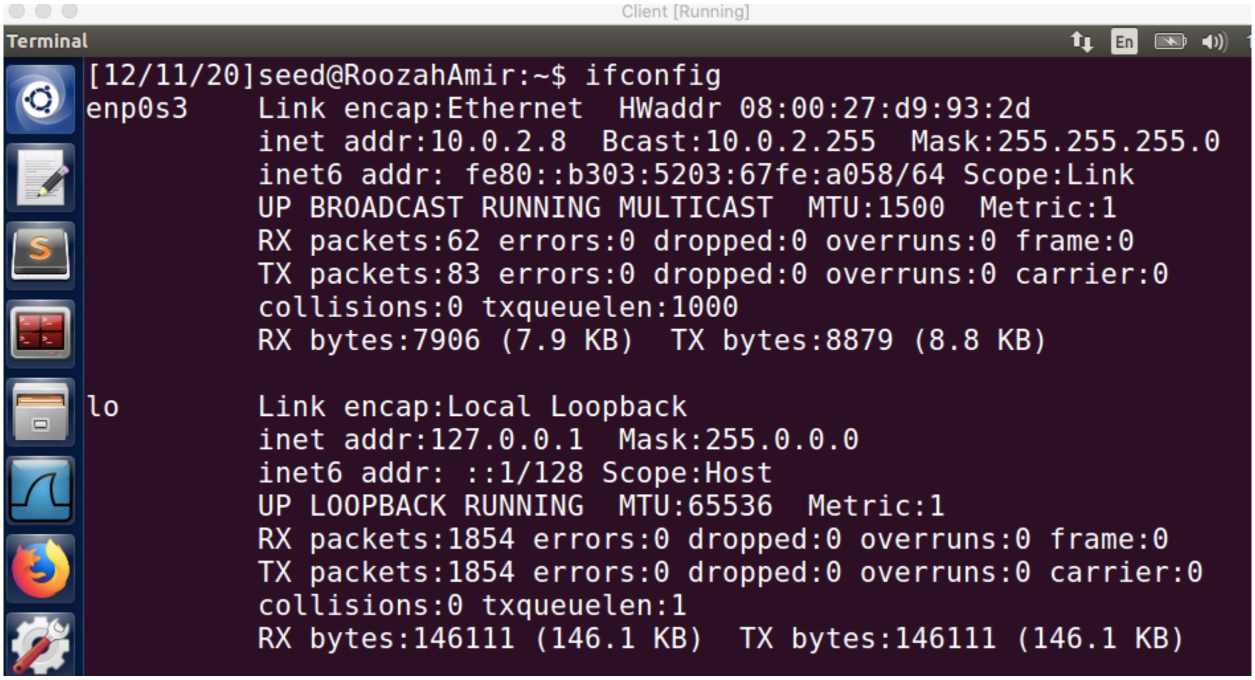
MAC Address: 08002777B5F1

☑ Cable Connected

To verify that the VMs are on the same local network we check their IP address.

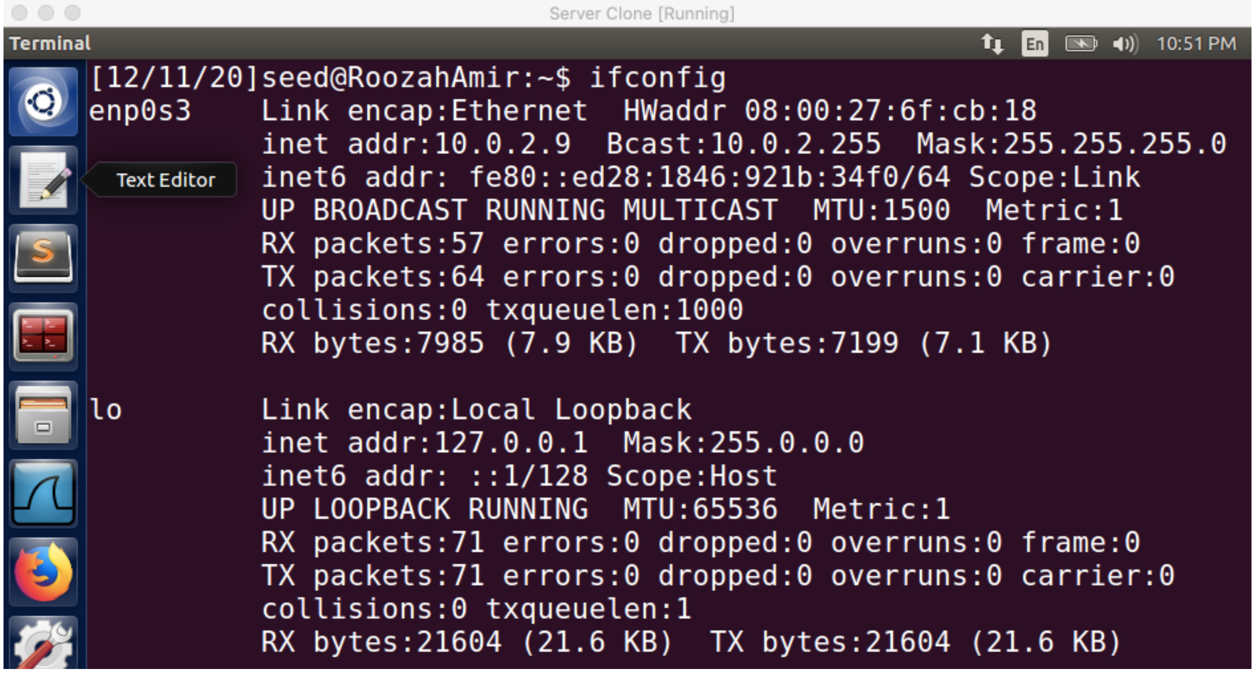
Server IP: 10.0.2.9

Client IP: 10.0.2.8



```
[12/11/20]seed@RoozahAmir:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:d9:93:2d
          inet addr:10.0.2.8  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::b303:5203:67fe:a058/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:62 errors:0 dropped:0 overruns:0 frame:0
          TX packets:83 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7906 (7.9 KB)  TX bytes:8879 (8.8 KB)

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:1854 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1854 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:146111 (146.1 KB)  TX bytes:146111 (146.1 KB)
```

