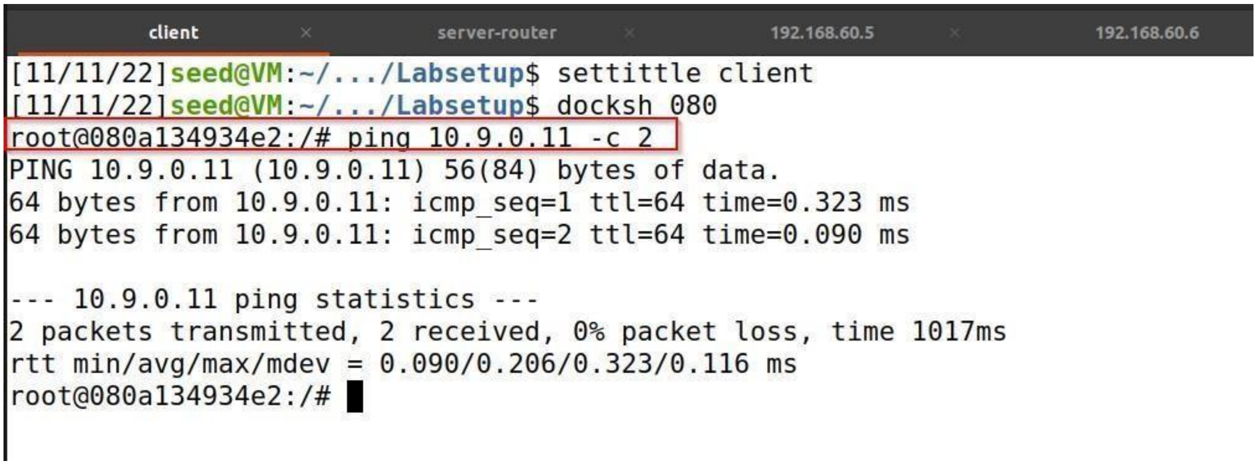
Lab – 8 Report

Name: Nithin Kumar Reddy Chelekam

Student Id: 700752333

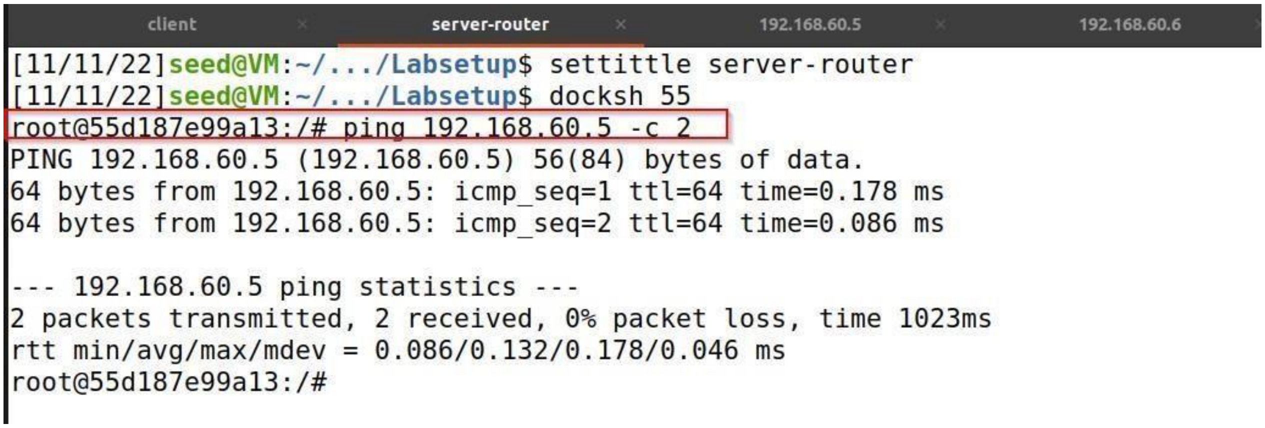
Task 1: Network Setup

Host U can communicate with VPN Server



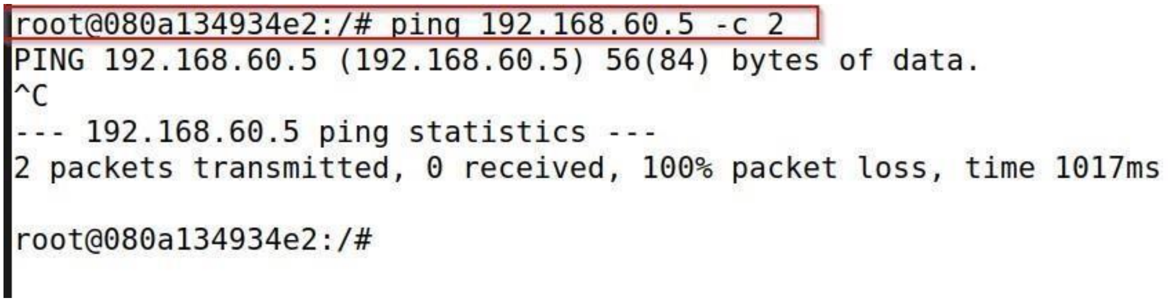
```
[11/11/22]seed@VM:~/.../Labsetup$ settittle client
[11/11/22]seed@VM:~/.../Labsetup$ docksh 080
root@080a134934e2:/# ping 10.9.0.11 -c 2
PING 10.9.0.11 (10.9.0.11) 56(84) bytes of data.
64 bytes from 10.9.0.11: icmp_seq=1 ttl=64 time=0.323 ms
64 bytes from 10.9.0.11: icmp_seq=2 ttl=64 time=0.090 ms

--- 10.9.0.11 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1017ms
rtt min/avg/max/mdev = 0.090/0.206/0.323/0.116 ms
root@080a134934e2:/#
```

VPN Server can communicate with Host V.



