

COMPUTER NETWORK

SECURITY

LAB-9

VPN Tunneling

NAME: VISHWAS M

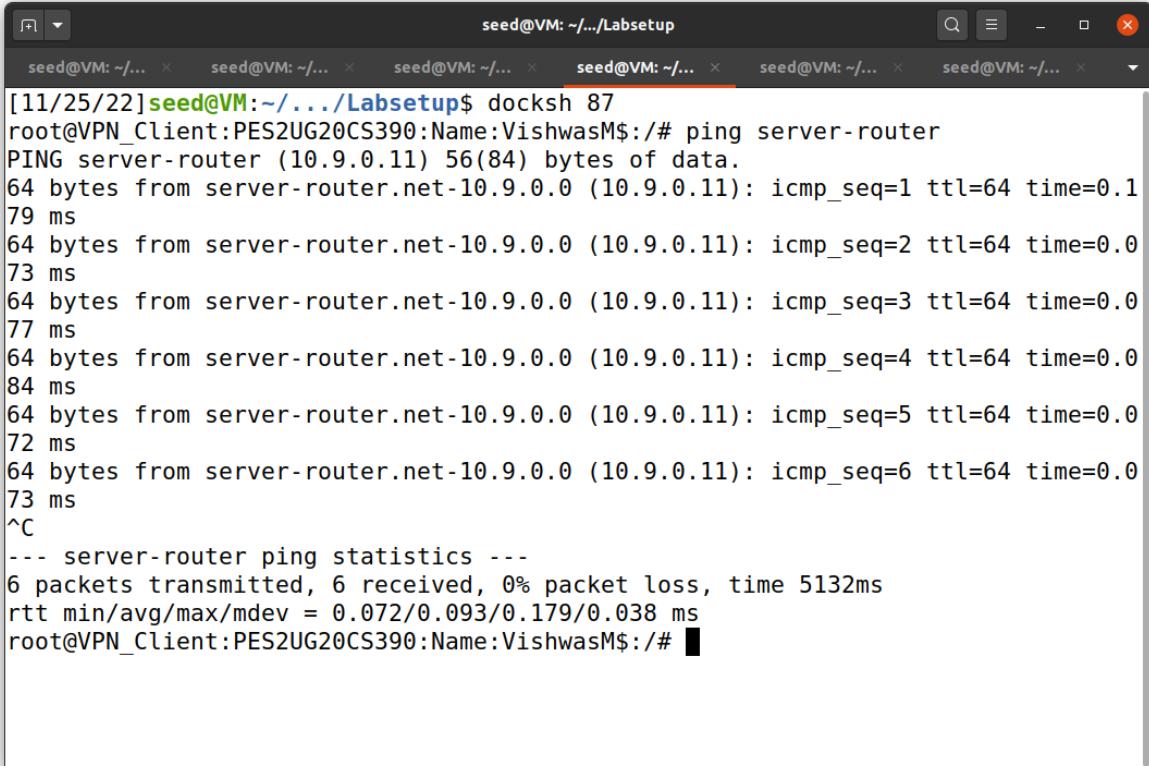
SRN: PES2UG20CS390

SEC: F

DATE:26/11/2022

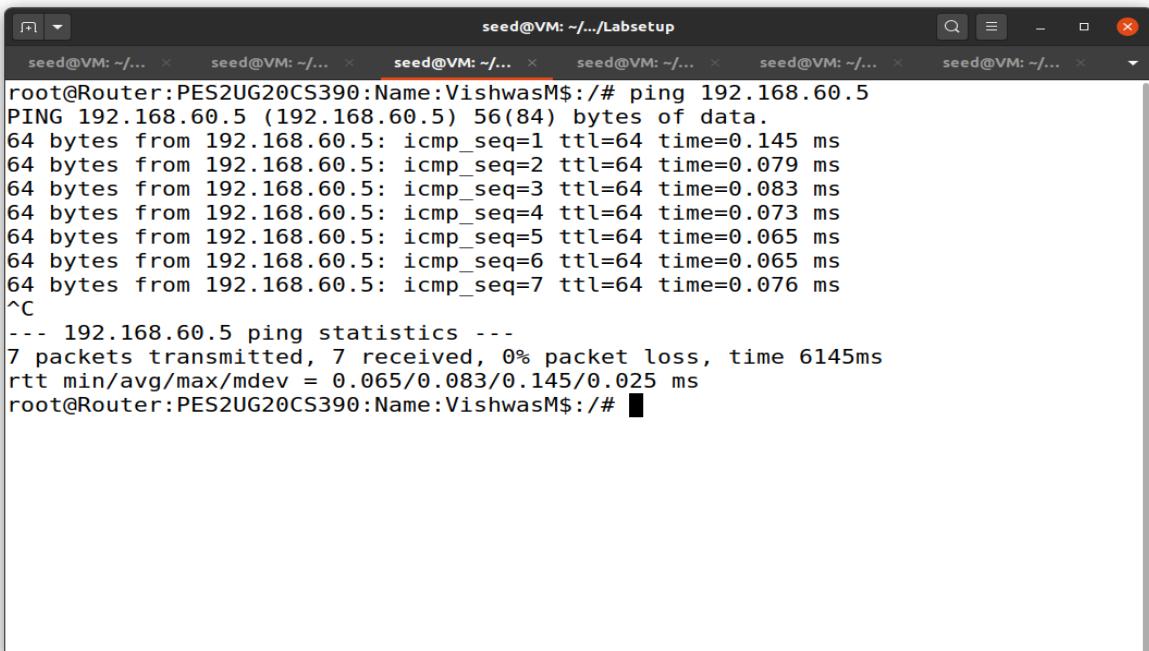
Task1: Network Security

Host U - 10.9.0.5 can communicate with VPN Server (server-router)



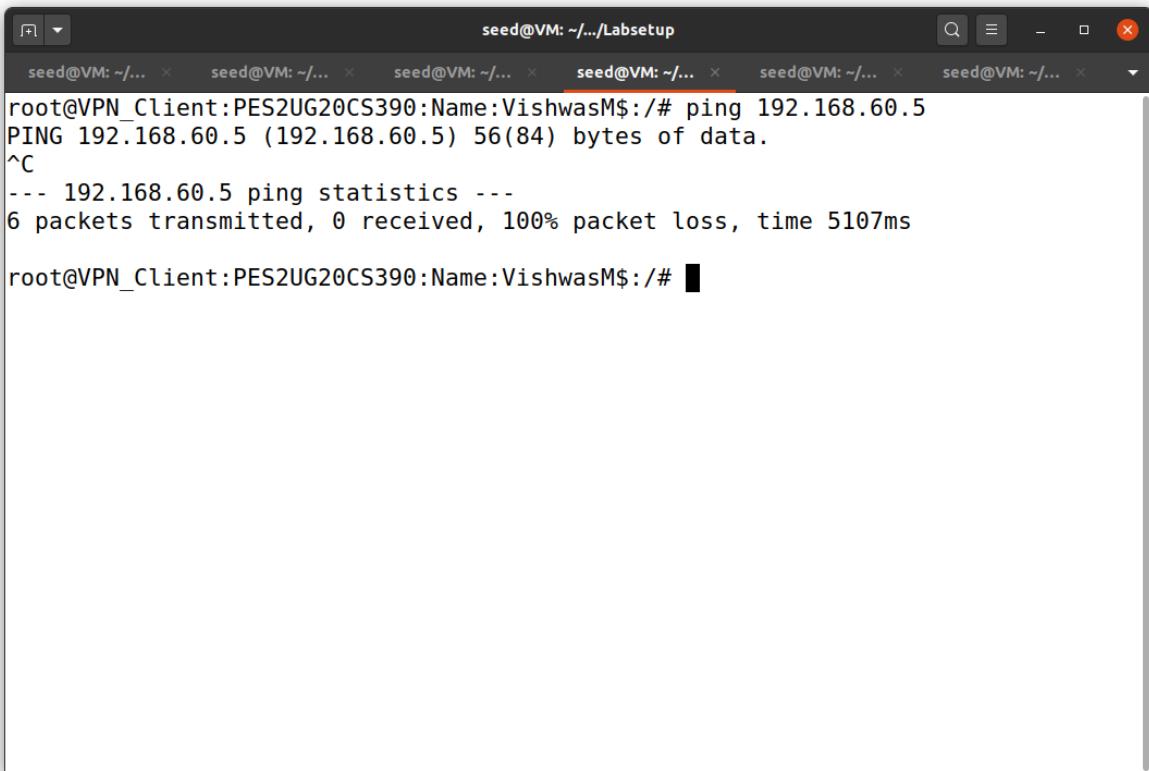
```
seed@VM: ~/.../Labsetup
[11/25/22]seed@VM:~/.../Labsetup$ docksh 87
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/# ping server-router
PING server-router (10.9.0.11) 56(84) bytes of data.
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=1 ttl=64 time=0.1
79 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=2 ttl=64 time=0.0
73 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=3 ttl=64 time=0.0
77 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=4 ttl=64 time=0.0
84 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=5 ttl=64 time=0.0
72 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=6 ttl=64 time=0.0
73 ms
^C
--- server-router ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5132ms
rtt min/avg/max/mdev = 0.072/0.093/0.179/0.038 ms
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/#
```

VPN Server (server-router) can communicate with Host V (host-192.168.60.5)
On server-router



```
seed@VM: ~/.../Labsetup
root@Router:PES2UG20CS390:Name:VishwasM$:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=64 time=0.145 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=64 time=0.079 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=64 time=0.083 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=64 time=0.073 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=64 time=0.065 ms
64 bytes from 192.168.60.5: icmp_seq=6 ttl=64 time=0.065 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=64 time=0.076 ms
^C
--- 192.168.60.5 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6145ms
rtt min/avg/max/mdev = 0.065/0.083/0.145/0.025 ms
root@Router:PES2UG20CS390:Name:VishwasM$:/#
```