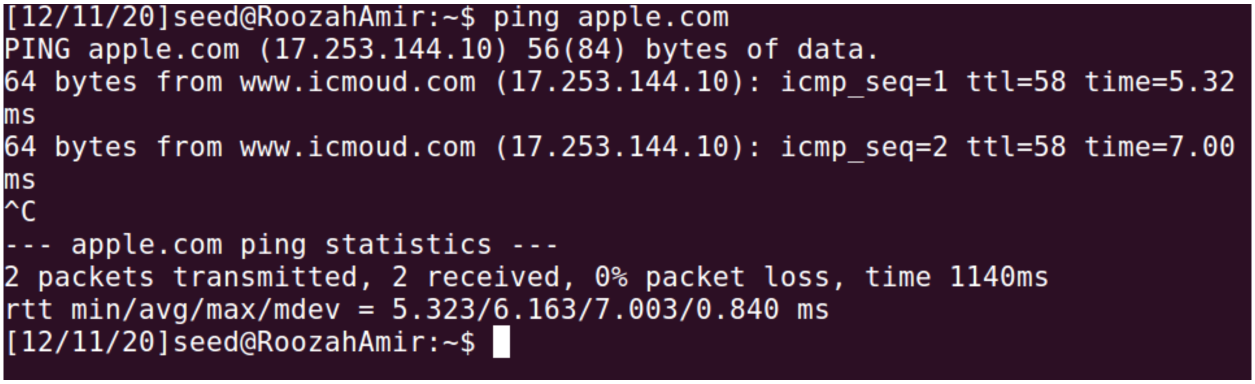


```
[12/11/20]seed@RoozahAmir:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:6f:cb:18
          inet addr:10.0.2.9  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::ed28:1846:921b:34f0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:57 errors:0 dropped:0 overruns:0 frame:0
          TX packets:64 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7985 (7.9 KB)  TX bytes:7199 (7.1 KB)

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:71 errors:0 dropped:0 overruns:0 frame:0
          TX packets:71 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:21604 (21.6 KB)  TX bytes:21604 (21.6 KB)
```

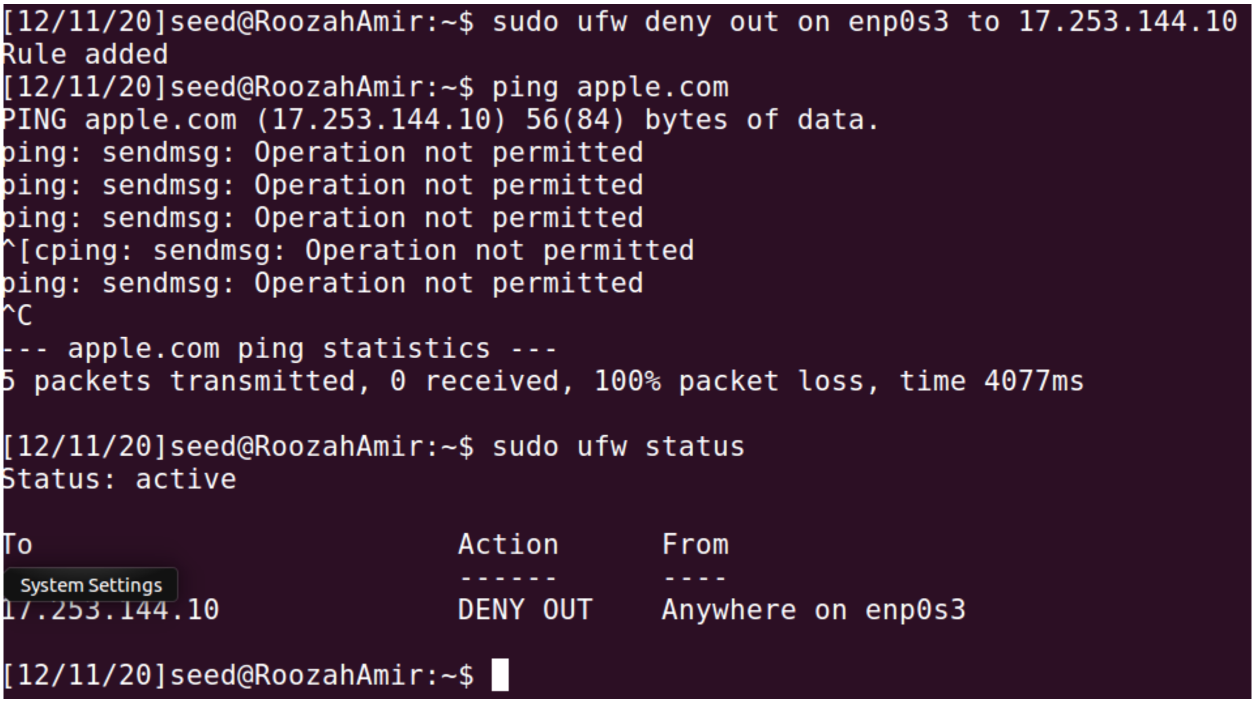
Task 2

In this task, we pick a target website and set up a firewall to check if we can block access to our target website. In our example, we chose Apple.com as our target website. We first use the “ping” command to get the IP address for Apple (17.253.144.10) to see if the firewall would block that specific IP address of Apple.



```
[12/11/20]seed@RoozahAmir:~$ ping apple.com
PING apple.com (17.253.144.10) 56(84) bytes of data.
64 bytes from www.icmoud.com (17.253.144.10): icmp_seq=1 ttl=58 time=5.32
ms
64 bytes from www.icmoud.com (17.253.144.10): icmp_seq=2 ttl=58 time=7.00
ms
^C
--- apple.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1140ms
rtt min/avg/max/mdev = 5.323/6.163/7.003/0.840 ms
[12/11/20]seed@RoozahAmir:~$ 
```

Next, we set up our firewall as shown in the screenshot below. We added the rule where the firewall denies Apple IP address and when we ping apple.com again, it says “operation not permitted” which means the firewall has blocked access to our target website. We also check the status and as you can see the firewall denies the IP address of Apple.



```
[12/11/20]seed@RoozahAmir:~$ sudo ufw deny out on enp0s3 to 17.253.144.10
Rule added
[12/11/20]seed@RoozahAmir:~$ ping apple.com
PING apple.com (17.253.144.10) 56(84) bytes of data.
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^[cping: sendmsg: Operation not permitted
ping: sendmsg: Operation not permitted
^C
--- apple.com ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4077ms

[12/11/20]seed@RoozahAmir:~$ sudo ufw status
Status: active

To                         Action      From
                           ------      ----
17.253.144.10              DENY OUT    Anywhere on enp0s3

[12/11/20]seed@RoozahAmir:~$ 
```