```
[11/11/22]seed@VM:~/.../Labsetup$ settittle server-router
[11/11/22]seed@VM:~/.../Labsetup$ docksh 55
root@55d187e99a13:/# ping 192.168.60.5 -c 2
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=0.178 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=64 time=0.086 ms

--- 192.168.60.5 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1023ms
rtt min/avg/max/mdev = 0.086/0.132/0.178/0.046 ms
root@55d187e99a13:/#
```

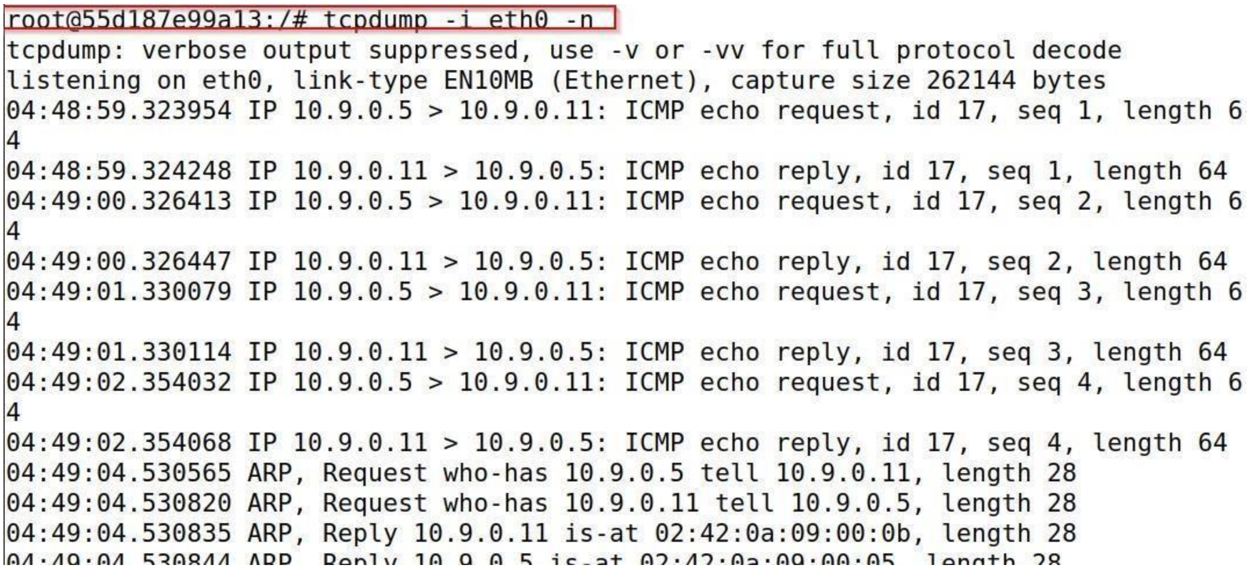
Host U should not be able to communicate with Host V



```
root@080a134934e2:/# ping 192.168.60.5 -c 2
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
^C
--- 192.168.60.5 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1017ms

root@080a134934e2:/#
```

