CNS LAB

LAB - 6

Virtual Private Network Lab

NAME : SIRI S

SEMESTER : 5

SECTION :H

SRN : PES1UG19CS485

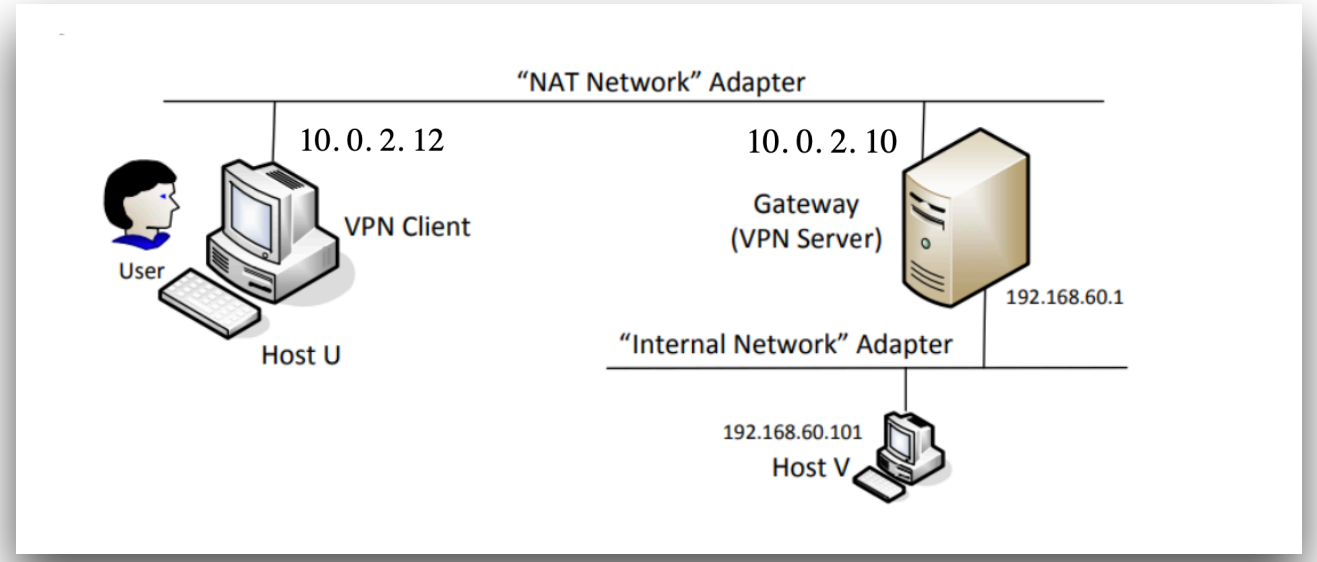
LAB SETUP:

VPN SERVER: 10. 0. 2. 10

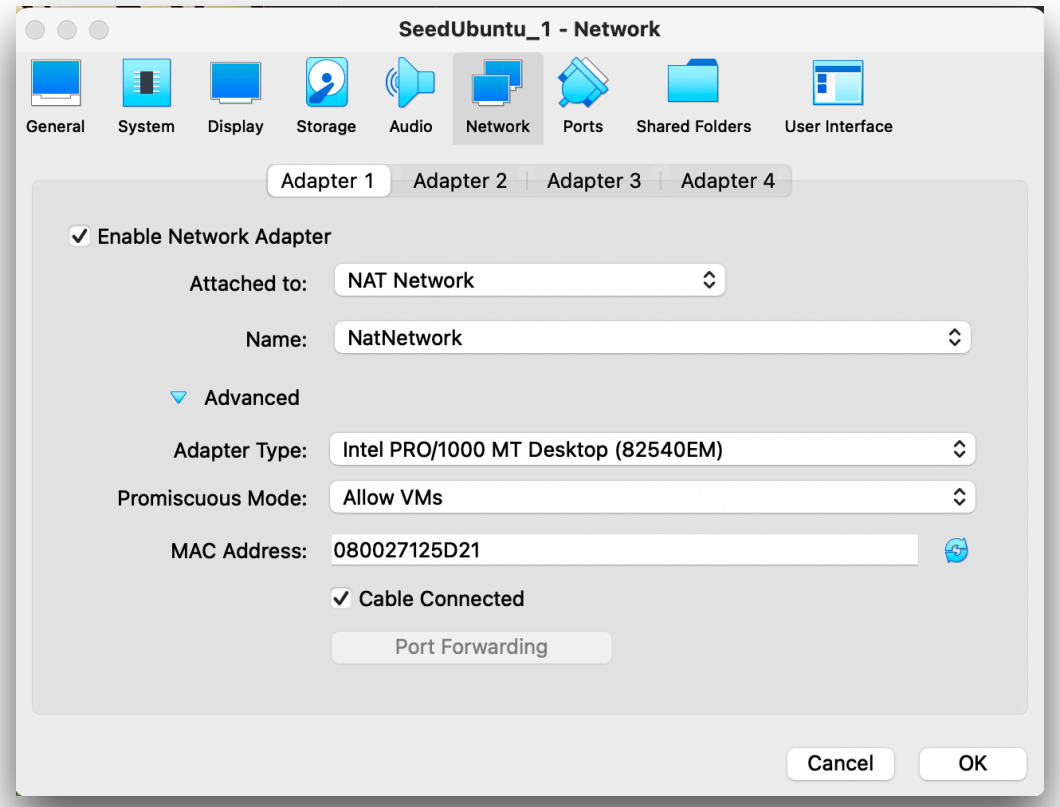
VPN CLIENT: 10. 0. 2. 12

VPN HOST: 10. 0. 2. 11

Task 1: VM Setup

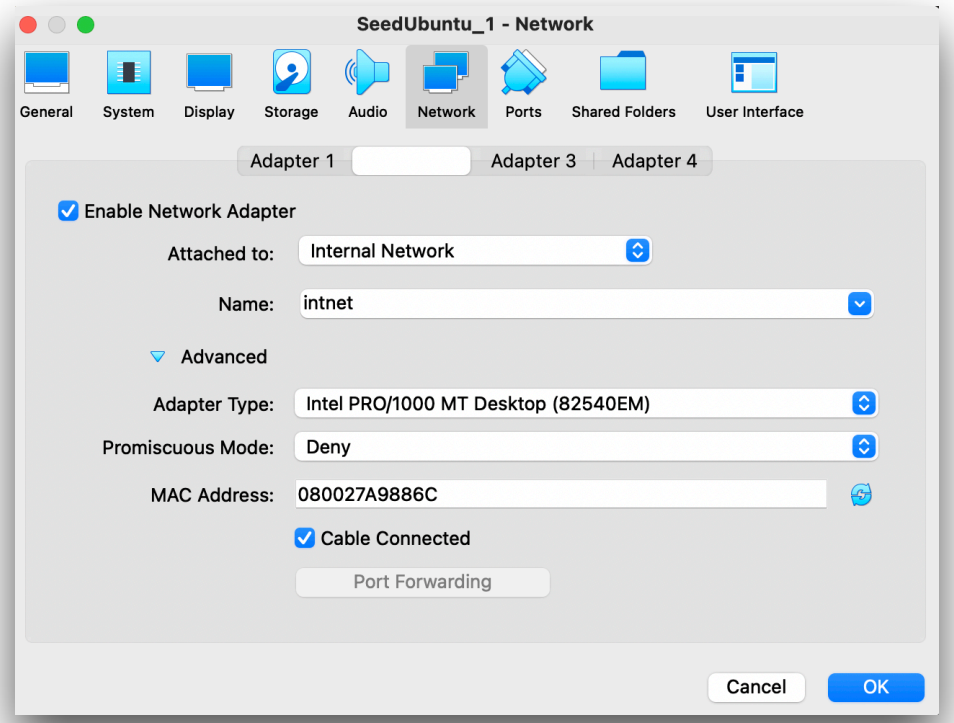


We need to establish a set up like above. In order to do this, the client and the server will have a NAT NETWORK as usual:



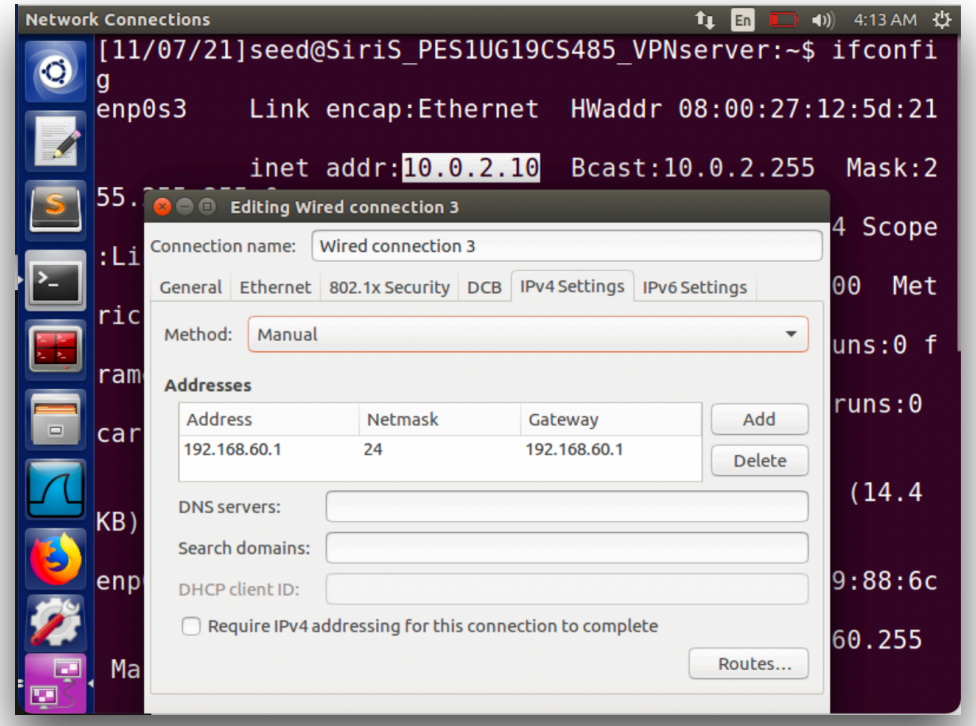
Screenshot of the Network settings in Server VM

However, the server and the host machines will be connected through an ‘internal network’ so that the client and the host have no connection:

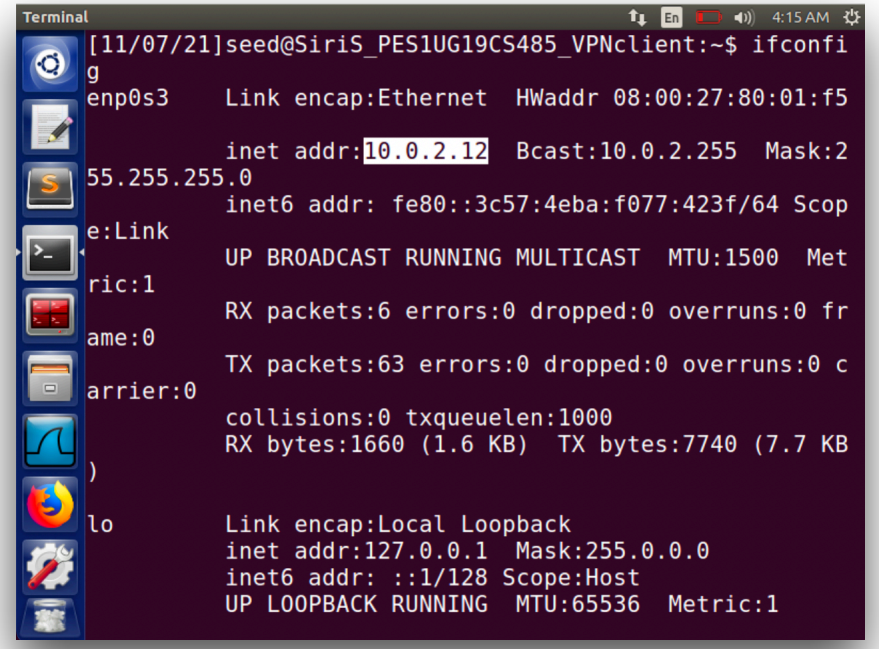


Screenshot of the Network settings in Server VM

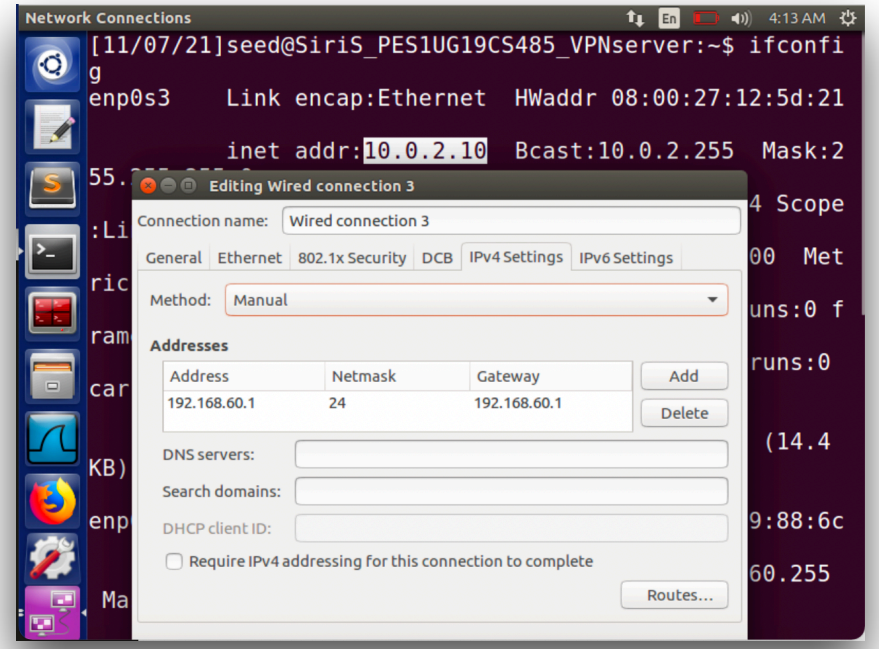
Now we will make changes in the edit connections:



Server VM



Client VM



Server VM

Therefore, from the above screenshots, we can see that:

VPN client – Adapter 1 – NAT Network

VPN Server – Adapter 1 – NAT network, Adapter 2 – Internal Network

HOST V – Adapter 1 – Internal Network

These connections are established

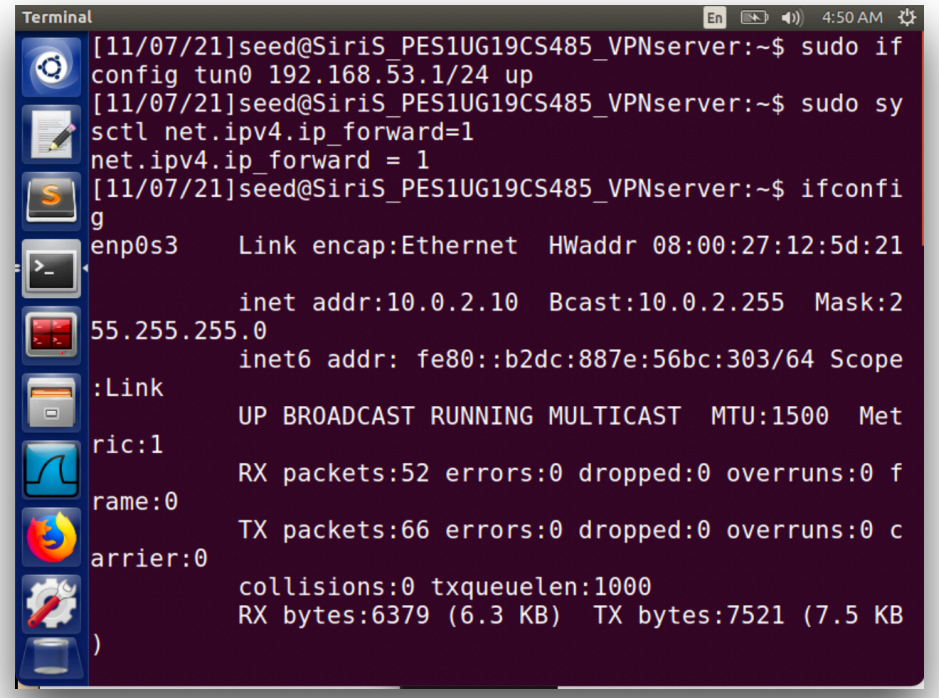
Task 2: Creating a VPN Tunnel using TUN/TAP

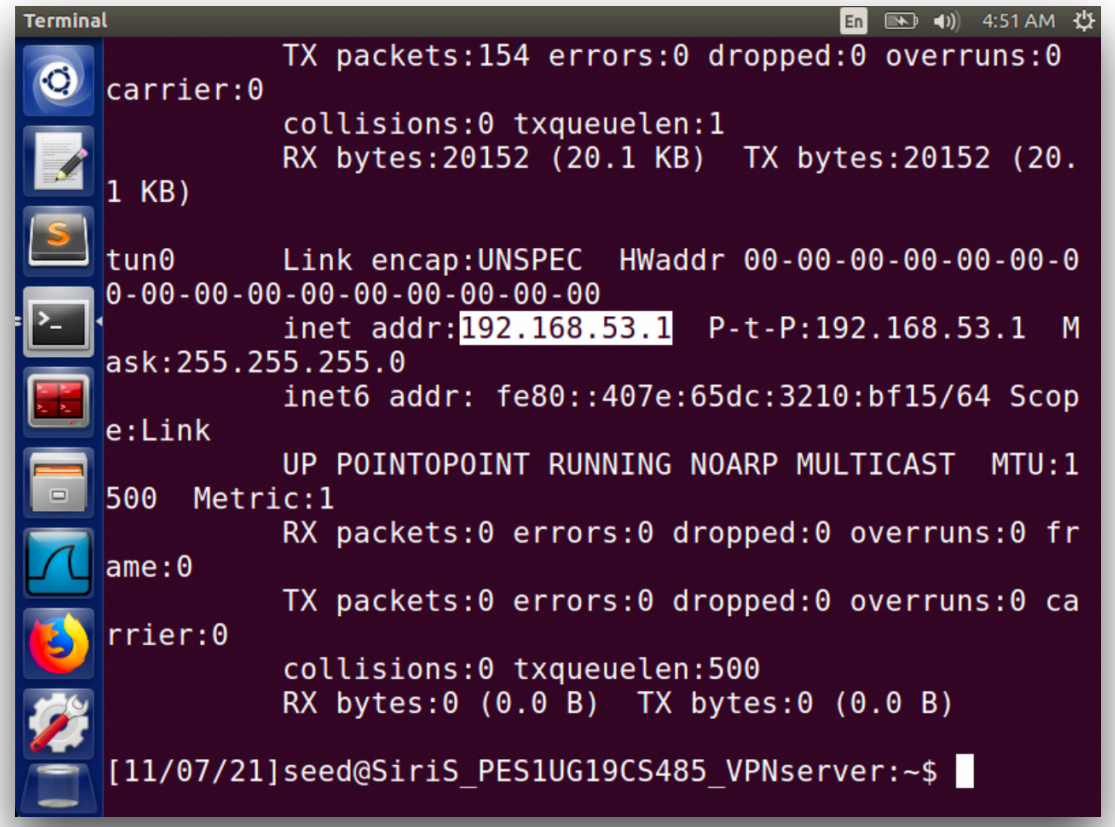
Step 1: Run VPN server and set it's IP address of the interface

Now we run the vpnserver.c code on the server machine.

Then we assign an IP address to the tun0 inferface and activate it. IP Address assigned: 192.168.53.1/24. We also enable port forwarding.

Upon checking ifconfig -a : we have an established tunnel:

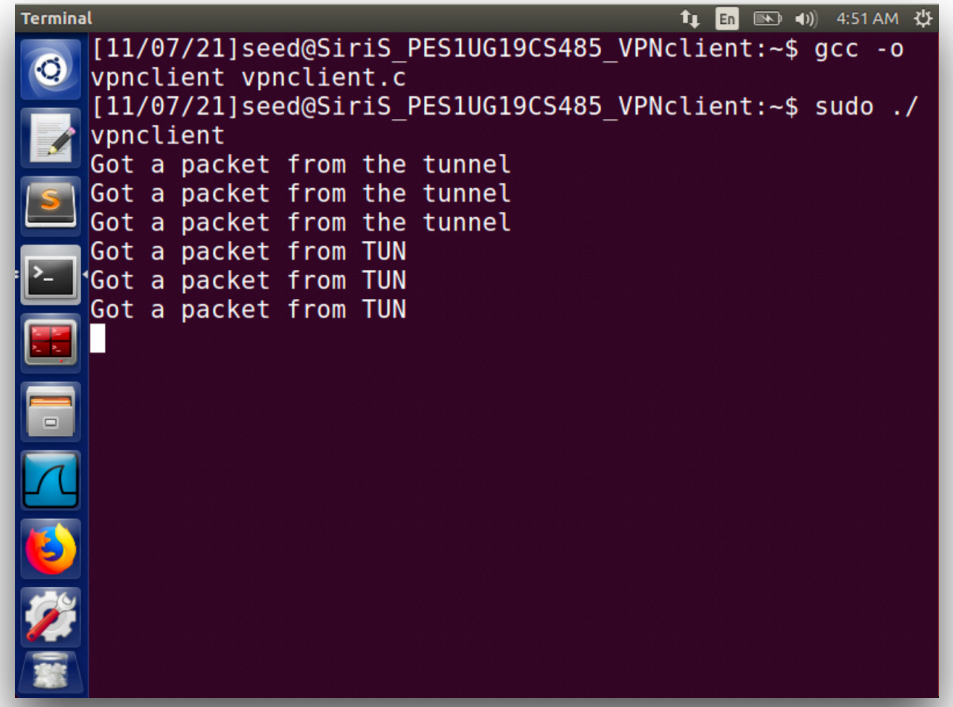




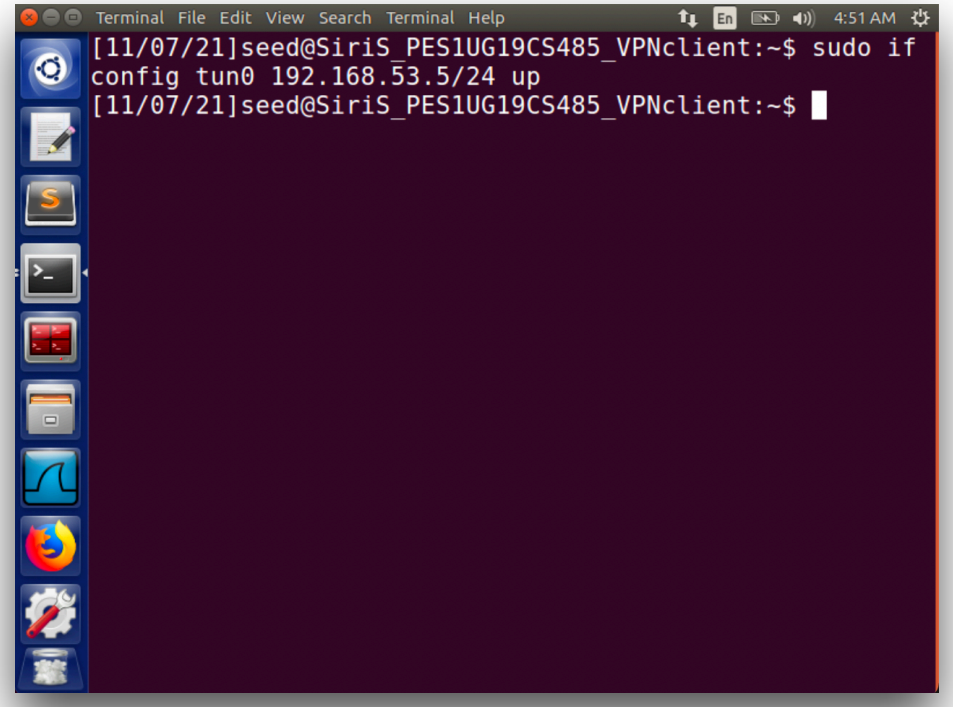
We can see that the tunnel is active. The VPN Server needs to forward packets to other destinations, so it needs to function as a gateway. We need to enable the IP forwarding for a computer to behave like a gateway.

Step 2: Run VPN Client and set IP address of the interface

Now we run the vpnclient.c code on the client machine.

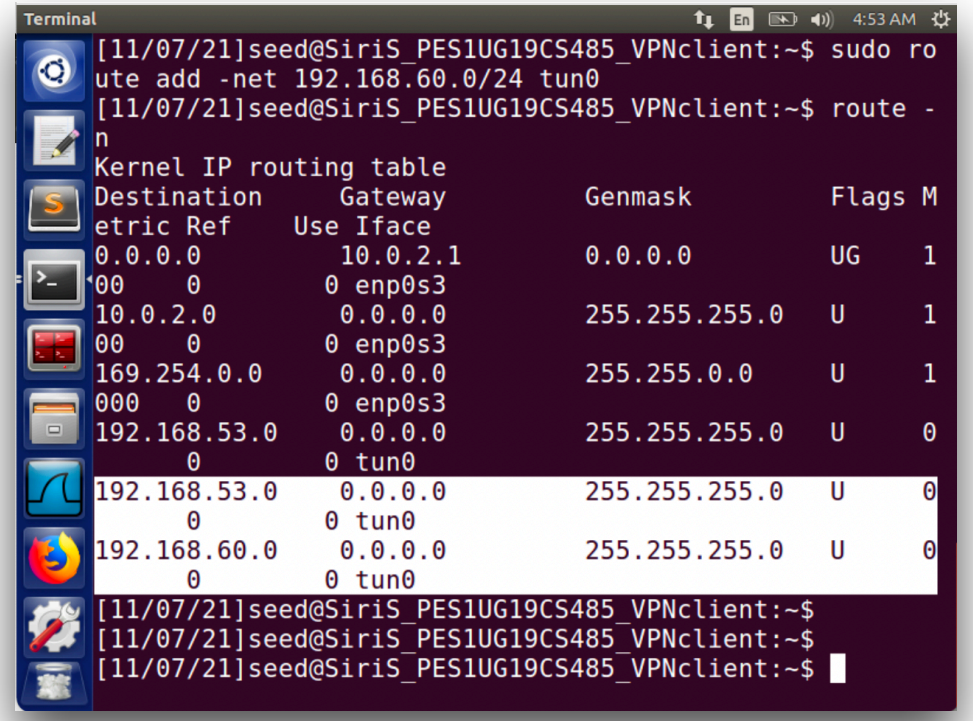


Then we assign an IP address to the tun0 inferface and activate it.
IP Address assigned: 192.168.53.5/24

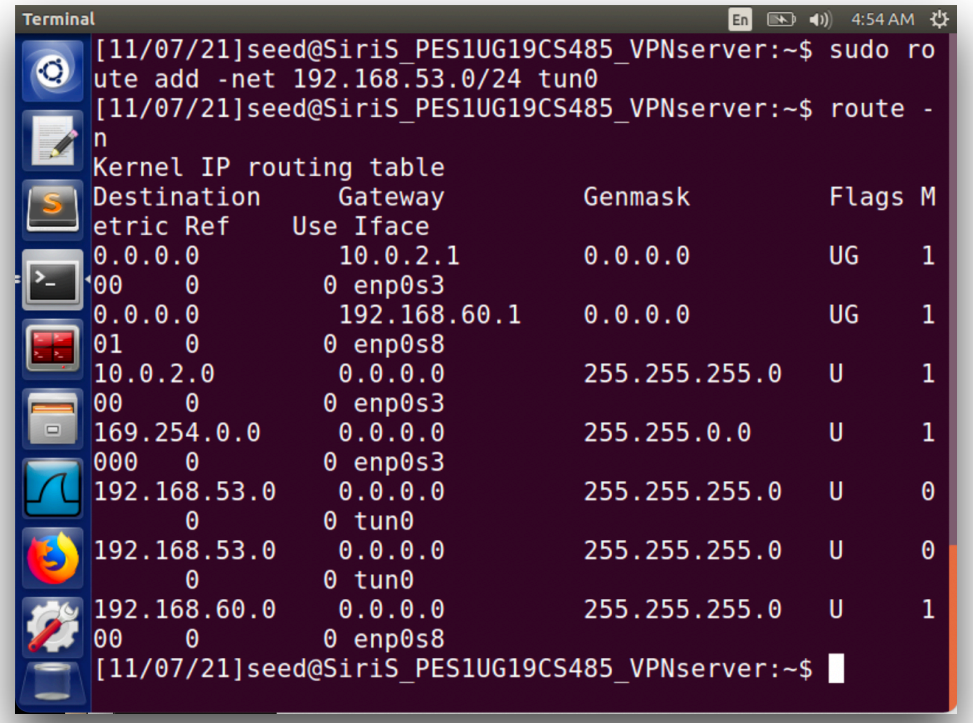


<u>Step 3:</u> Set up routing on Client and Server VMs

Setting up a routing table on the client VM:

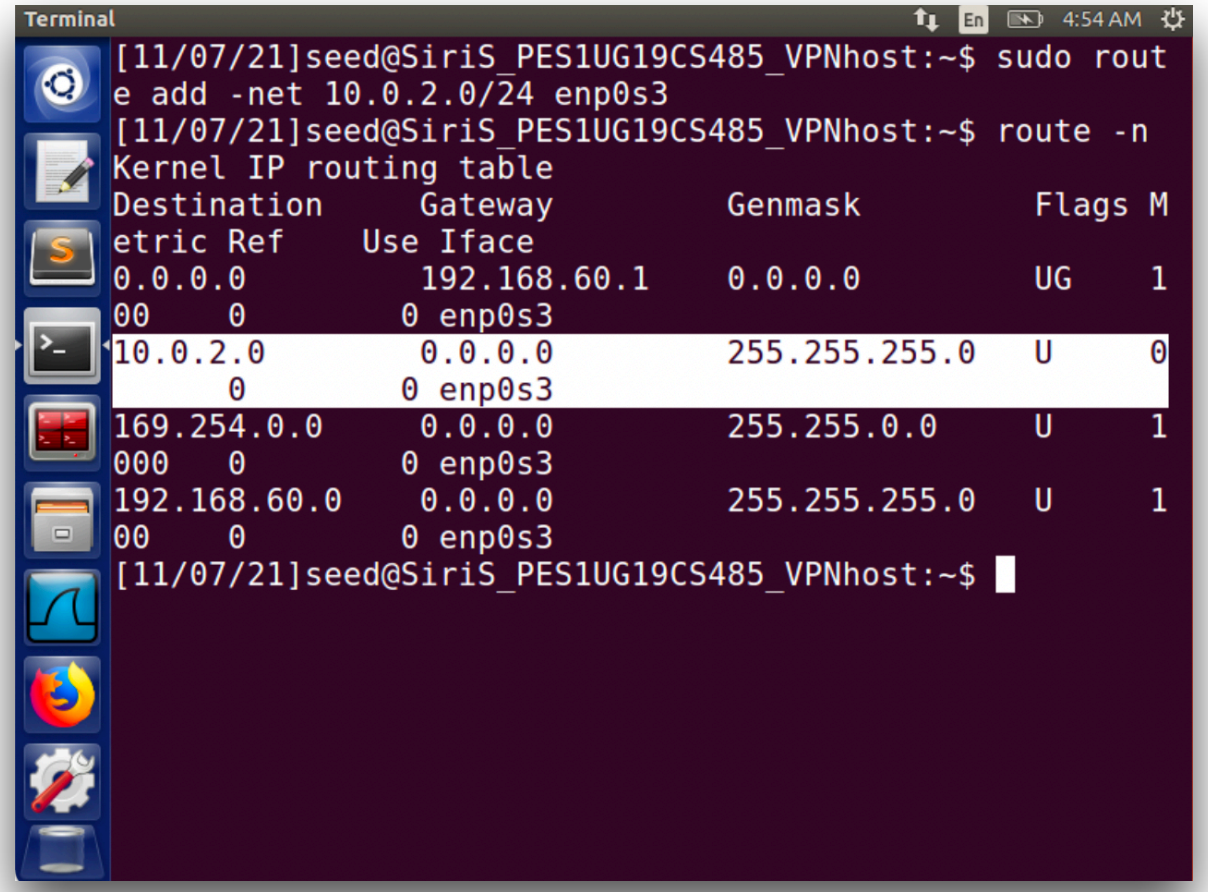


Setting up a routing table on the Server VM:



<u>Step 4:</u> Set up routing on HOST V

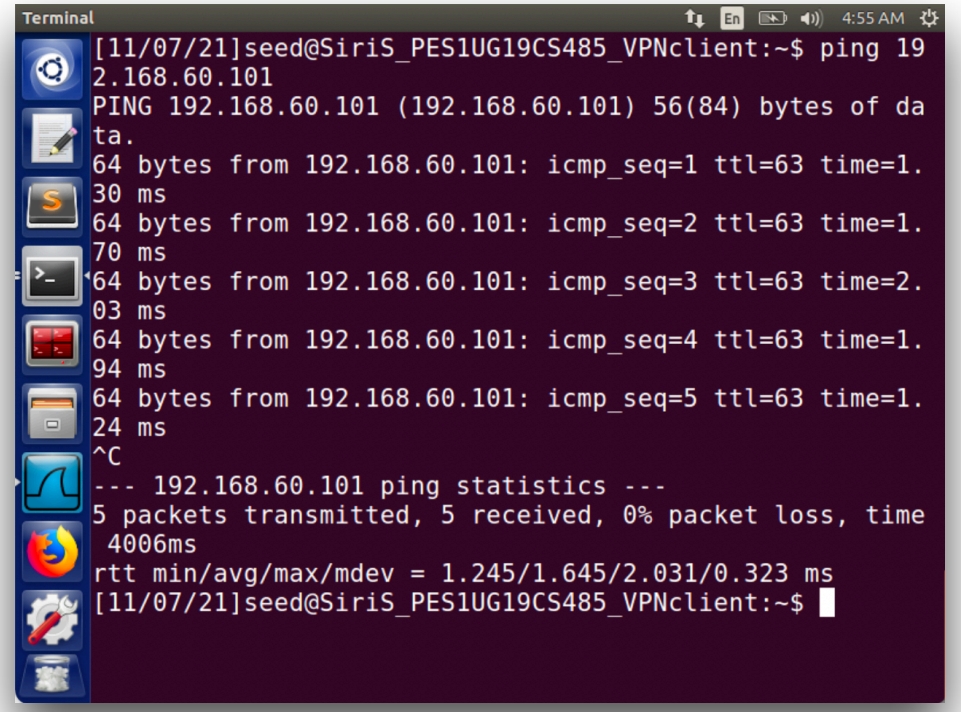
Setting up a routing table on the Host VM:



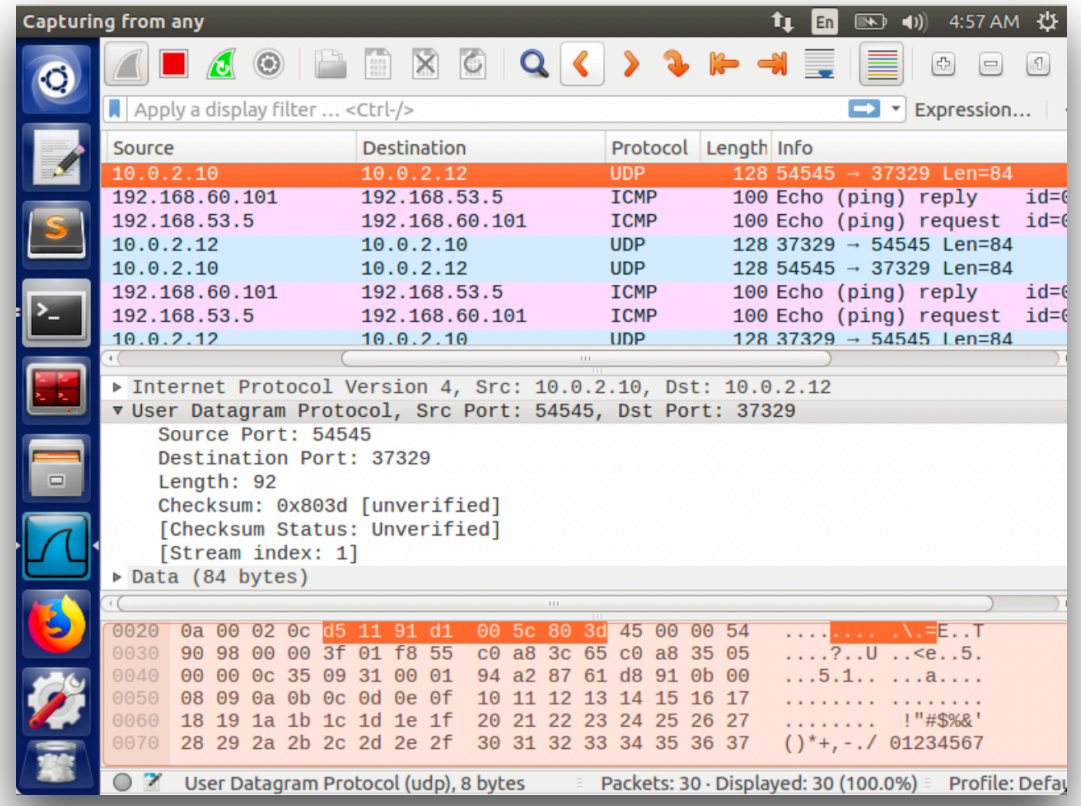
In this step we set up the routing path on host V. This is so, that when host V replies to Host U it needs to go through the routing path through the VPN server. Since, Host V being an internal network does not know where to send the packets and cannot let the packets go through any arbitrary route else the packet is not protected.