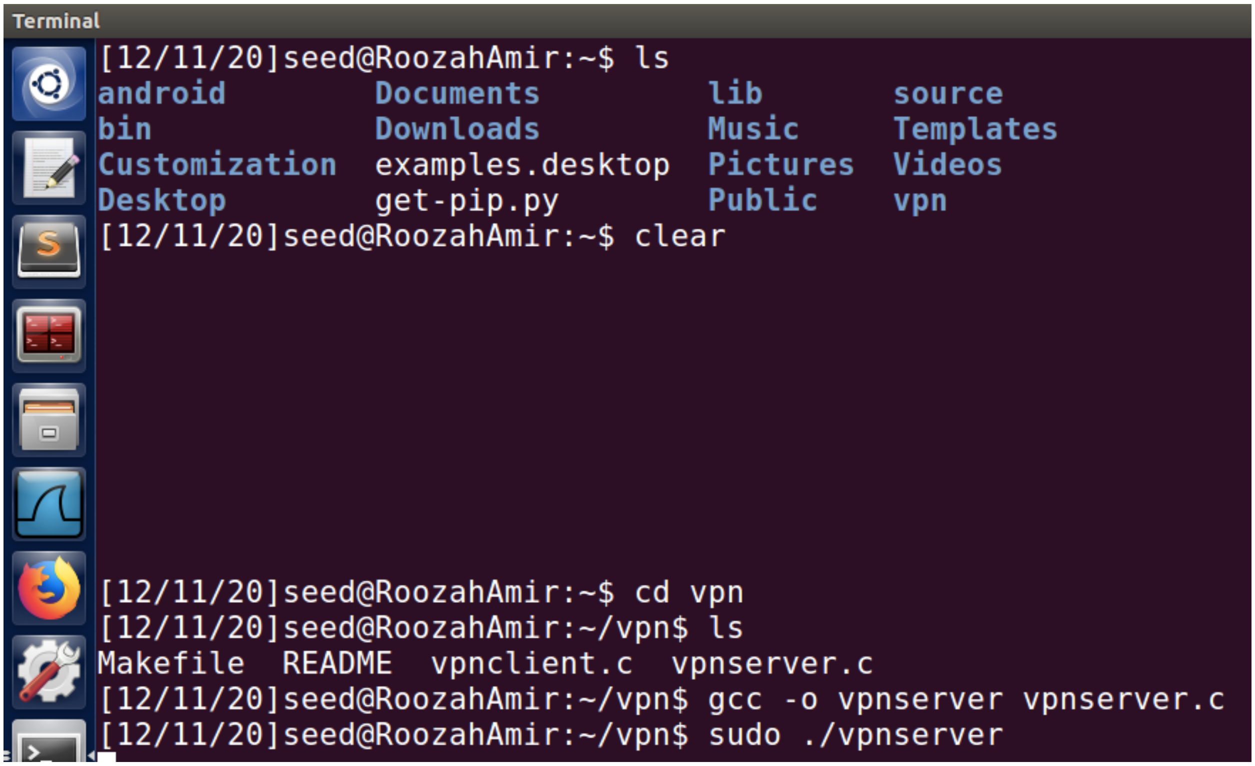
Task 3

In this task, we downloaded the VPN file to create a VPN tunnel between Client VM and Server VM. The VPN tunnel is used to bypass the firewall in order to access our target website which is Apple.com in our case.

Step 1: Server VM

We compile and run the vpnserver code to generate a TUN interface to establish the Server VM end of the tunnel for the IP tunneling.

Terminal 1

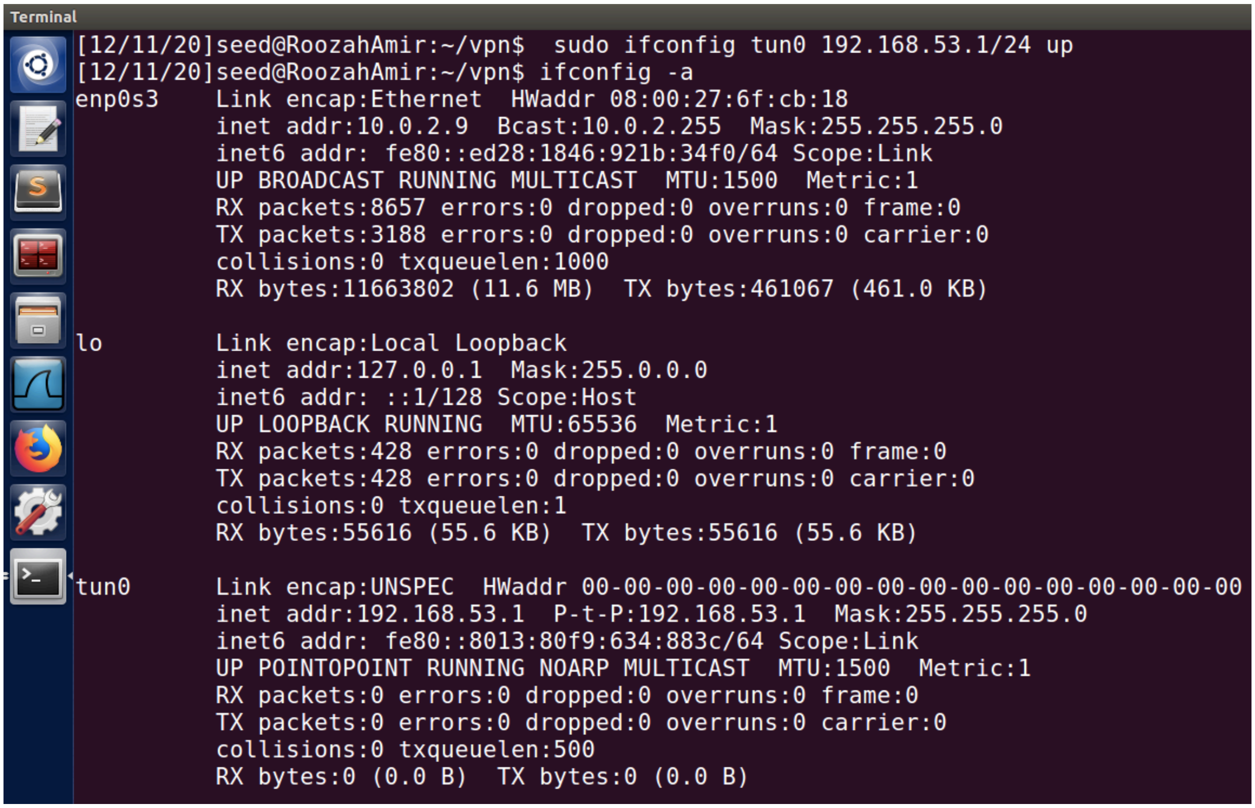


```
[12/11/20]seed@RoozahAmir:~$ ls
android        Documents         lib       source
bin            Downloads         Music     Templates
Customization  examples.desktop  Pictures  Videos
Desktop        get-pip.py        Public    vpn
[12/11/20]seed@RoozahAmir:~$ clear
```

```
[12/11/20]seed@RoozahAmir:~$ cd vpn
[12/11/20]seed@RoozahAmir:~/vpn$ ls
Makefile  README  vpnclient.c  vpnserver.c
[12/11/20]seed@RoozahAmir:~/vpn$ gcc -o vpnserver vpnserver.c
[12/11/20]seed@RoozahAmir:~/vpn$ sudo ./vpnserver
```