Host U (Client - 10.9.0.5) should not be able to communicate with Host V (host 192.168.60.5)
On Client 10.9.0.5



The screenshot shows a terminal window titled "seed@VM: ~/.../Labsetup". The terminal is running on a host system (VM) and has multiple tabs open, all showing the same prompt: "seed@VM: ~/.../Labsetup". The current tab displays the following command and its output:

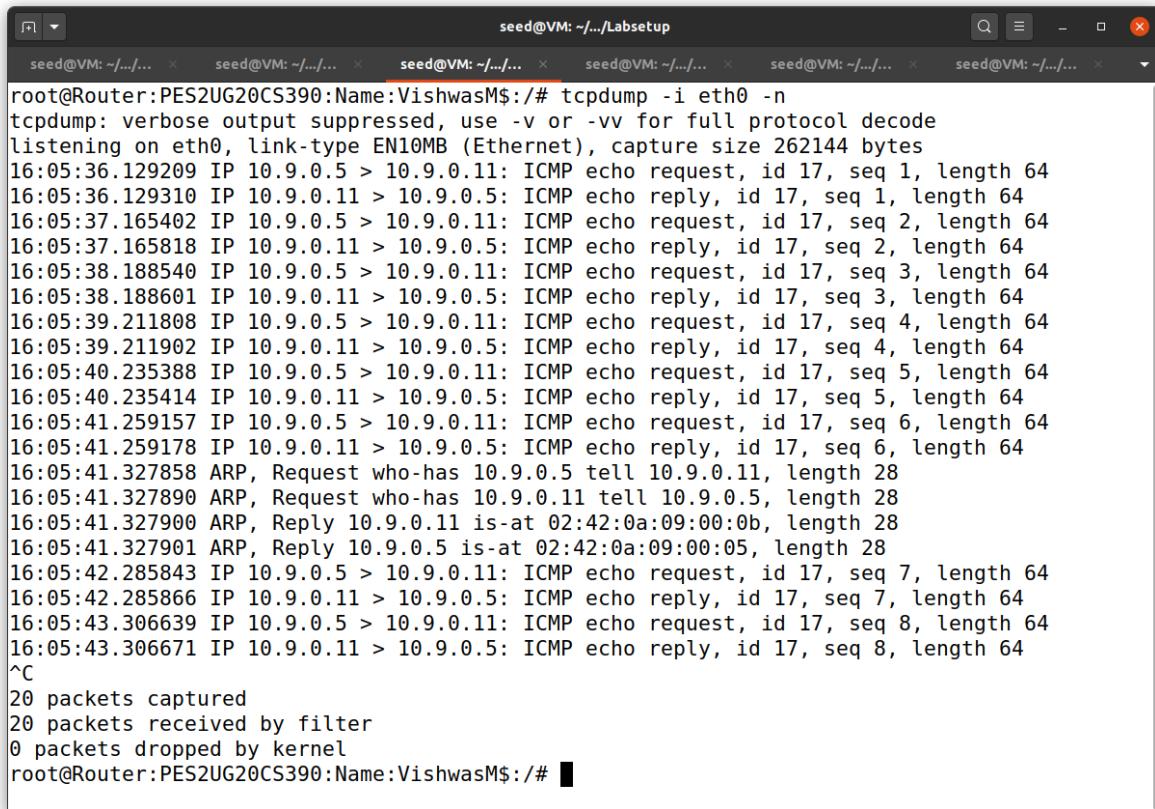
```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
^C
--- 192.168.60.5 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5107ms

root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/#
```

The output indicates that the ping command was interrupted (^C) and showed 100% packet loss.

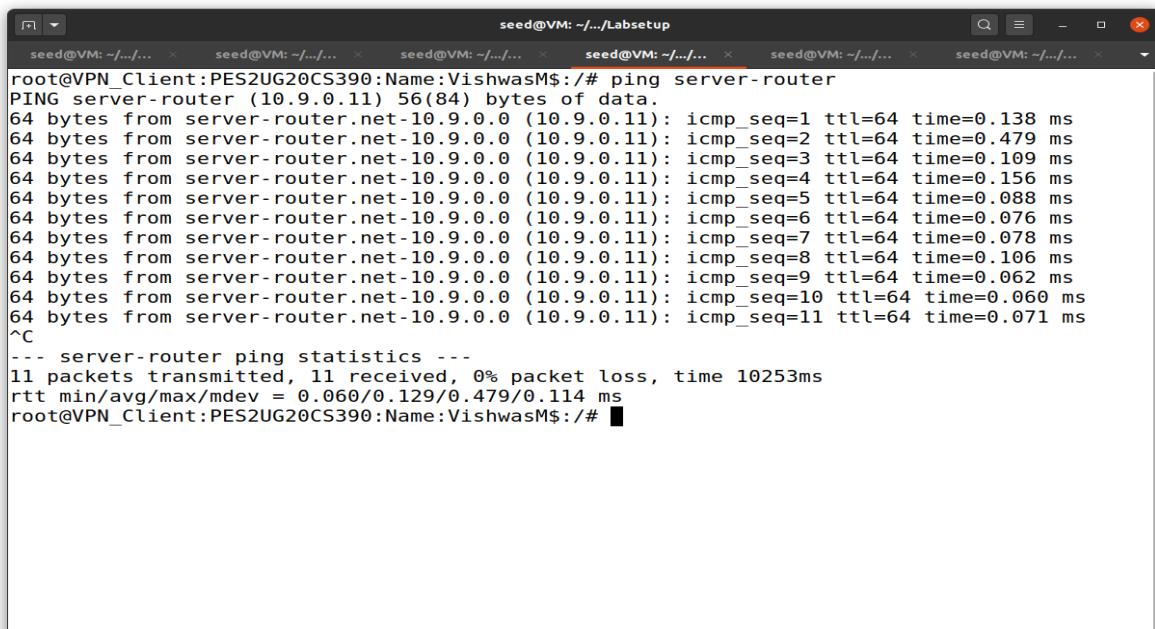
Run tcpdump on the router, and sniff the traffic on each of the networks. Show that you can capture packets.

On server-router run –



```
root@Router: PES2UG20CS390:Name:VishwasM$:# tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
16:05:36.129209 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 1, length 64
16:05:36.129310 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 1, length 64
16:05:37.165402 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 2, length 64
16:05:37.165818 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 2, length 64
16:05:38.188540 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 3, length 64
16:05:38.188601 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 3, length 64
16:05:39.211808 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 4, length 64
16:05:39.211902 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 4, length 64
16:05:40.235388 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 5, length 64
16:05:40.235414 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 5, length 64
16:05:41.259157 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 6, length 64
16:05:41.259178 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 6, length 64
16:05:41.327858 ARP, Request who-has 10.9.0.5 tell 10.9.0.11, length 28
16:05:41.327890 ARP, Request who-has 10.9.0.11 tell 10.9.0.5, length 28
16:05:41.327900 ARP, Reply 10.9.0.11 is-at 02:42:0a:09:00:0b, length 28
16:05:41.327901 ARP, Reply 10.9.0.5 is-at 02:42:0a:09:00:05, length 28
16:05:42.285843 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 7, length 64
16:05:42.285866 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 7, length 64
16:05:43.306639 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 8, length 64
16:05:43.306671 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 8, length 64
^C
20 packets captured
20 packets received by filter
0 packets dropped by kernel
root@Router: PES2UG20CS390:Name:VishwasM$:#
```

On Client - 10.9.0.5:



```
root@VPN_Client: PES2UG20CS390:Name:VishwasM$:# ping server-router
PING server-router (10.9.0.11) 56(84) bytes of data.
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=1 ttl=64 time=0.138 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=2 ttl=64 time=0.479 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=3 ttl=64 time=0.109 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=4 ttl=64 time=0.156 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=5 ttl=64 time=0.088 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=6 ttl=64 time=0.076 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=7 ttl=64 time=0.078 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=8 ttl=64 time=0.106 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=9 ttl=64 time=0.062 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=10 ttl=64 time=0.060 ms
64 bytes from server-router.net-10.9.0.0 (10.9.0.11): icmp_seq=11 ttl=64 time=0.071 ms
^C
--- server-router ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time 10253ms
rtt min/avg/max/mdev = 0.060/0.129/0.479/0.114 ms
root@VPN_Client: PES2UG20CS390:Name:VishwasM$:#
```

Task2: Create and Configure TUN Interface

Task 2.a: Name of the Interface

```
seed@VM: ~/.../Labsetup
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# chmod a+x tun.py
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ./tun.py &
[1] 25
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# # ip addrInterface Name: tun0
^C
```

```
seed@VM: ~/.../Labsetup
[2]+ Exit 127 ./tun0.py
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ./tun.py & ip addr
[2] 31
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
10: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: tun1
^C
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
2: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
3: tun1: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
10: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# jobs
```