Step 5: Test the VPN tunnel (ping and telnet)

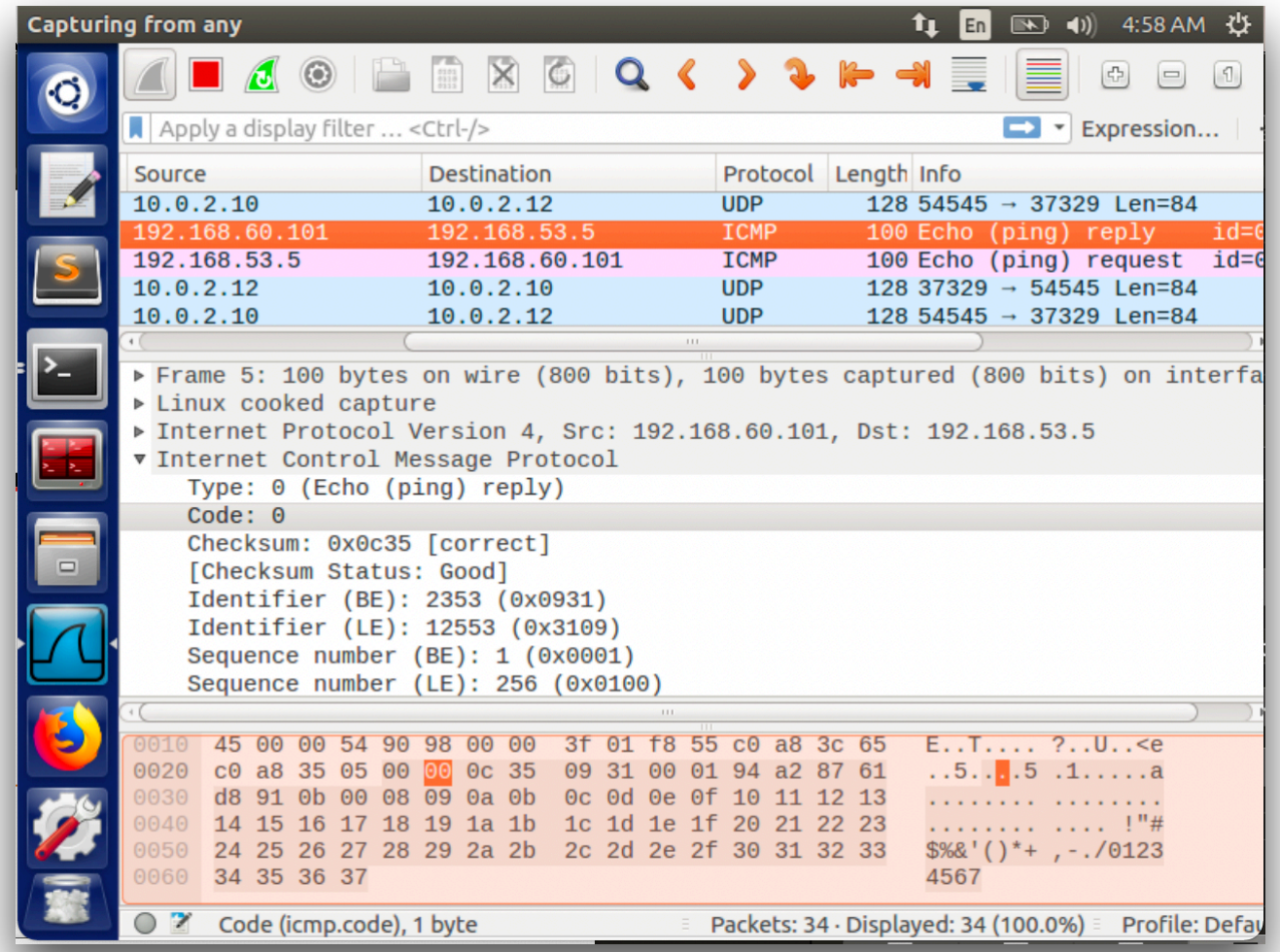
First we will perform the ping command to see if the VPN tunnel has been established:



We are successfully able to ping as seen. The Wireshark screenshot explaining the UDP packet:

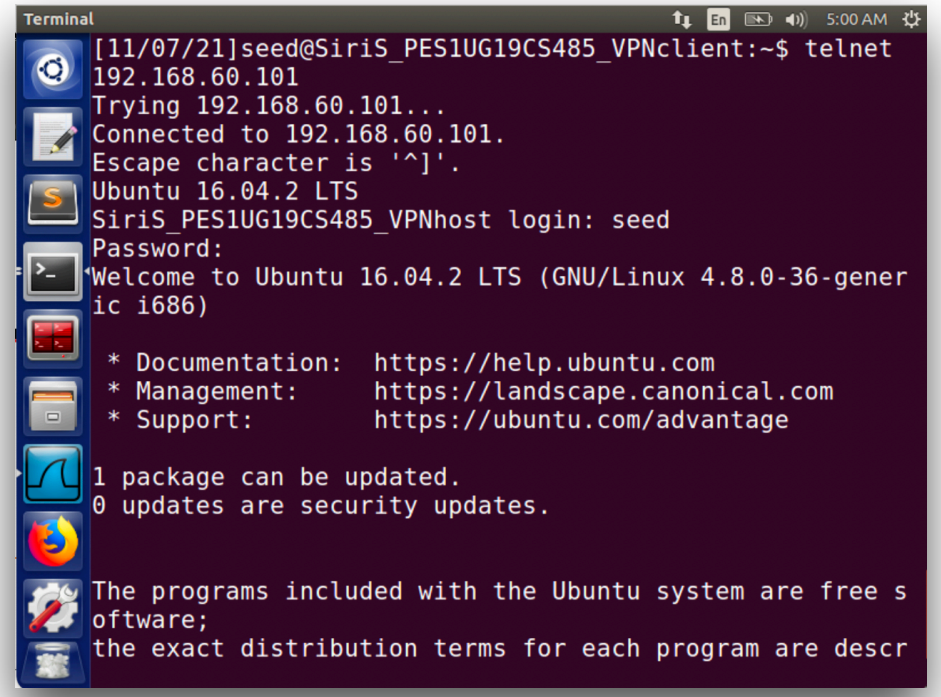


The Wireshark screenshot explaining the ICMP packet:

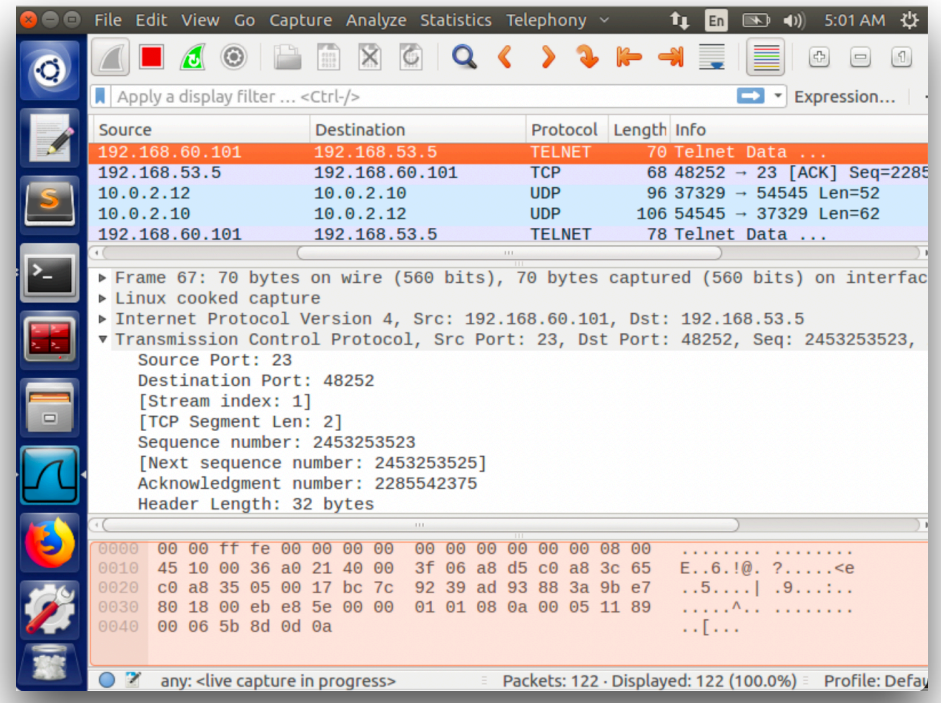


We can see from the wireshark screenshot that packet source and destination 192.168.53.5 (Client- tun0) and 192.168.60.101 (Host V) are the tunnel traffic and the rest are the normal traffic.