Terminal 2

In the 2nd terminal window, we need to configure and activate the TUN interface and give an IP address as shown down below. As you can see “tun0” which is the TUN interface has the new IP address that we assigned.

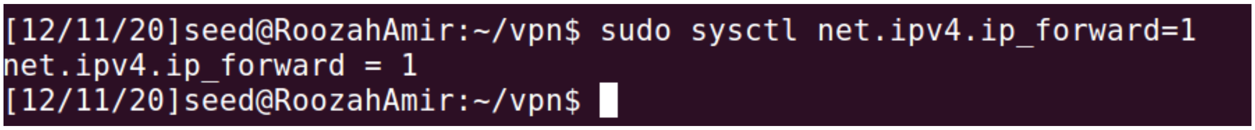


```
[12/11/20]seed@RoozahAmir:~/vpn$  sudo ifconfig tun0 192.168.53.1/24 up
[12/11/20]seed@RoozahAmir:~/vpn$ ifconfig -a
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:6f:cb:18
          inet addr:10.0.2.9  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::ed28:1846:921b:34f0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:8657 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3188 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11663802 (11.6 MB)  TX bytes:461067 (461.0 KB)

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:428 errors:0 dropped:0 overruns:0 frame:0
          TX packets:428 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:55616 (55.6 KB)  TX bytes:55616 (55.6 KB)

tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:192.168.53.1  P-t-P:192.168.53.1  Mask:255.255.255.0
          inet6 addr: fe80::8013:80f9:634:883c/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP 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:500
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

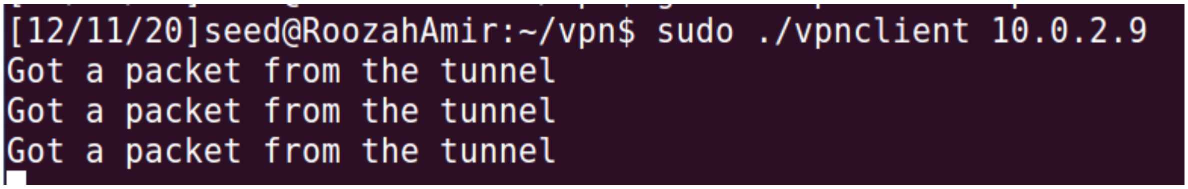
We enabled the IP forwarding for the Server VM to act like a gateway and not a host.



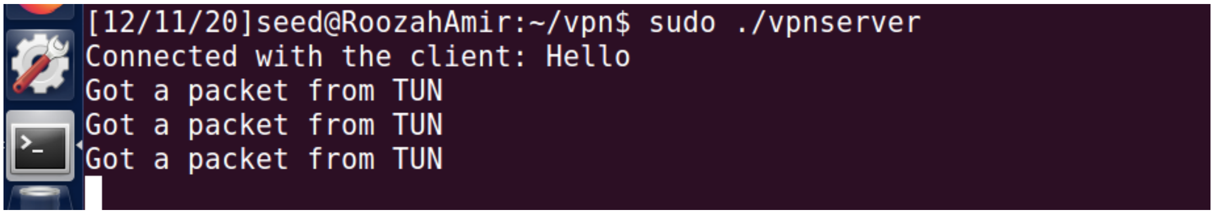
```
[12/11/20]seed@RoozahAmir:~/vpn$ sudo sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
[12/11/20]seed@RoozahAmir:~/vpn$ 
```

Step 2: Client VM

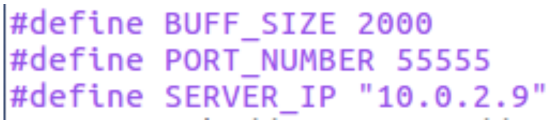
In this step, we create the Client VM end of the tunnel by creating the TUN interface and activating it by assigning an IP Address like we did in Step 1. When we ran the vpnclient code and connected to the Server VM IP address, the Client and Server VM were connected – “Connected with the client.” We altered the vpnclient code by assigning the Server VM IP address. We established a VPN tunnel.



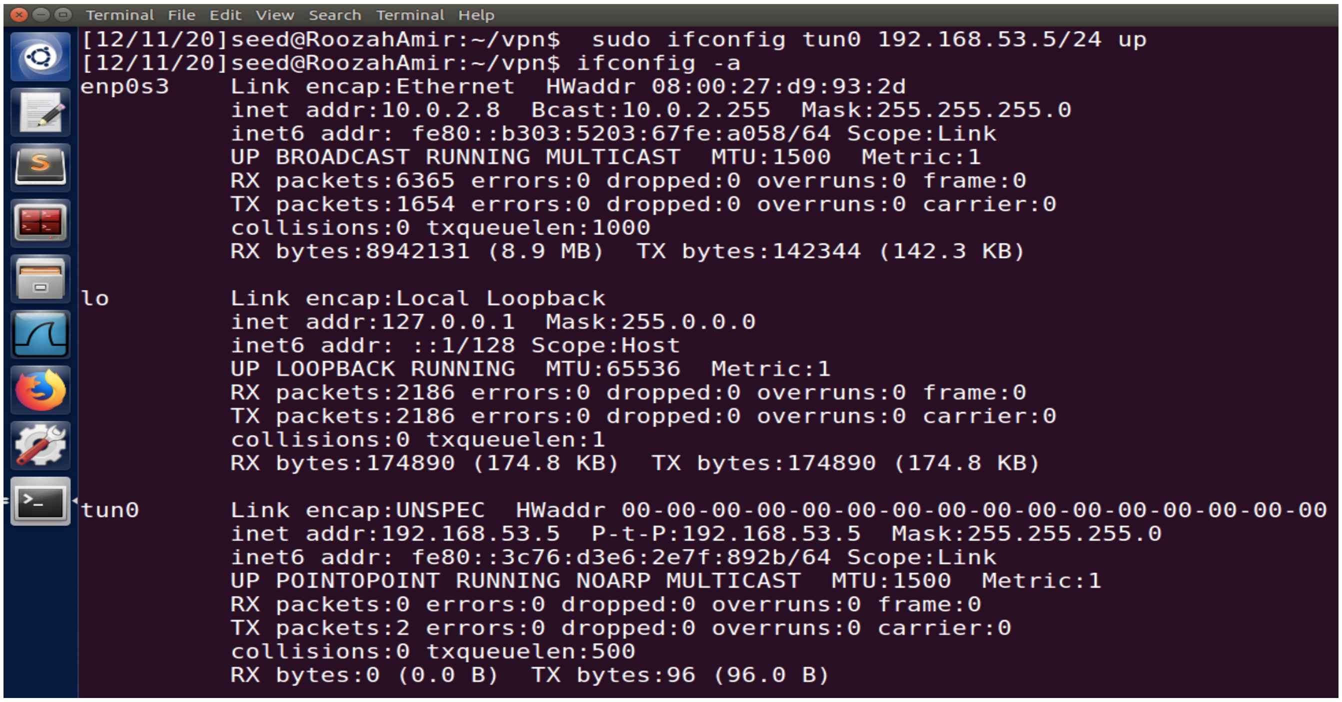
```
[12/11/20]seed@RoozahAmir:~/vpn$ sudo ./vpnclient 10.0.2.9
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel
```