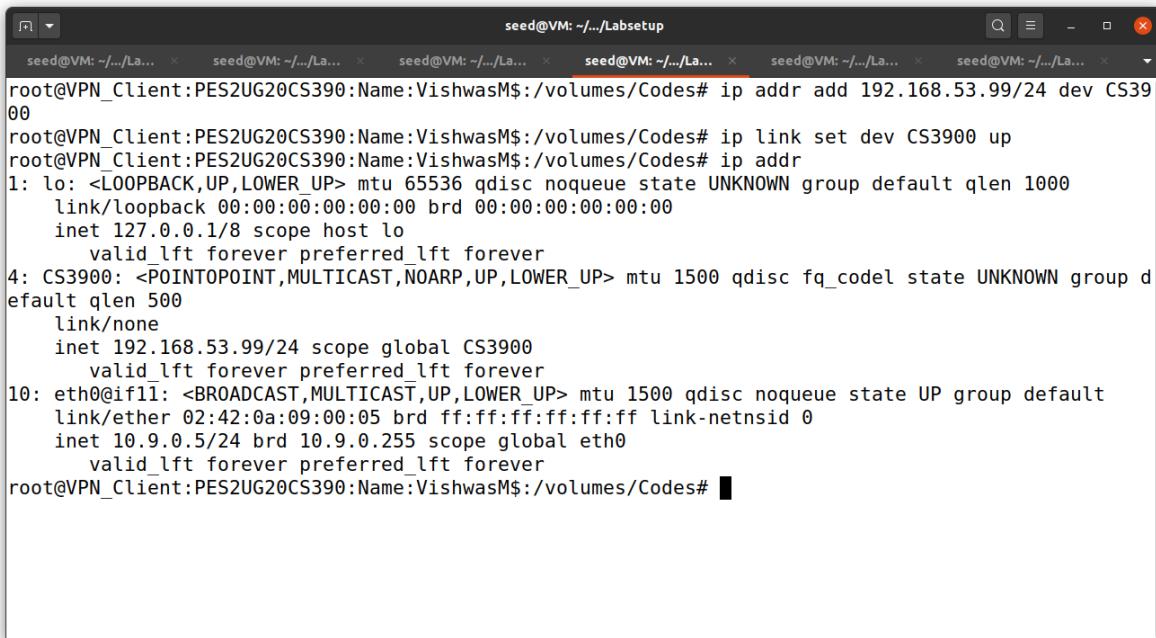
We use the cmd Jobs to check the TUN interface and we use kill command to kill the tunnel process.

```
seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs...
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: tun1
^C
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr
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
2: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
3: tun1: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
10: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# jobs
[1]-  Running                  ./tun.py &
[2]+  Running                  ./tun.py &
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# kill %2
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# jobs
[1]-  Running                  ./tun.py &
[2]+  Terminated                ./tun.py
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# kill %1
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# jobs
[1]+  Terminated                ./tun.py
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

Here I am changing the name of the TUN interface as ‘CS390’.

```
seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs... seed@VM: ~/.../Labs...
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# chmod a+x tun.py
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ./tun.py & ip addr
[1] 39
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
10: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: CS3900
^C
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr
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
4: CS3900: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default qlen 500
    link/none
10: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

Task 2.b: Set up the TUN interface



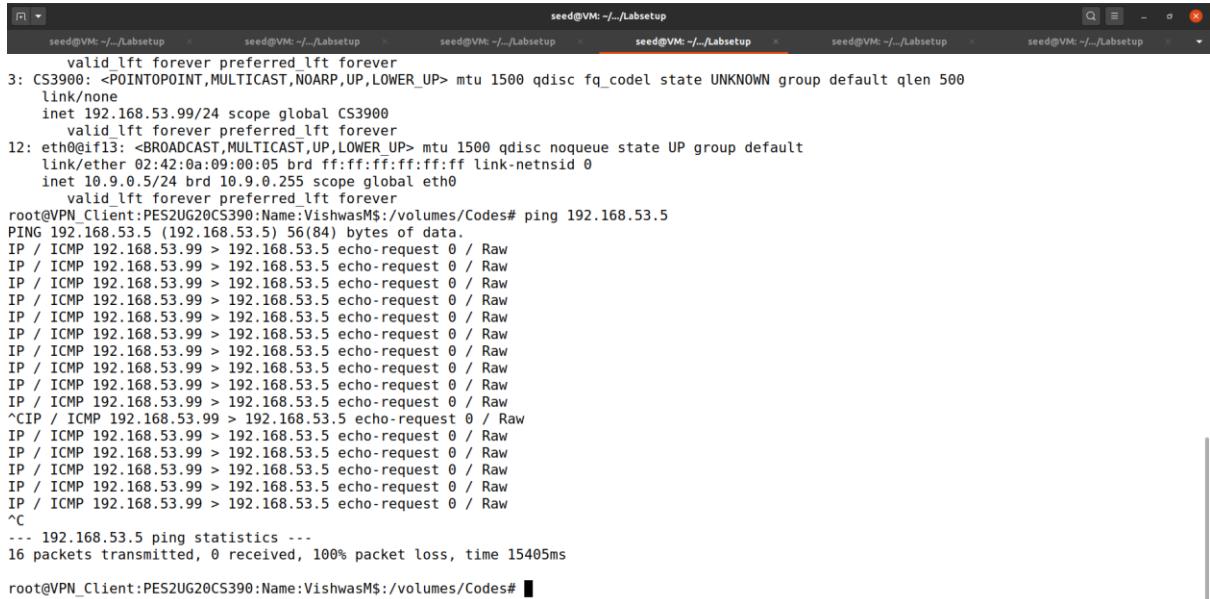
The screenshot shows a terminal window titled "seed@VM: ~/Labsetup" with multiple tabs open. The current tab displays the output of the command "ip addr add 192.168.53.99/24 dev CS3900". The output shows the creation of a new TUN interface (CS3900) with an IP address of 192.168.53.99/24. It also lists other interfaces like lo and eth0, and their respective configurations.

```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr add 192.168.53.99/24 dev CS3900
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip link set dev CS3900 up
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
4: CS3900: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 192.168.53.99/24 scope global CS3900
        valid_lft forever preferred_lft forever
10: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

First, we assigned an IP address to the TUN interface. Then we need to bring up the interface because it is still in down state.

Task 2.c: Read from the TUN interface

When we ping 192.168.53.5:



```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ping 192.168.53.5
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.
^C
--- 192.168.53.5 ping statistics ---
16 packets transmitted, 0 received, 100% packet loss, time 15405ms
```

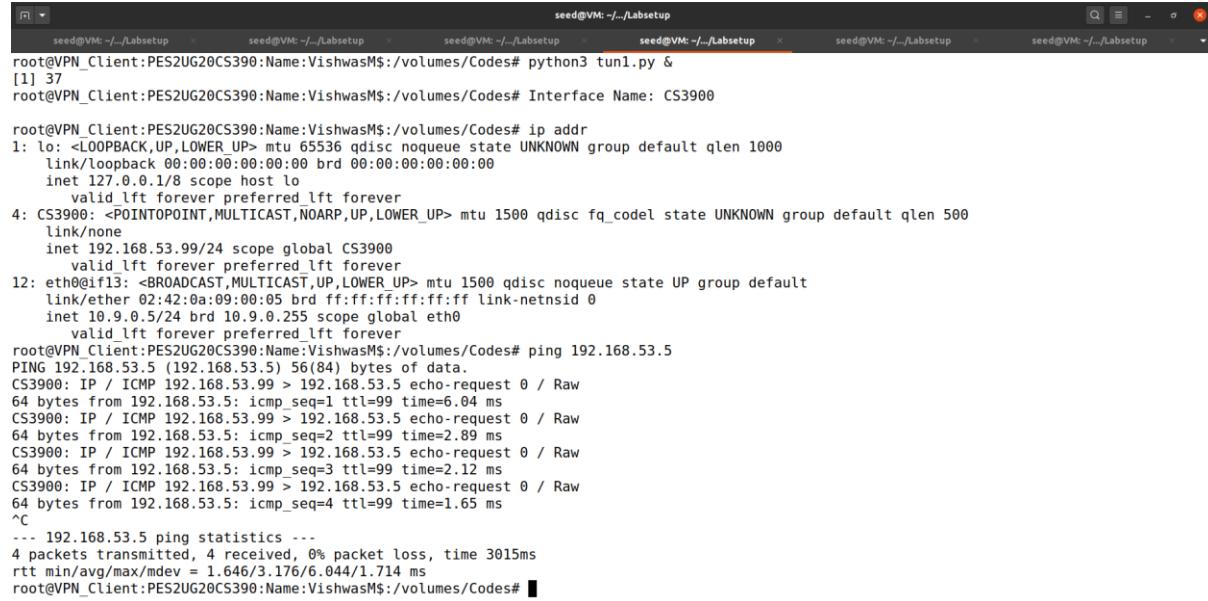
First, we had to change the code in tun.py. Then we had to run the updated python file. Then we are trying to read all the packets coming to TUN interface

When we ping 192.168.60.5:

```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
^C
--- 192.168.60.5 ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 6128ms
```

We don't get any output when we ping this ip addr because it is still not receiving any packets from the TUN interface.

Task 2.d: Write to the TUN interface

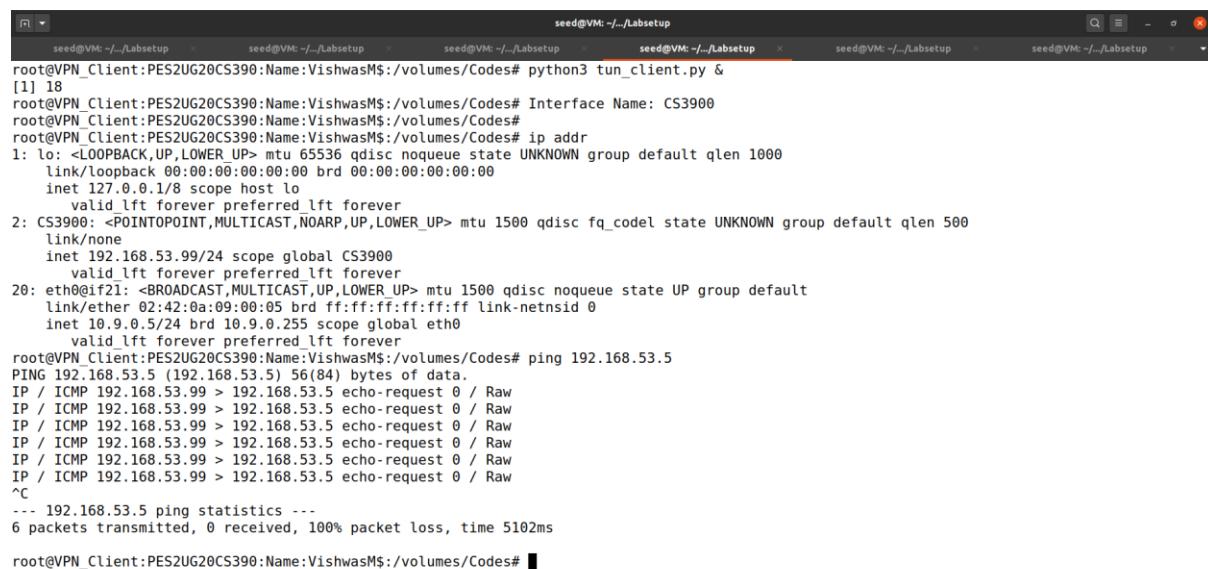