Next we will perform the telnet connection to verify if the VPN tunnel has been established:

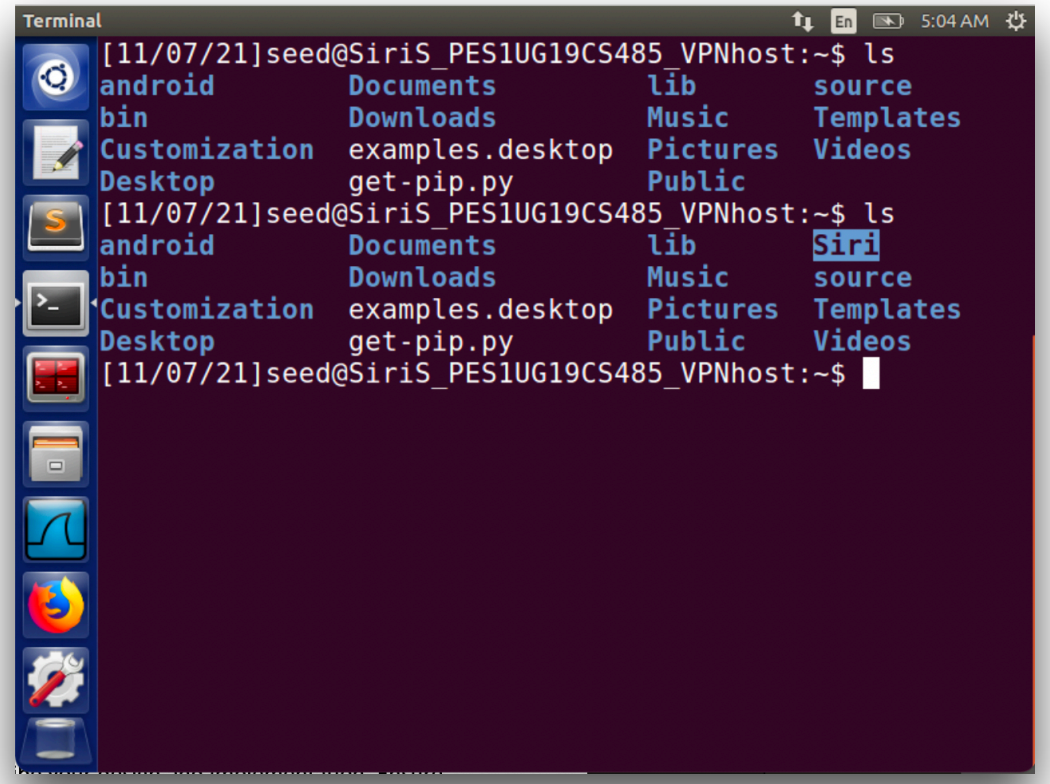


We can see that we are successfully able to establish the telnet connection. Wireshark screenshot to prove it:

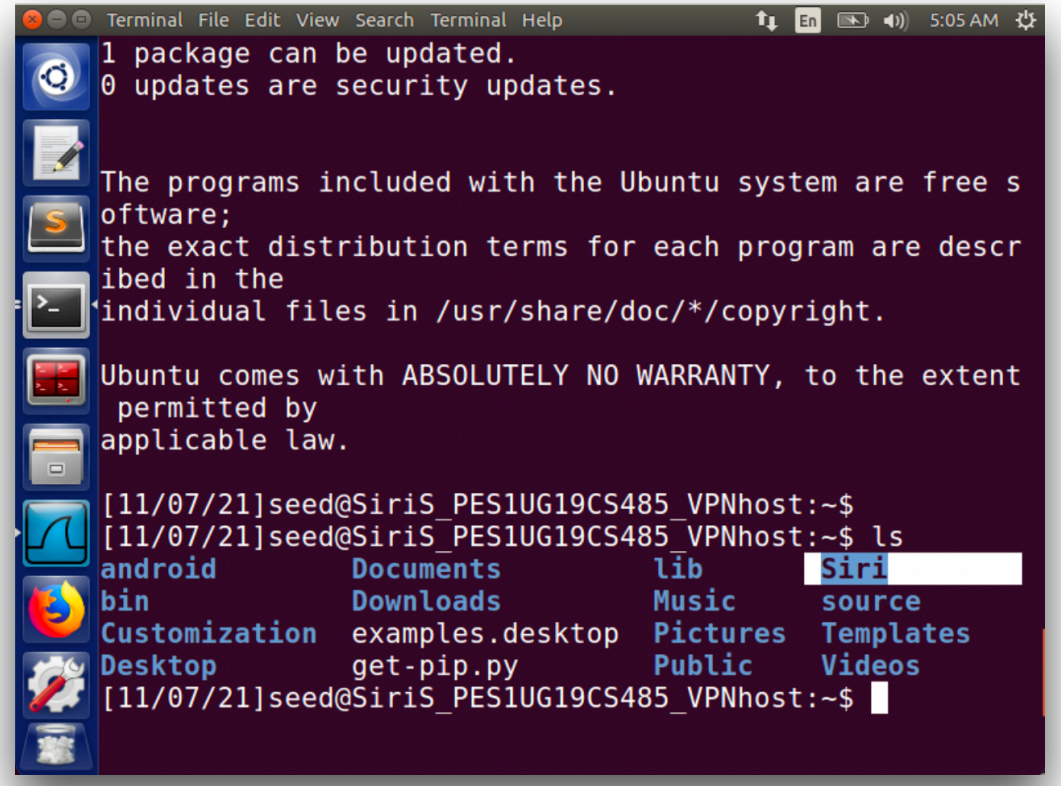


From the above screenshot we can conclude that we were successful in establishing the VPN connection.