```
[12/11/20]seed@RoozahAmir:~/vpn$ sudo ./vpnserver
Connected with the client: Hello
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
```



```
#define BUFF_SIZE 2000
#define PORT_NUMBER 55555
#define SERVER_IP "10.0.2.9"
```



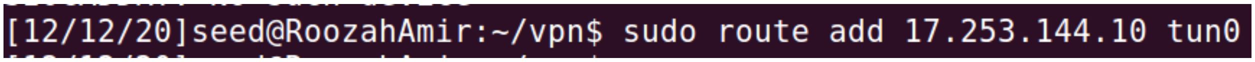
```
[12/11/20]seed@RoozahAmir:~/vpn$  sudo ifconfig tun0 192.168.53.5/24 up
[12/11/20]seed@RoozahAmir:~/vpn$ ifconfig -a
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:d9:93:2d
          inet addr:10.0.2.8  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::b303:5203:67fe:a058/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6365 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1654 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8942131 (8.9 MB)  TX bytes:142344 (142.3 KB)

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:2186 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2186 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:174890 (174.8 KB)  TX bytes:174890 (174.8 KB)

tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:192.168.53.5  P-t-P:192.168.53.5  Mask:255.255.255.0
          inet6 addr: fe80::3c76:d3e6:2e7f:892b/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:0 (0.0 B)  TX bytes:96 (96.0 B)
```

Step 3: Routing

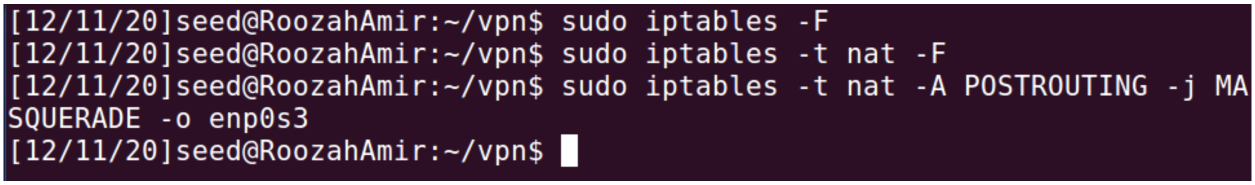
Now that the VPN tunnel is created, we set up a routing path on the Client and Server VM where all packets from apple.com IP address will be directed through TUN interface. This ensures that the traffic from our blocked website will go through the VPN tunnel.