```
seed@VM: ~/Labsetup
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# python3 tun1.py &
[1] 37
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: CS3900

root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
4: CS3900: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 192.168.53.99/24 scope global CS3900
        valid_lft forever preferred_lft forever
12: eth0@if13: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
        inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
            valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ping 192.168.53.5
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.
CS3900: IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
64 bytes from 192.168.53.5: icmp_seq=1 ttl=99 time=6.04 ms
CS3900: IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
64 bytes from 192.168.53.5: icmp_seq=2 ttl=99 time=2.89 ms
CS3900: IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
64 bytes from 192.168.53.5: icmp_seq=3 ttl=99 time=2.12 ms
CS3900: IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
64 bytes from 192.168.53.5: icmp_seq=4 ttl=99 time=1.65 ms
^C
--- 192.168.53.5 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3015ms
rtt min/avg/max/mdev = 1.646/3.176/6.044/1.714 ms
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

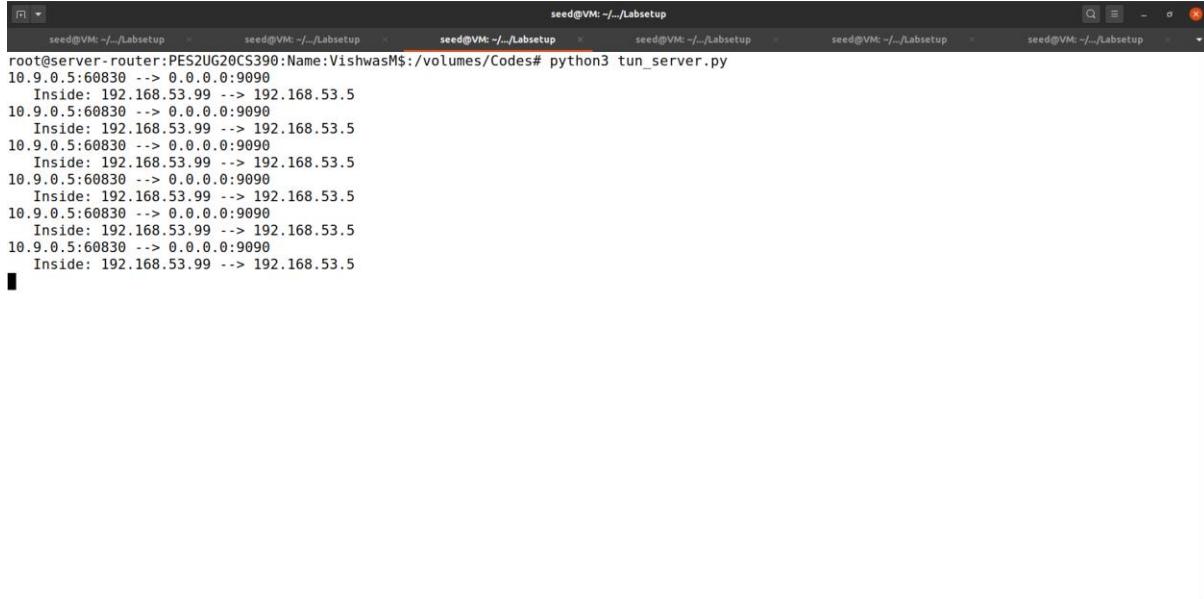
As we can see here both request and reply ICMP packets are seen which means that TUN interface is accepting as well as receiving the packets.

Task 3: Send the IP packet to VPN Server through a Tunnel



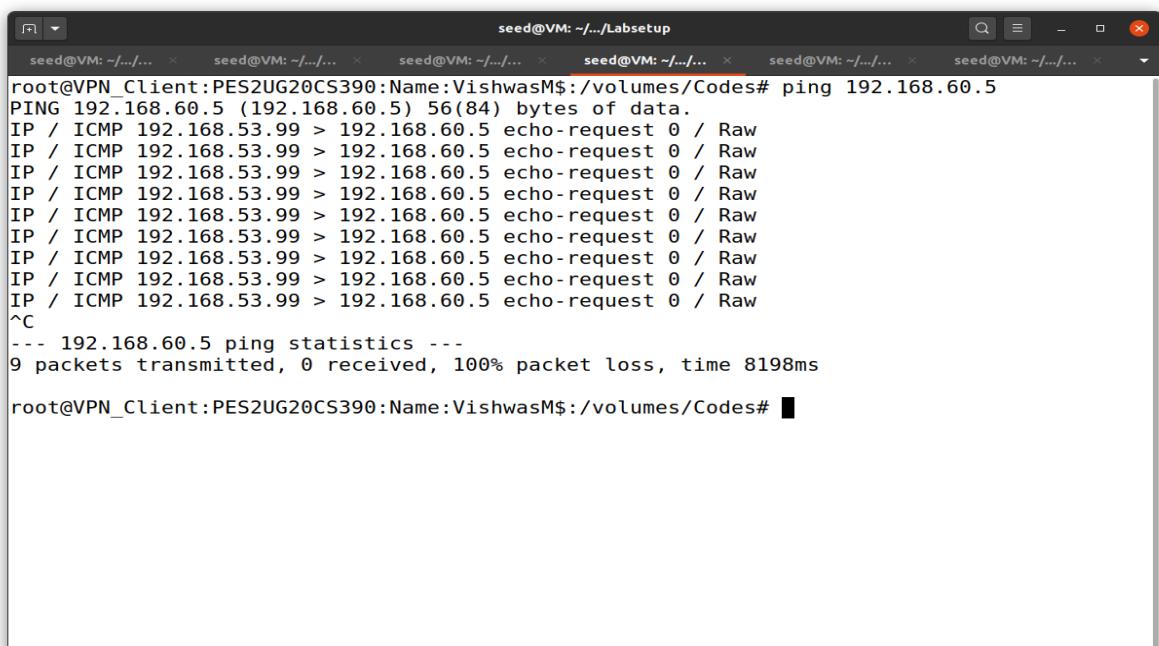
```
seed@VM: ~/Labsetup
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# python3 tun_client.py &
[1] 18
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: CS3900
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
2: CS3900: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 192.168.53.99/24 scope global CS3900
        valid_lft forever preferred_lft forever
20: eth0@if21: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
        inet 10.9.0.5/24 brd 10.9.0.255 scope global eth0
            valid_lft forever preferred_lft forever
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ping 192.168.53.5
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
^C
--- 192.168.53.5 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5102ms
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

We can see that the packets are 100% lost in the above screenshot.



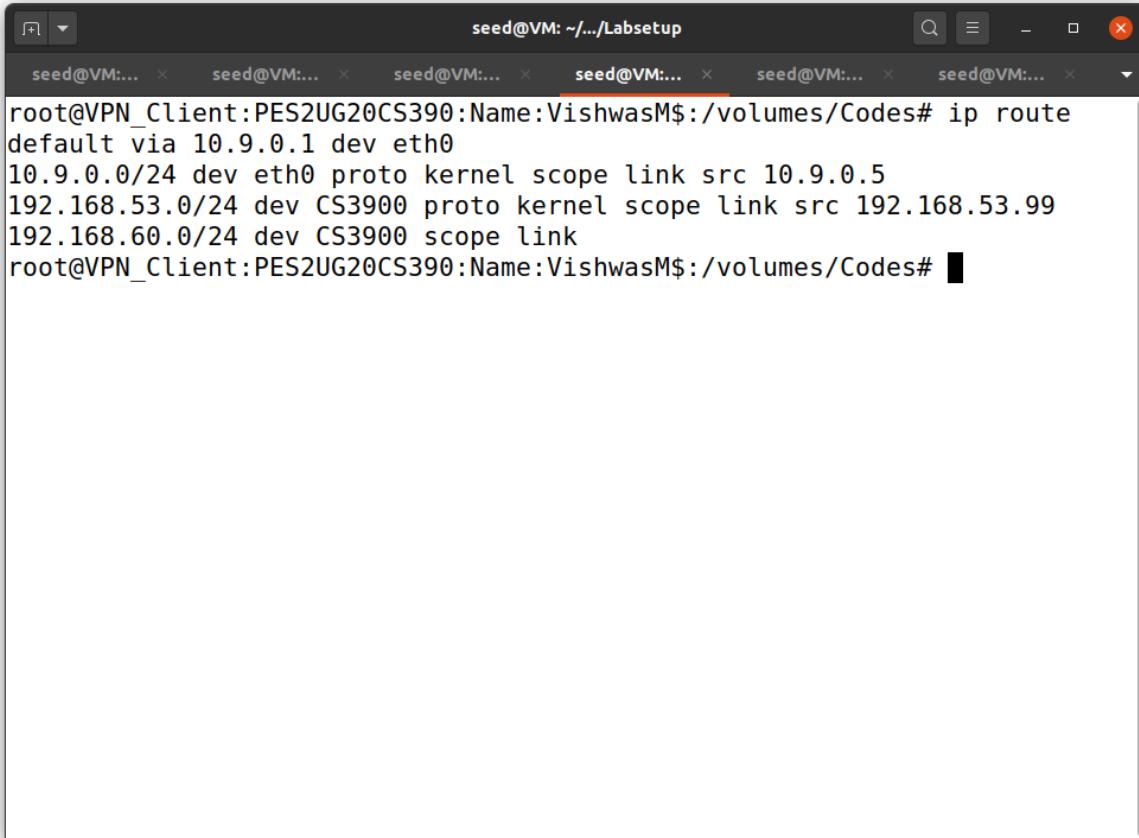
```
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# python3 tun_server.py
Inside: 192.168.53.99 --> 192.168.53.5
10.9.0.5:60830 --> 0.0.0.0:9090
Inside: 192.168.53.99 --> 192.168.53.5
```

In the server we can see how the packets are moving inside the tunnel as we can see in the above screenshot.