We run ‘ls’ command on the Host VM and in addition, create a new folder called ‘Siri’ as seen in the screenshot:

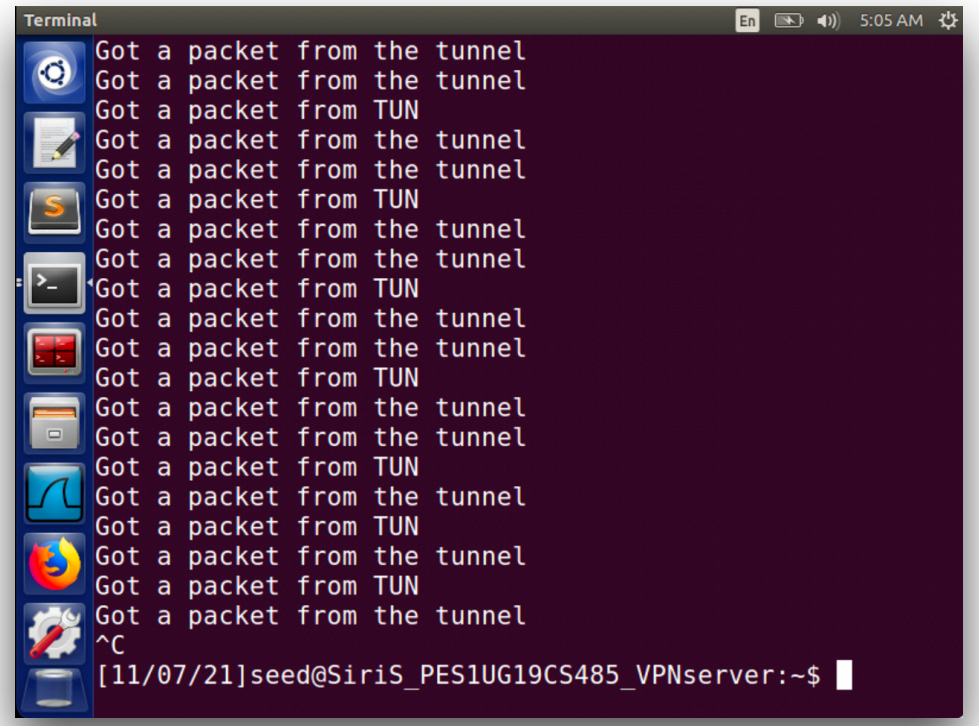


Now when we run ‘ls’ command on the telnet connection, we are able to notice that the new folder create is visible:

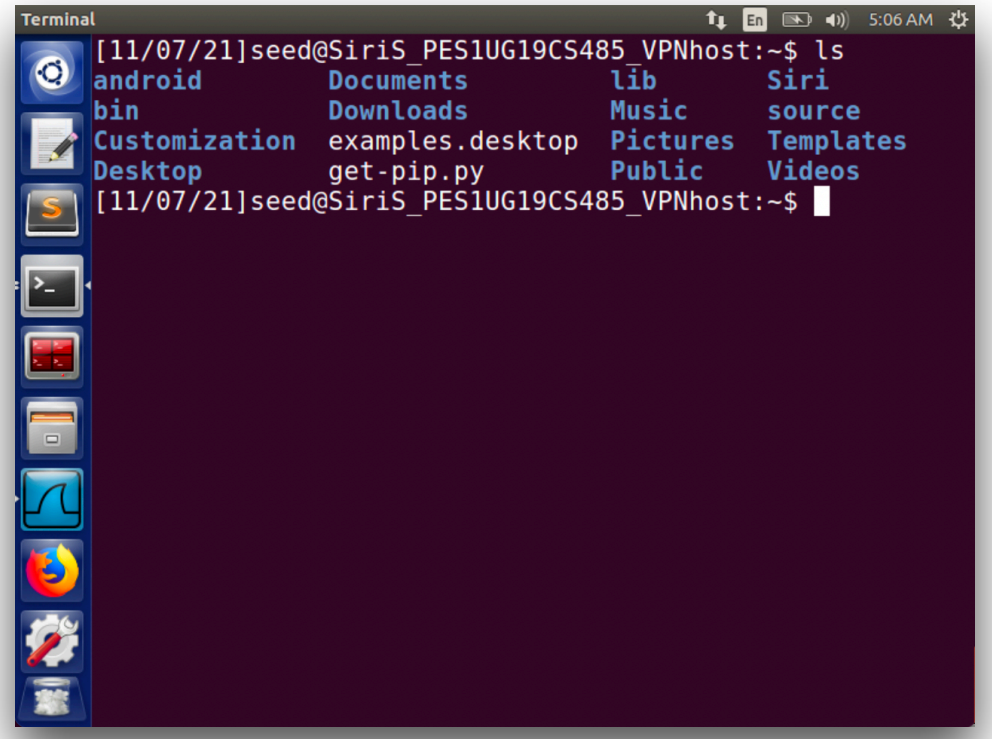


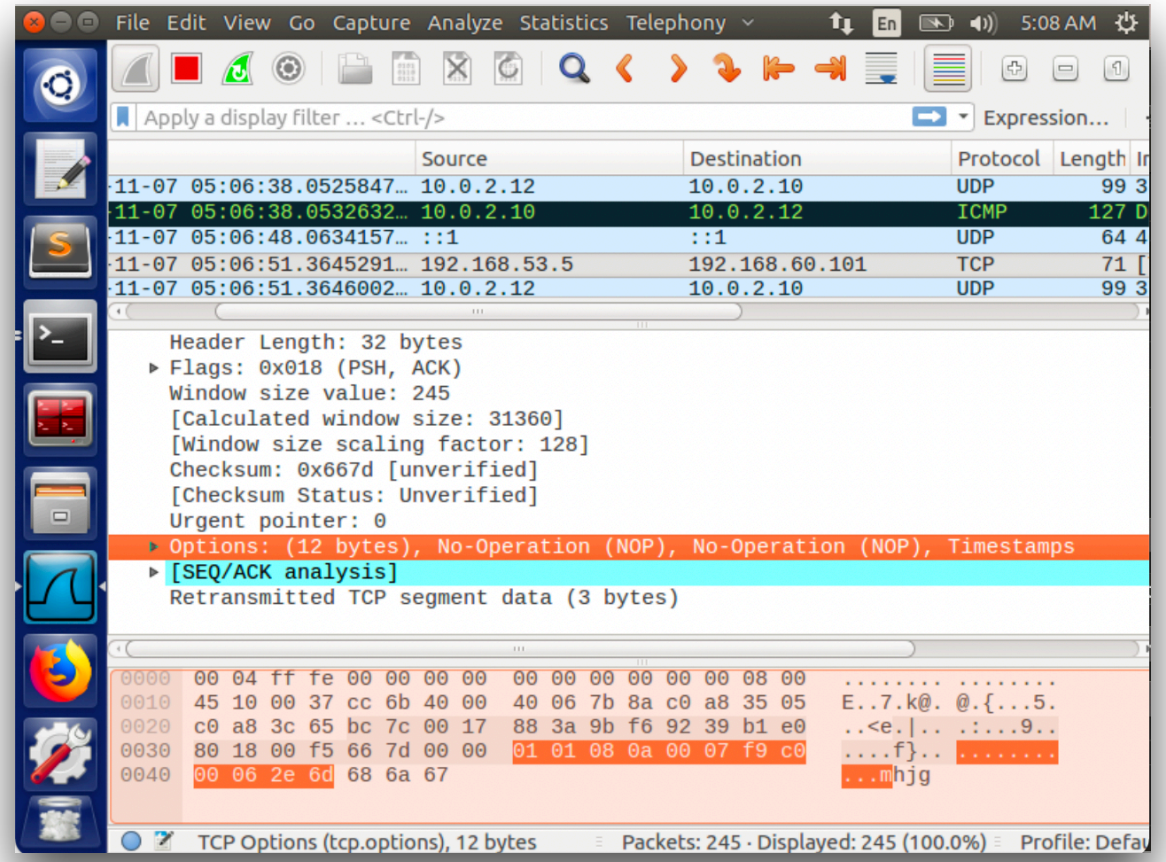
Step 6: Tunnel-Breaking Test

We disconnect the vpnserver program in order to break the VPN tunnel connection as shown in the screenshot:



Now when we try to type ls in the telnet connection, we unable to do so:





As we can see from the Wireshark screenshot, we are getting a TCP redirect message.

And now if we again connect the server machine we can see that the randomly typed command on host U machine is popped up.

The telnet becomes unresponsive when we break the connection and then after reconnecting all the typed characters we had given reaches the telnet server which we are later able to see on VPN client.