```
[12/12/20]seed@RoozahAmir:~/vpn$ sudo route add 17.253.144.10 tun0
```

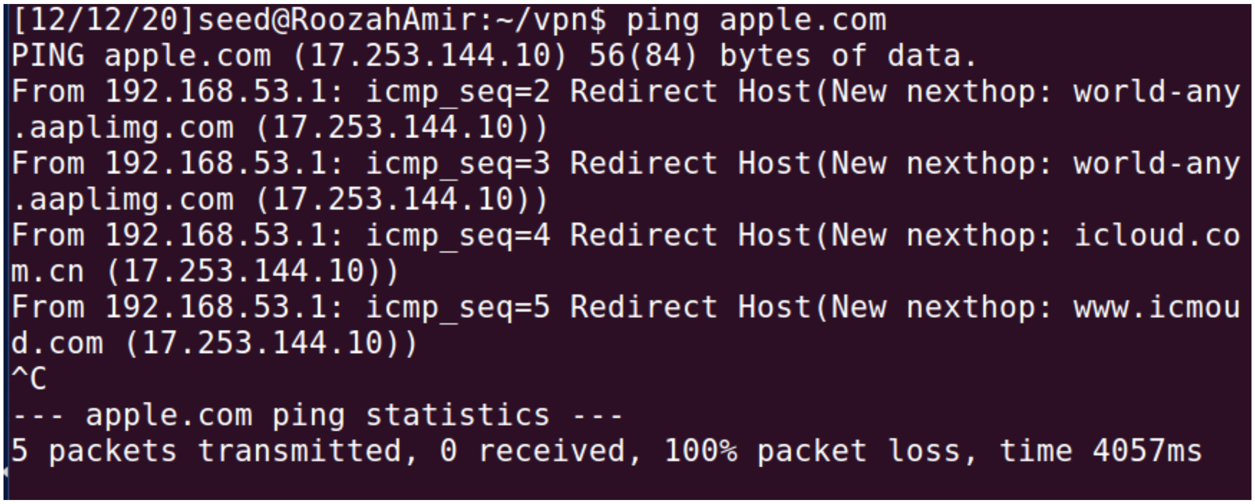
Step 4: Set up NAT on Server VM

NAT has to be set up like a gateway so it can route the packets for the Client VM TUN interface IP address to the Server VPN. This way the packets can be delivered through the VPN tunnel to the Client VM. All the packets that come from the Server VM have the VM's IP address as the source IP address, this is achieved by creating another NAT on the Server VM. We clean all iptables rules so we can add a rule that reroutes the NAT network adapter to let us go around the firewall when we access our blocked website (apple.com).

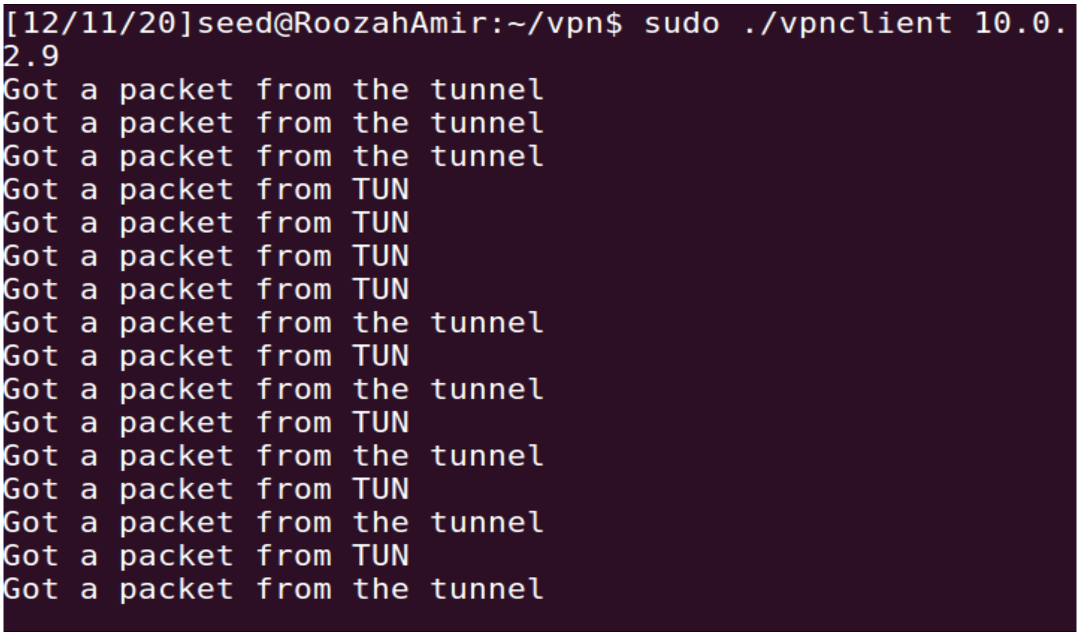


```
[12/11/20]seed@RoozahAmir:~/vpn$ sudo iptables -F
[12/11/20]seed@RoozahAmir:~/vpn$ sudo iptables -t nat -F
[12/11/20]seed@RoozahAmir:~/vpn$ sudo iptables -t nat -A POSTROUTING -j MA
SQUERADE -o enp0s3
[12/11/20]seed@RoozahAmir:~/vpn$ 
```

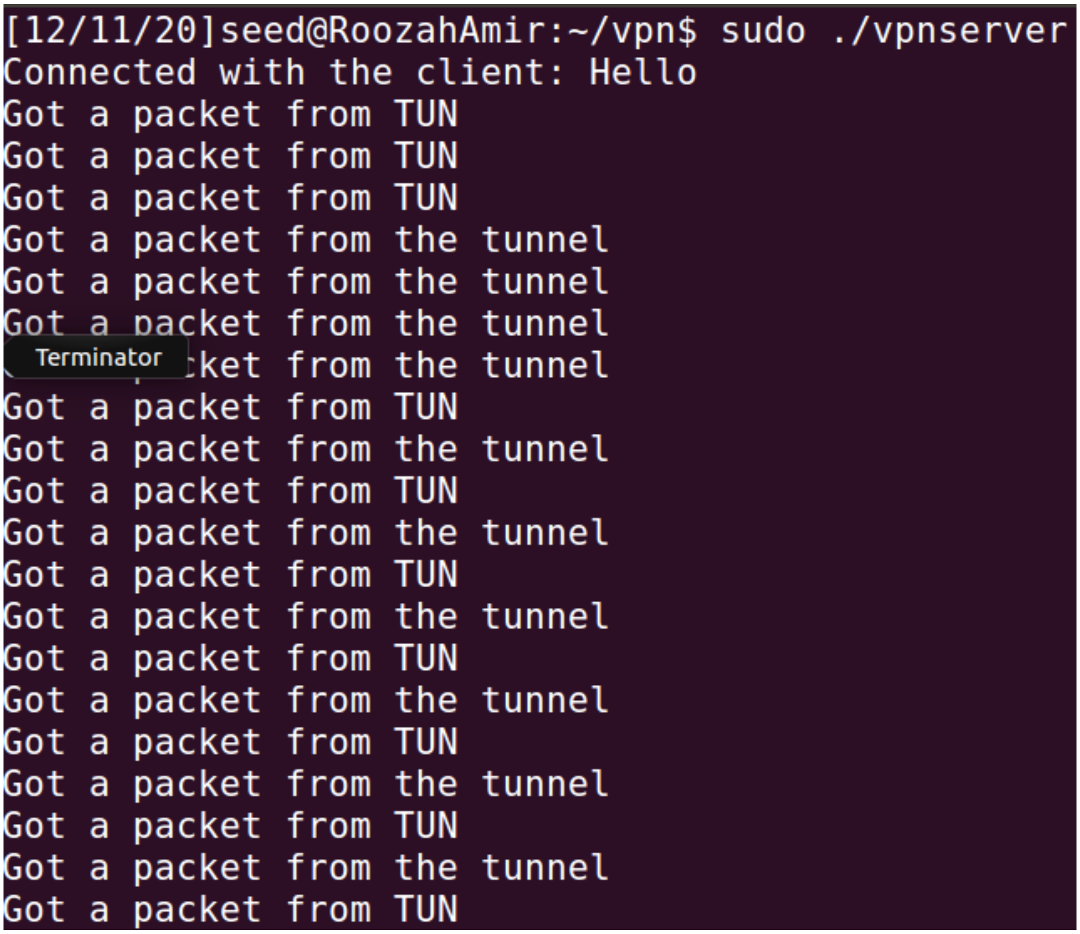
We go back to the Client VM and ping apple.com. As you can see, the Client VM can bypass the firewall and access apple.com through the VPN tunnel. As we ping apple.com you can see the packets from the VPN tunnel on Client and Server VMs.