```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
^C
--- 192.168.60.5 ping statistics ---
9 packets transmitted, 0 received, 100% packet loss, time 8198ms
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

Here we are pinging to 192.168.60.5.



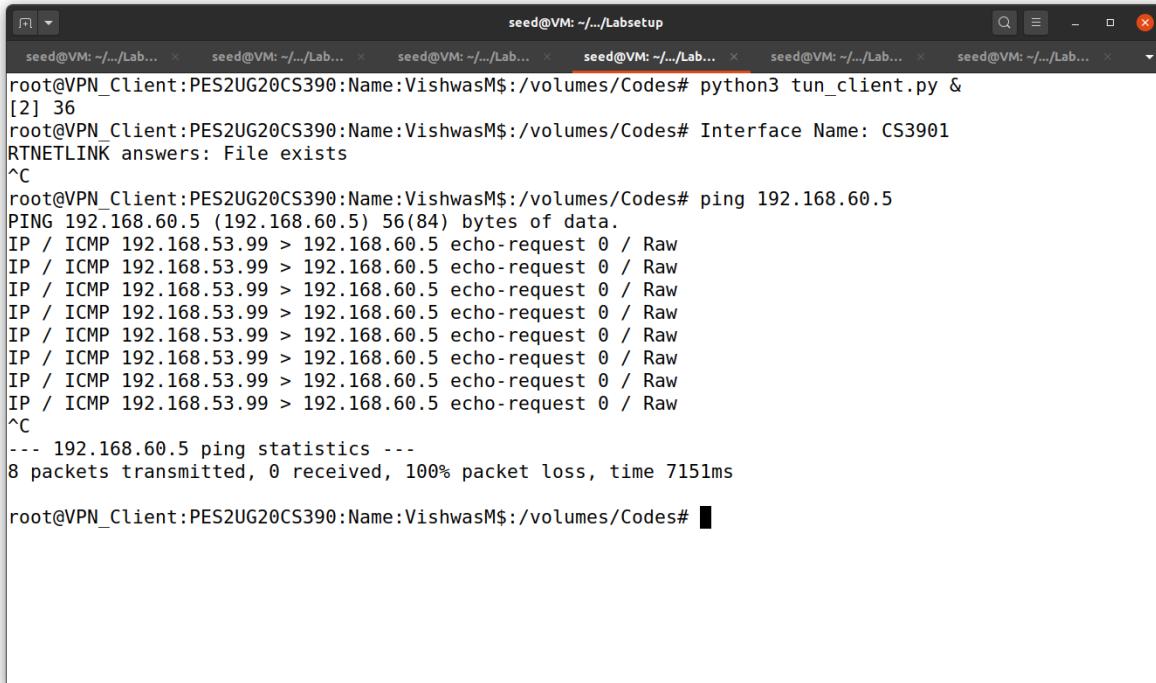
A screenshot of a terminal window titled "seed@VM: ~/Labsetup". The window contains several tabs, all labeled "seed@VM:...". The active tab shows the output of the command "ip route". The output is as follows:

```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ip route
default via 10.9.0.1 dev eth0
10.9.0.0/24 dev eth0 proto kernel scope link src 10.9.0.5
192.168.53.0/24 dev CS3900 proto kernel scope link src 192.168.53.99
192.168.60.0/24 dev CS3900 scope link
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

We can see the IP routing routing table with the help of the above command mentioned in the above screenshot.

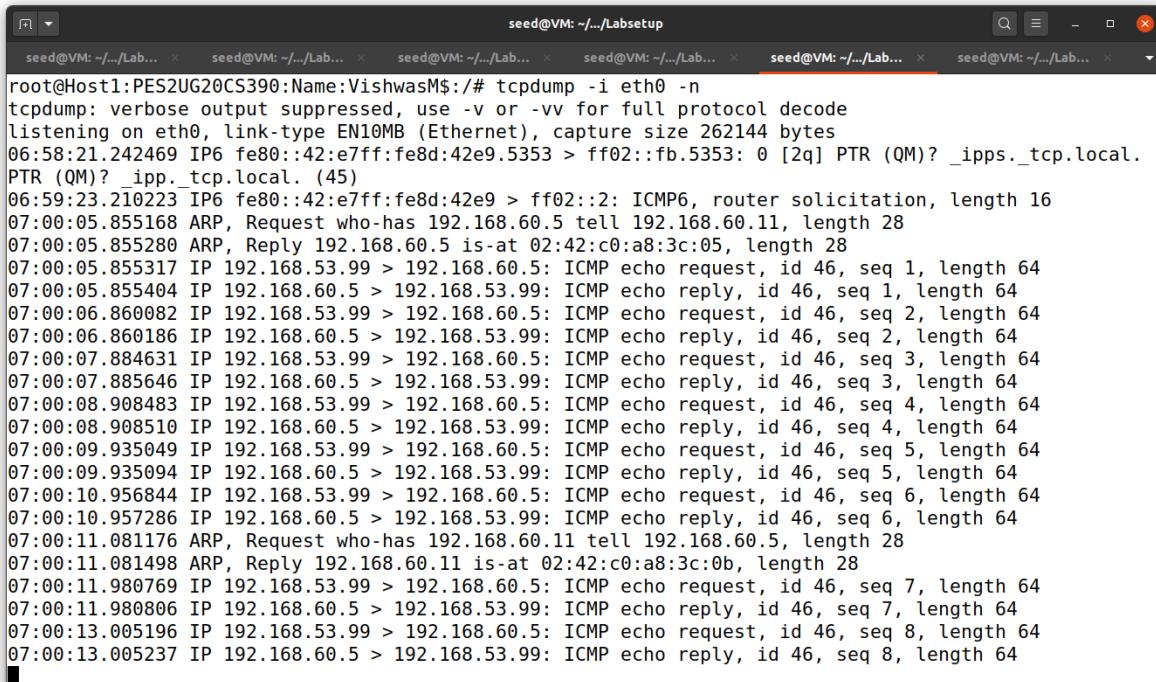
Task 4: Set up the VPN server

At Client:



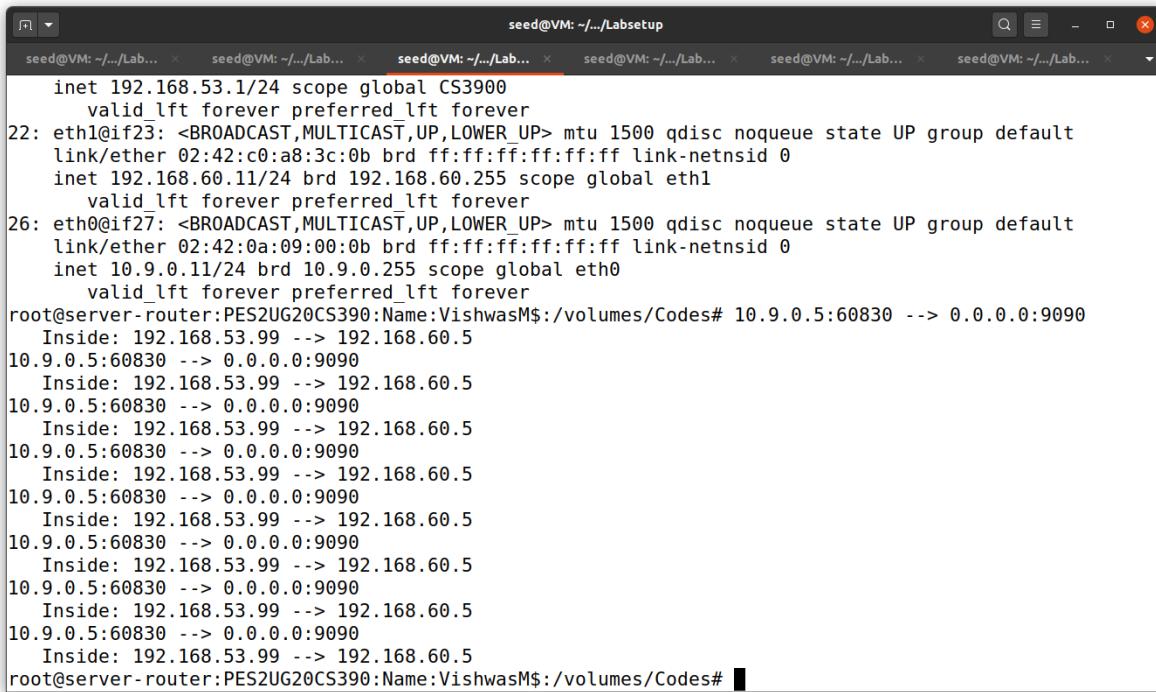
```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# python3 tun_client.py &
[2] 36
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: CS3901
RTNETLINK answers: File exists
^C
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.60.5 echo-request 0 / Raw
^C
--- 192.168.60.5 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7151ms
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