```
[12/12/20]seed@RoozahAmir:~/vpn$ ping apple.com
PING apple.com (17.253.144.10) 56(84) bytes of data.
From 192.168.53.1: icmp_seq=2 Redirect Host(New nexthop: world-any
.aaplimg.com (17.253.144.10))
From 192.168.53.1: icmp_seq=3 Redirect Host(New nexthop: world-any
.aaplimg.com (17.253.144.10))
From 192.168.53.1: icmp_seq=4 Redirect Host(New nexthop: icloud.co
m.cn (17.253.144.10))
From 192.168.53.1: icmp_seq=5 Redirect Host(New nexthop: www.icmou
d.com (17.253.144.10))
^C
--- apple.com ping statistics ---
5 packets transmitted, 0 received, 100% packet loss, time 4057ms
```

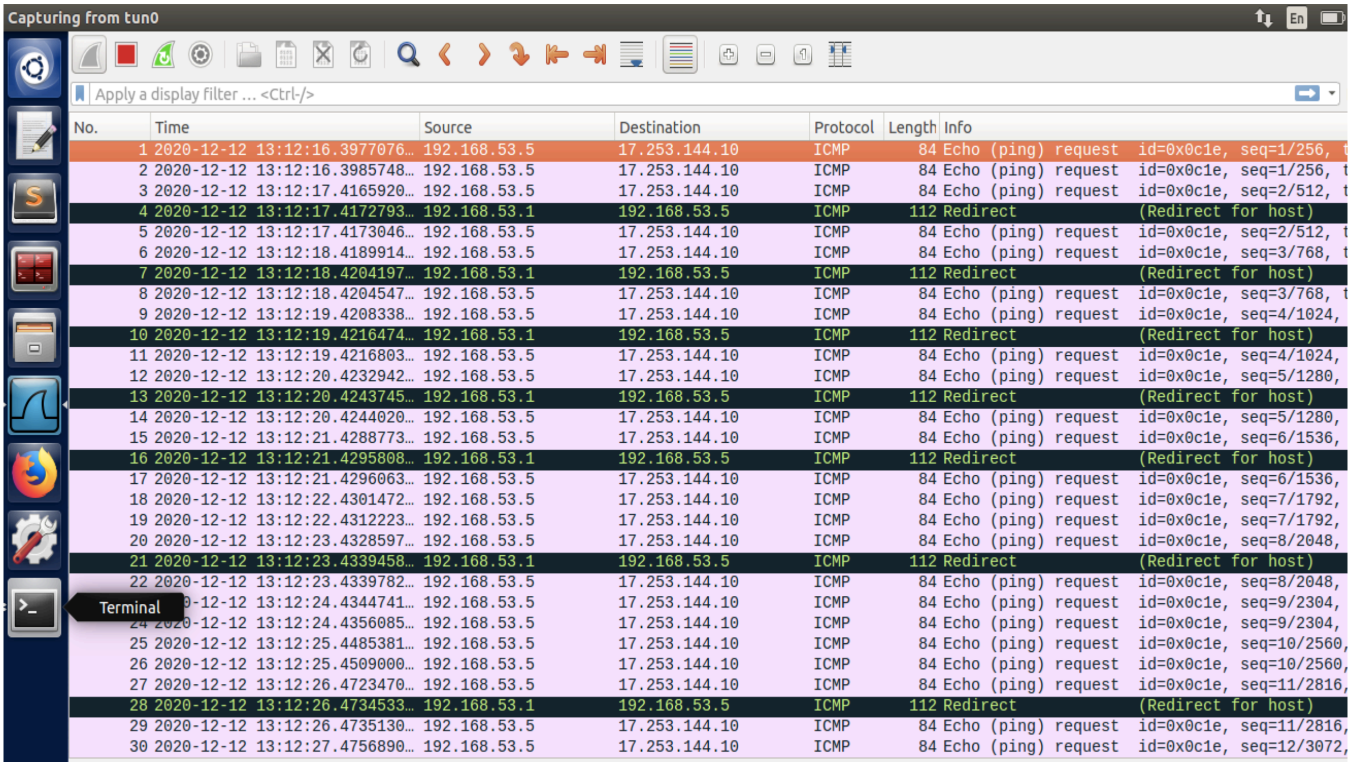


```
[12/11/20]seed@RoozahAmir:~/vpn$ sudo ./vpnclient 10.0.
2.9
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
```

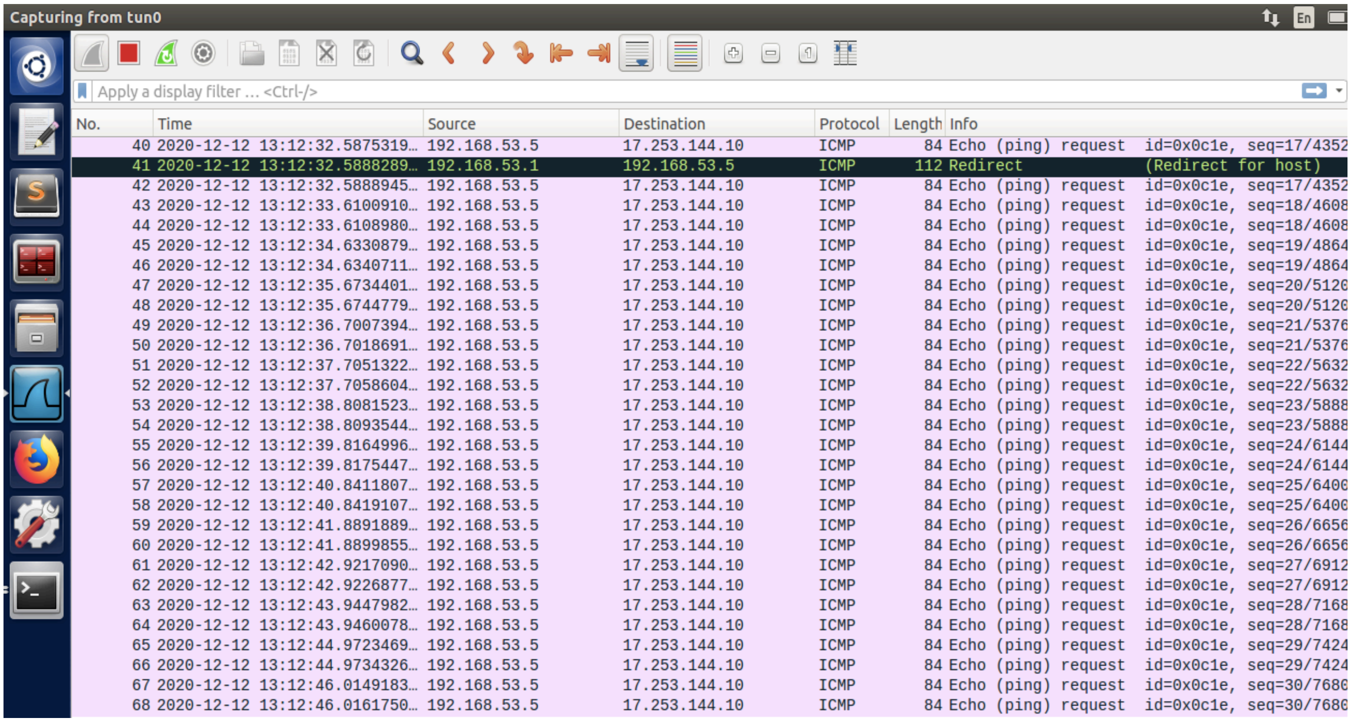


```
[12/11/20]seed@RoozahAmir:~/vpn$ sudo ./vpnserver
Connected with the client: Hello
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel
       cket from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
```

To verify that the traffic went through the VPN tunnel, we can use Wireshark to check. If you zoom in, you can see that the traffic is going through the VPN tunnel because it displays the TUN interface IP address of the Client and Server VPN and the apple.com IP address.



```
No.  Time                          Source          Destination     Protocol Length Info
   1 2020-12-12 13:12:16.3977076… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=1/256, t
   2 2020-12-12 13:12:16.3985748… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=1/256, t
   3 2020-12-12 13:12:17.4165920… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=2/512, t
   4 2020-12-12 13:12:17.4172793… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
   5 2020-12-12 13:12:17.4173046… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=2/512, t
   6 2020-12-12 13:12:18.4189914… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=3/768, t
   7 2020-12-12 13:12:18.4204197… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
   8 2020-12-12 13:12:18.4204547… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=3/768, t
   9 2020-12-12 13:12:19.4208338… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=4/1024,
  10 2020-12-12 13:12:19.4216474… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
  11 2020-12-12 13:12:19.4216803… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=4/1024,
  12 2020-12-12 13:12:20.4232942… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=5/1280,
  13 2020-12-12 13:12:20.4243745… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
  14 2020-12-12 13:12:20.4244020… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=5/1280,
  15 2020-12-12 13:12:21.4288773… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=6/1536,
  16 2020-12-12 13:12:21.4295808… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
  17 2020-12-12 13:12:21.4296063… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=6/1536,
  18 2020-12-12 13:12:22.4301472… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=7/1792,
  19 2020-12-12 13:12:22.4312223… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=7/1792,
  20 2020-12-12 13:12:23.4328597… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=8/2048,
  21 2020-12-12 13:12:23.4339458… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
  22 2020-12-12 13:12:23.4339782… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=8/2048,
        0-12-12 13:12:24.4344741… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=9/2304,
  24 2020-12-12 13:12:24.4356085… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=9/2304,
  25 2020-12-12 13:12:25.4485381… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=10/2560,
  26 2020-12-12 13:12:25.4509000… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=10/2560,
  27 2020-12-12 13:12:26.4723470… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=11/2816,
  28 2020-12-12 13:12:26.4734533… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
  29 2020-12-12 13:12:26.4735130… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=11/2816,
  30 2020-12-12 13:12:27.4756890… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=12/3072,
```



```
No.  Time                          Source          Destination     Protocol Length Info
  40 2020-12-12 13:12:32.5875319… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=17/4352
  41 2020-12-12 13:12:32.5888289… 192.168.53.1    192.168.53.5    ICMP       112 Redirect             (Redirect for host)
  42 2020-12-12 13:12:32.5888945… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=17/4352
  43 2020-12-12 13:12:33.6100910… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=18/4608
  44 2020-12-12 13:12:33.6108980… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=18/4608
  45 2020-12-12 13:12:34.6330879… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=19/4864
  46 2020-12-12 13:12:34.6340711… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=19/4864
  47 2020-12-12 13:12:35.6734401… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=20/5120
  48 2020-12-12 13:12:35.6744779… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=20/5120
  49 2020-12-12 13:12:36.7007394… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=21/5376
  50 2020-12-12 13:12:36.7018691… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=21/5376
  51 2020-12-12 13:12:37.7051322… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=22/5632
  52 2020-12-12 13:12:37.7058604… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=22/5632
  53 2020-12-12 13:12:38.8081523… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=23/5888
  54 2020-12-12 13:12:38.8093544… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=23/5888
  55 2020-12-12 13:12:39.8164996… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=24/6144
  56 2020-12-12 13:12:39.8175447… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=24/6144
  57 2020-12-12 13:12:40.8411807… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=25/6400
  58 2020-12-12 13:12:40.8419107… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=25/6400
  59 2020-12-12 13:12:41.8891889… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=26/6656
  60 2020-12-12 13:12:41.8899855… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=26/6656
  61 2020-12-12 13:12:42.9217090… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=27/6912
  62 2020-12-12 13:12:42.9226877… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=27/6912
  63 2020-12-12 13:12:43.9447982… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=28/7168
  64 2020-12-12 13:12:43.9460078… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=28/7168
  65 2020-12-12 13:12:44.9723469… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=29/7424
  66 2020-12-12 13:12:44.9734326… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=29/7424
  67 2020-12-12 13:12:46.0149183… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=30/7680
  68 2020-12-12 13:12:46.0161750… 192.168.53.5    17.253.144.10   ICMP        84 Echo (ping) request  id=0x0c1e, seq=30/7680
```