At Host1(192.168.60.5):



```
root@Host1:PES2UG20CS390:Name:VishwasM$:# tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:58:21.242469 IP6 fe80::42:e7ff:fe8d:42e9.5353 > ff02::fb.5353: 0 [2q] PTR (QM)? _ipps._tcp.local.
PTR (QM)? _ipp._tcp.local. (45)
06:59:23.210223 IP6 fe80::42:e7ff:fe8d:42e9 > ff02::2: ICMP6, router solicitation, length 16
07:00:05.855168 ARP, Request who-has 192.168.60.5 tell 192.168.60.11, length 28
07:00:05.855280 ARP, Reply 192.168.60.5 is-at 02:42:c0:a8:3c:05, length 28
07:00:05.855317 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 1, length 64
07:00:05.855404 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 1, length 64
07:00:06.860082 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 2, length 64
07:00:06.860186 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 2, length 64
07:00:07.884631 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 3, length 64
07:00:07.885646 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 3, length 64
07:00:08.908483 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 4, length 64
07:00:08.908510 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 4, length 64
07:00:09.935049 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 5, length 64
07:00:09.935094 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 5, length 64
07:00:10.956844 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 6, length 64
07:00:10.957286 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 6, length 64
07:00:11.081176 ARP, Request who-has 192.168.60.11 tell 192.168.60.5, length 28
07:00:11.081498 ARP, Reply 192.168.60.11 is-at 02:42:c0:a8:3c:0b, length 28
07:00:11.980769 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 7, length 64
07:00:11.980806 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 7, length 64
07:00:13.005196 IP 192.168.53.99 > 192.168.60.5: ICMP echo request, id 46, seq 8, length 64
07:00:13.005237 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, id 46, seq 8, length 64
```

At Serve-Router:



The screenshot shows a terminal window titled "seed@VM: ~/.../Labsetup" with multiple tabs. The main pane displays the following output:

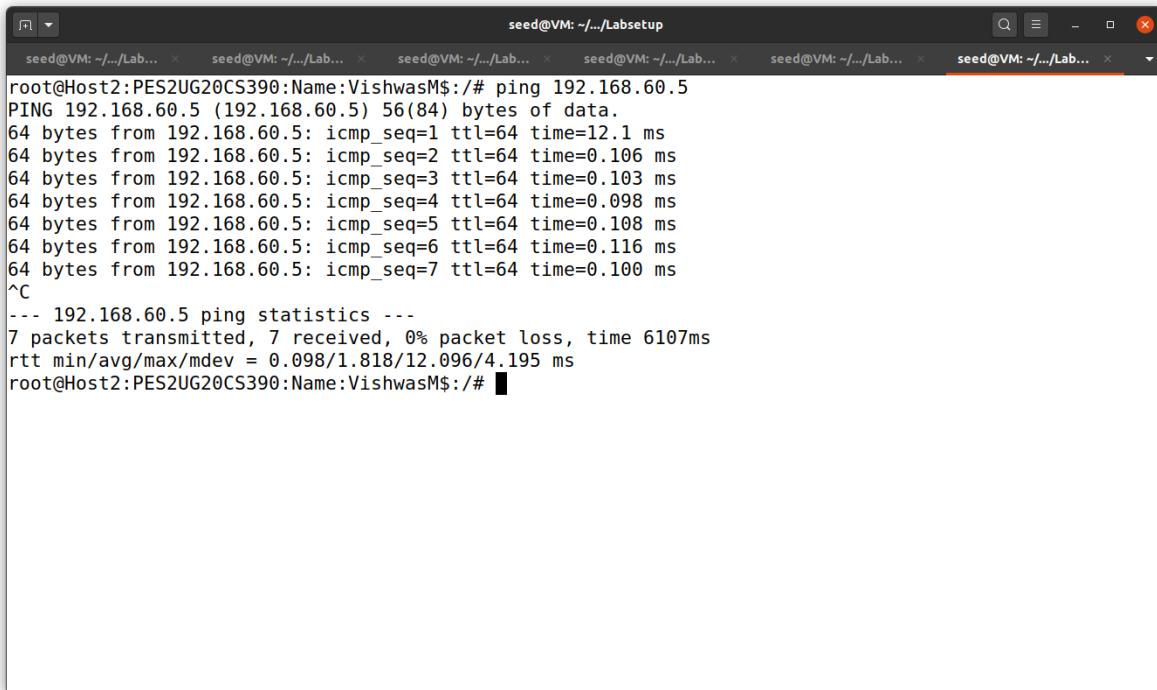
```
inet 192.168.53.1/24 scope global CS3900
    valid_lft forever preferred_lft forever
22: eth1@if23: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:a8:3c:0b brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.168.60.11/24 brd 192.168.60.255 scope global eth1
        valid_lft forever preferred_lft forever
26: eth0@if27: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:09:00:0b brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.9.0.11/24 brd 10.9.0.255 scope global eth0
        valid_lft forever preferred_lft forever
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# 10.9.0.5:60830 --> 0.0.0.0:9090
  Inside: 192.168.53.99 --> 192.168.60.5
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes#
```

Task 5: Handling Traffic in Both Directions

In server-router:

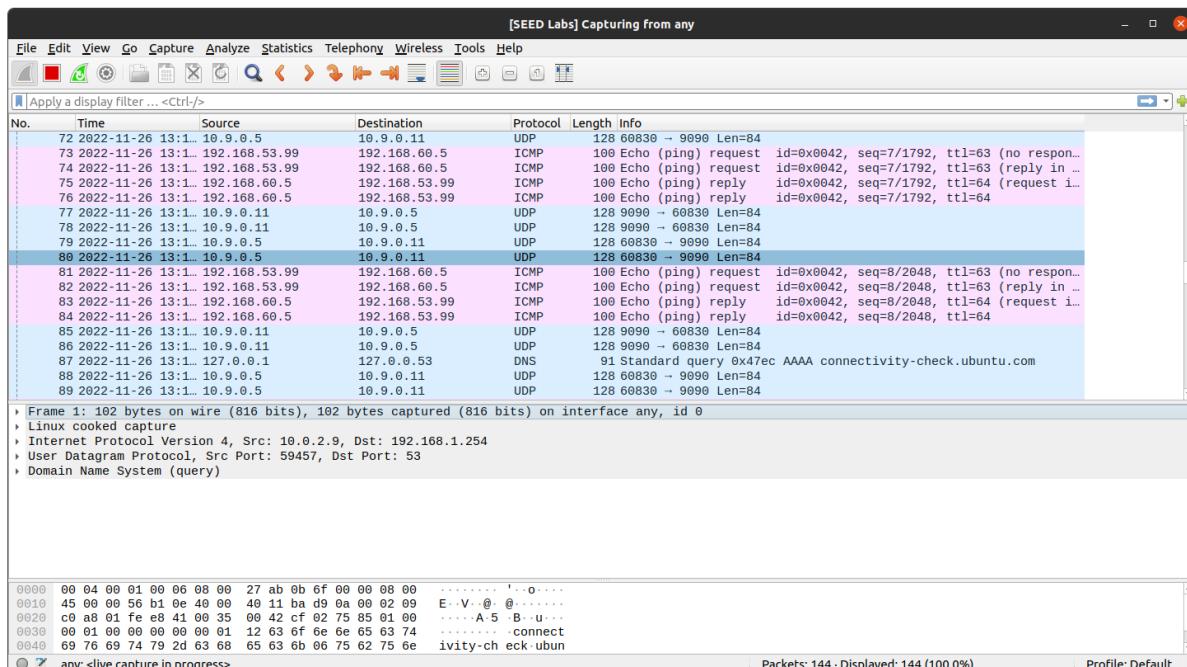
In VPN Client:

In another VPN_Client server:



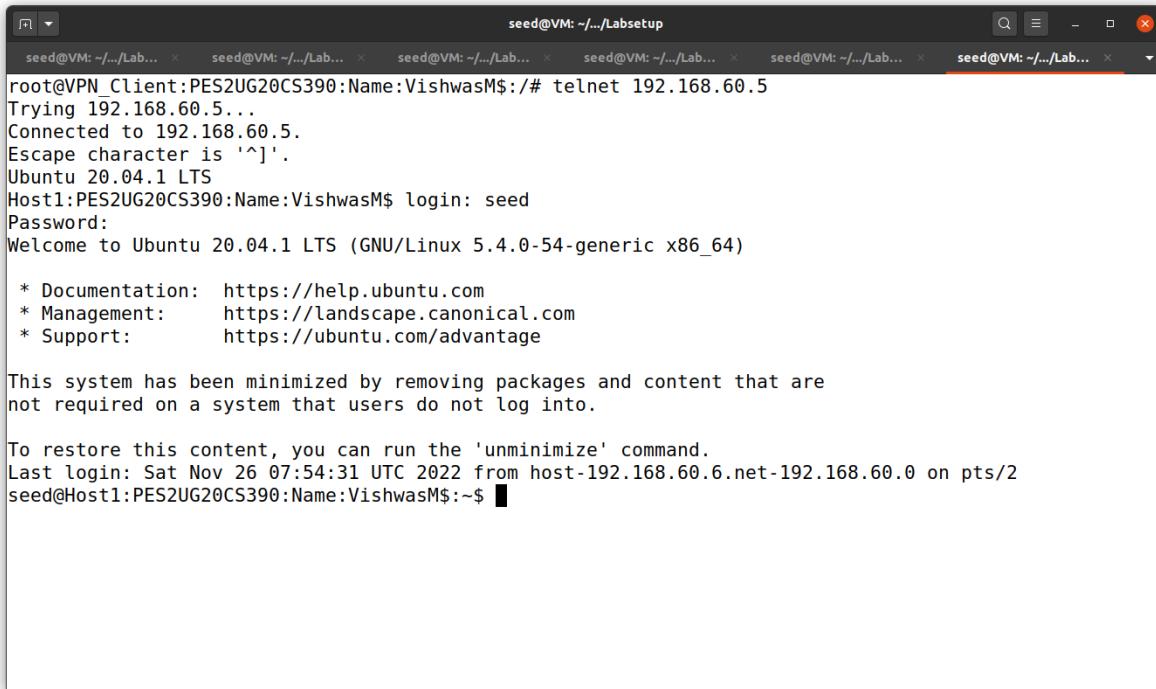
```
root@Host2:PES2UG20CS390:Name:VishwasM$:# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=64 time=12.1 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=64 time=0.106 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=64 time=0.103 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=64 time=0.098 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=64 time=0.108 ms
64 bytes from 192.168.60.5: icmp_seq=6 ttl=64 time=0.116 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=64 time=0.100 ms
^C
--- 192.168.60.5 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6107ms
rtt min/avg/max/mdev = 0.098/1.818/12.096/4.195 ms
root@Host2:PES2UG20CS390:Name:VishwasM$:/#
```

Wireshark capture:



When we do a telnet connection:

In another VPN_Client where we establish telnet connection:



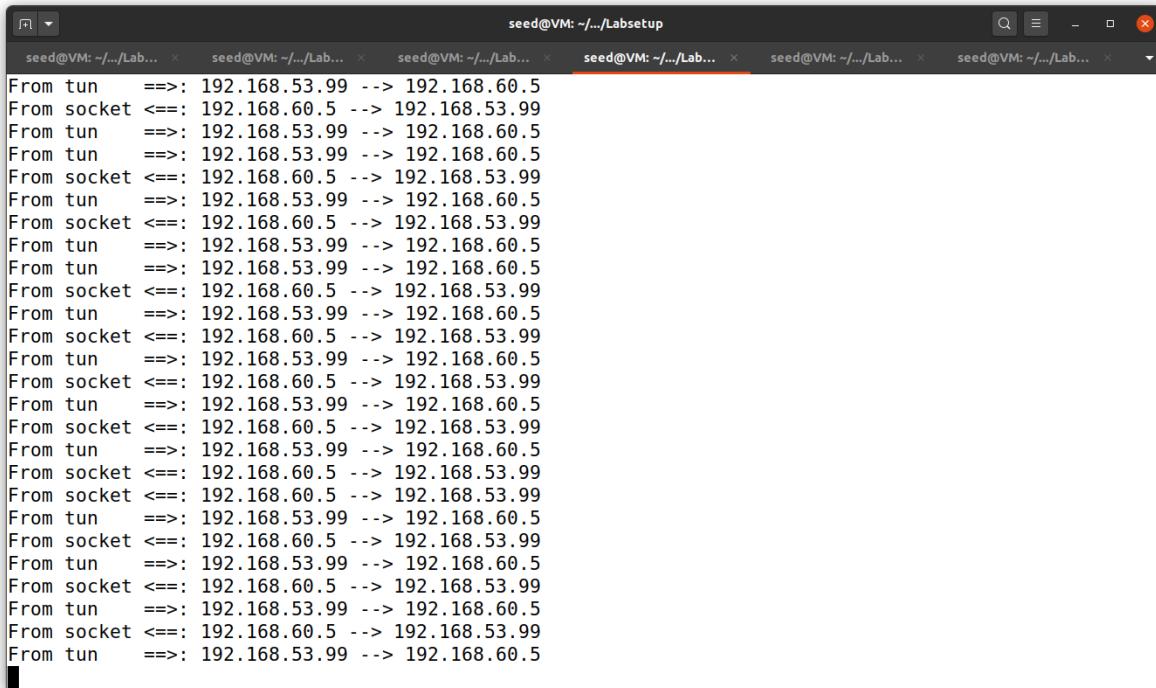
```
seed@VM: ~/.../Lab... root@VPN_Client:PES2UG20CS390:Name:VishwasM$:# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
Host1:PES2UG20CS390:Name:VishwasM$ login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Sat Nov 26 07:54:31 UTC 2022 from host-192.168.60.6.net-192.168.60.0 on pts/2
seed@Host1:PES2UG20CS390:Name:VishwasM$:
```

VPN_Client:

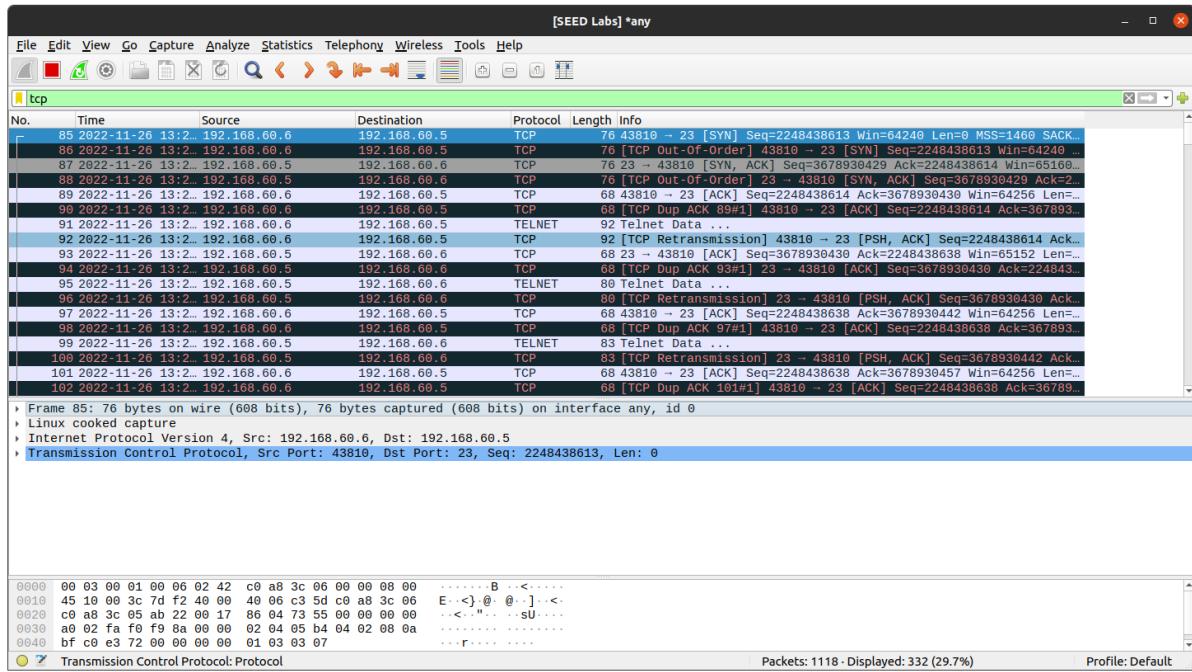


```
seed@VM: ~/.../Lab... seed@VM: ~/.../Lab... seed@VM: ~/.../Lab... seed@VM: ~/.../Lab... seed@VM: ~/.../Lab... seed@VM: ~/.../Lab...
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.53.99 --> 192.168.60.5
```

Server- router :

```
seed@VM: ~/.../Lab... x seed@VM: ~/.../Lab... x
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun    ==> 192.168.60.5 --> 192.168.53.99
```

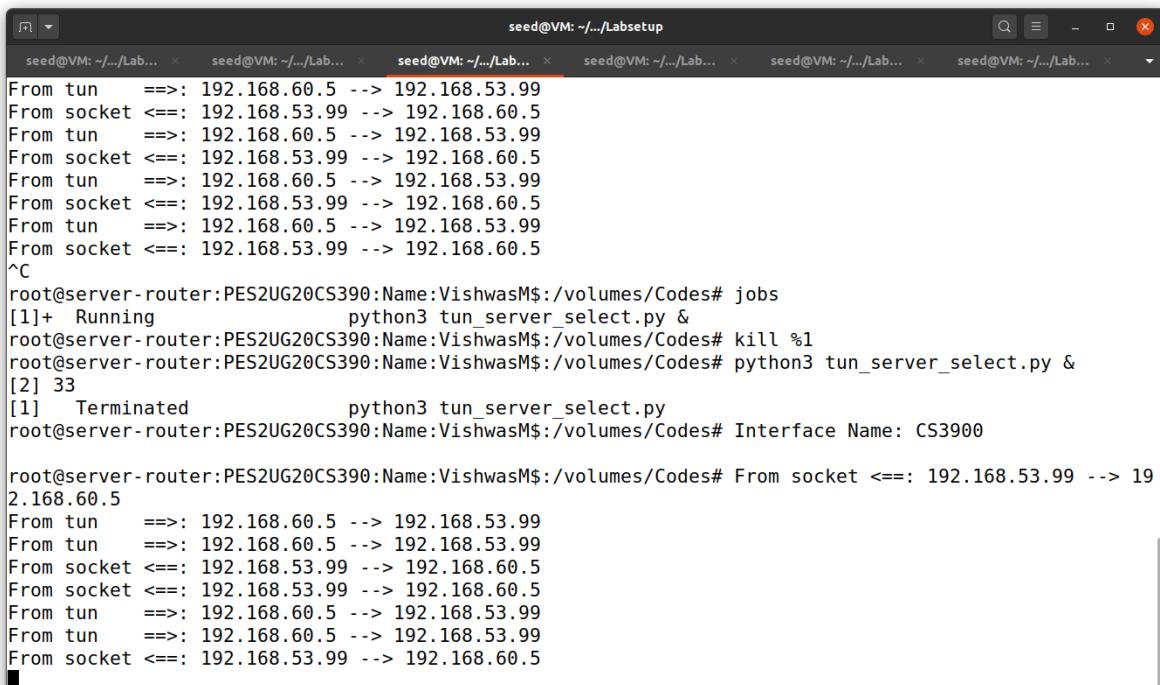
Wireshark Capture:



Task 6: Tunnel Breaking Experiment

Here we are breaking the Tunnel connection by pressing ctrl+C. When we try and press something in the client it will work as the screen gets stuck.

When we connect the tunnel again the word that we typed in terminal gets printed on the terminal which indicates that the tunnel is back and the connection is successful.

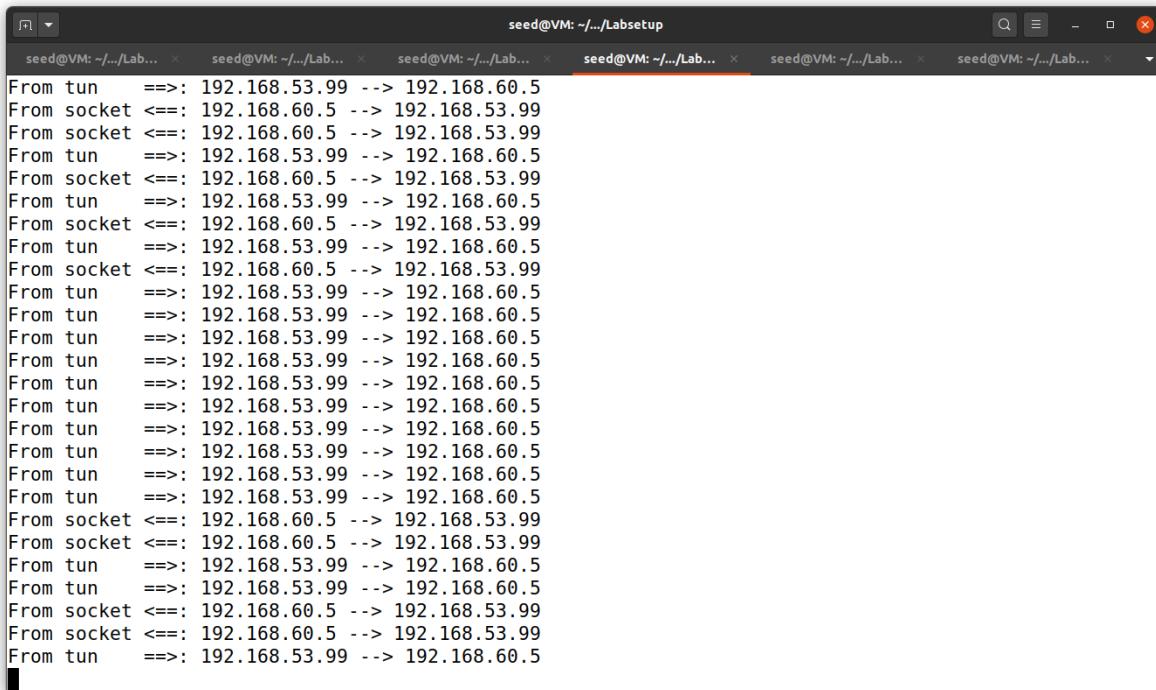


The screenshot shows a terminal window with multiple tabs, all titled 'seed@VM: ~.../Lab...'. The terminal displays network traffic logs and command-line interactions. At the top, there are several 'From tun' and 'From socket' entries showing bidirectional connections between 192.168.60.5 and 192.168.53.99. In the middle, a user types '^C' to interrupt a process. Following this, commands like 'jobs', 'kill %1', and 'python3 tun_server_select.py &' are run. The terminal then shows a message 'Interface Name: CS3900'. Finally, more 'From tun' and 'From socket' logs appear, indicating the connection has been restored.

```
From tun ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
^C
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# jobs
[1]+  Running                  python3 tun_server_select.py &
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# kill %1
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# python3 tun_server_select.py &
[2] 33
[1]  Terminated                python3 tun_server_select.py
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# Interface Name: CS3900

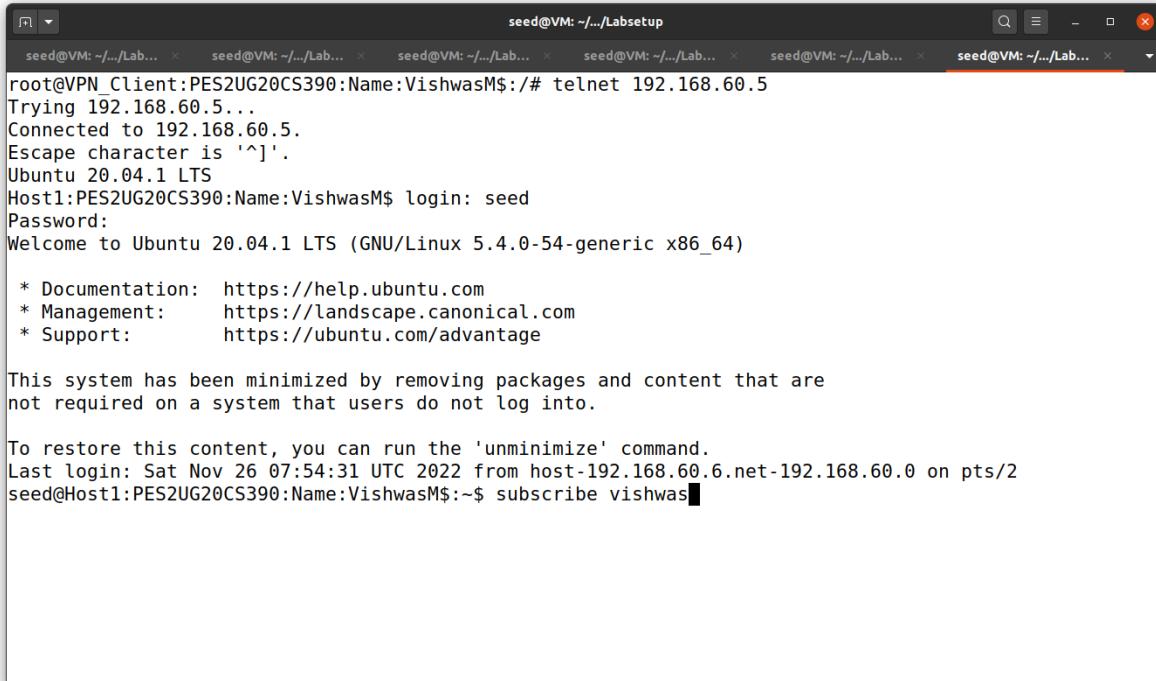
root@server-router:PES2UG20CS390:Name:VishwasM$:/volumes/Codes# From socket <==: 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun ==> 192.168.60.5 --> 192.168.53.99
From tun ==> 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.53.99 --> 192.168.60.5
```

VPN_Client:



```
From tun  ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun  ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun  ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun  ==> 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.53.99 --> 192.168.60.5
From tun  ==> 192.168.60.5 --> 192.168.53.99
From tun  ==> 192.168.53.99 --> 192.168.60.5
```

Server-Router:



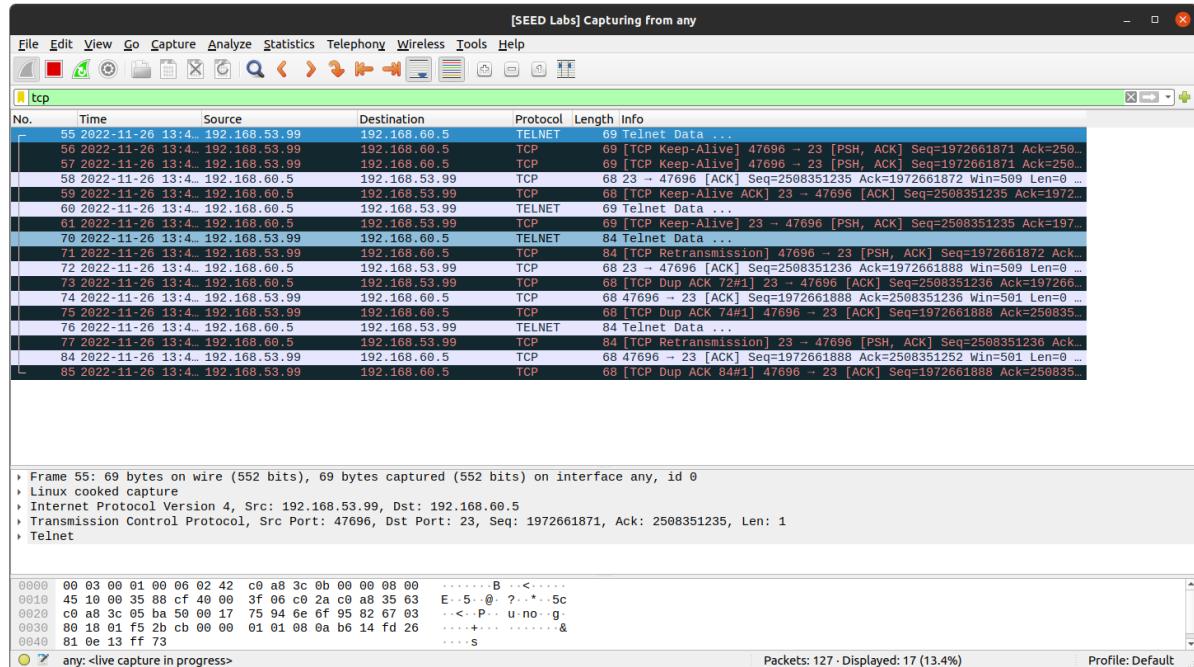
```
root@VPN_Client:PES2UG20CS390:Name:VishwasM$:# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^].
Ubuntu 20.04.1 LTS
Host1:PES2UG20CS390:Name:VishwasM$ login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

This system has been minimized by removing packages and content that are
not required on a system that users do not log into.

To restore this content, you can run the 'unminimize' command.
Last login: Sat Nov 26 07:54:31 UTC 2022 from host-192.168.60.6.net-192.168.60.0 on pts/2
seed@Host1:PES2UG20CS390:Name:VishwasM$~$ subscribe vishwas
```

Wireshark Capture:



Here we can observe that the connection which we just broke and connected in clearly depicted here.