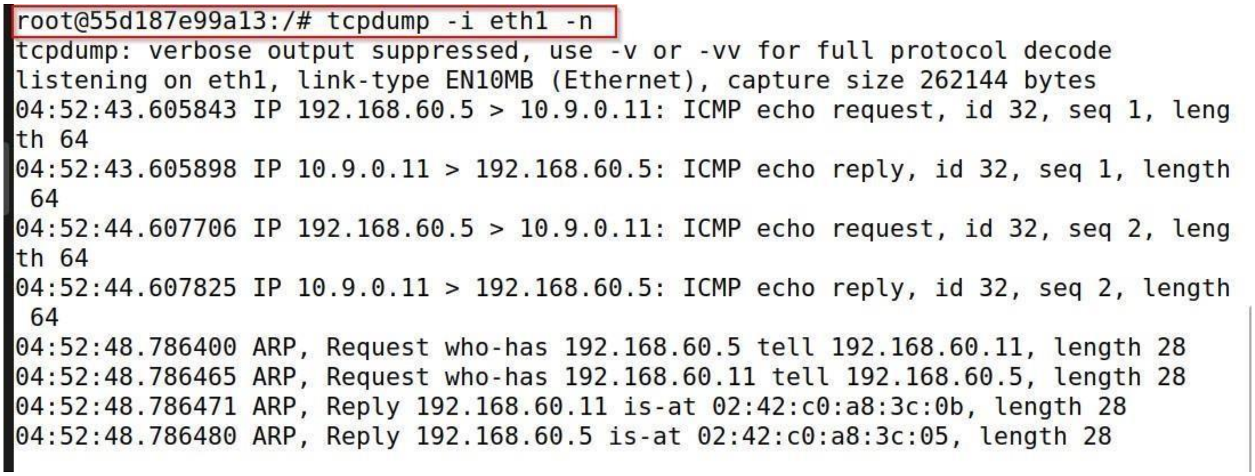
Run tcpdump on the router and sniff the traffic on each of the network. Show that you can capture packets.

1)Captured the traffic from the client to router



```
root@55d187e99a13:/# 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
04:48:59.323954 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 1, length 6
4
04:48:59.324248 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 1, length 64
04:49:00.326413 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 2, length 6
4
04:49:00.326447 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 2, length 64
04:49:01.330079 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 3, length 6
4
04:49:01.330114 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 3, length 64
04:49:02.354032 IP 10.9.0.5 > 10.9.0.11: ICMP echo request, id 17, seq 4, length 6
4
04:49:02.354068 IP 10.9.0.11 > 10.9.0.5: ICMP echo reply, id 17, seq 4, length 64
04:49:04.530565 ARP, Request who-has 10.9.0.5 tell 10.9.0.11, length 28
04:49:04.530820 ARP, Request who-has 10.9.0.11 tell 10.9.0.5, length 28
04:49:04.530835 ARP, Reply 10.9.0.11 is-at 02:42:0a:09:00:0b, length 28
04:49:04.530844 ARP, Reply 10.9.0.5 is-at 02:42:0a:09:00:05, length 28
```

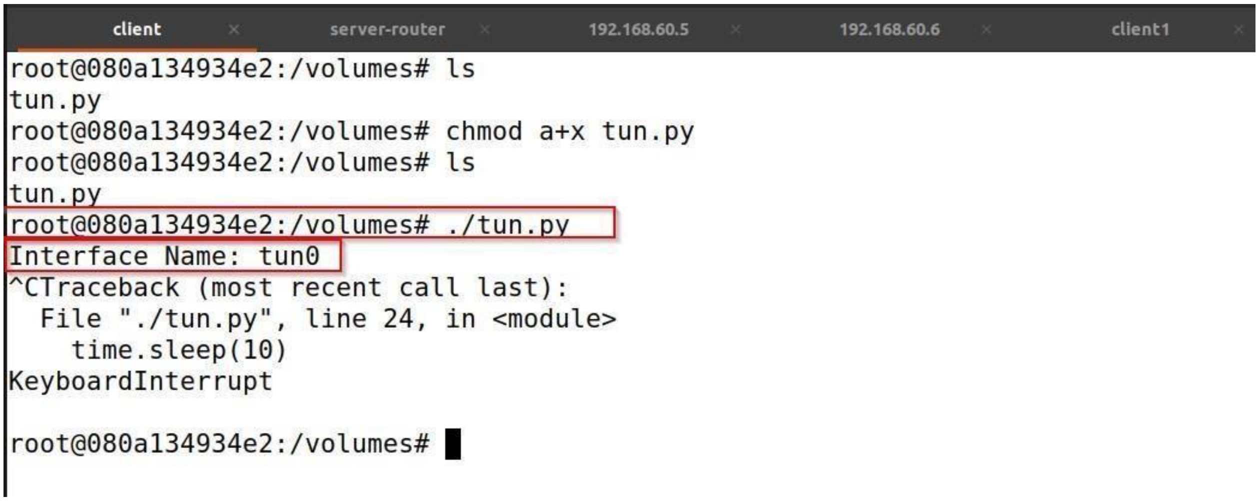
2) Captured the traffic from the host V to router



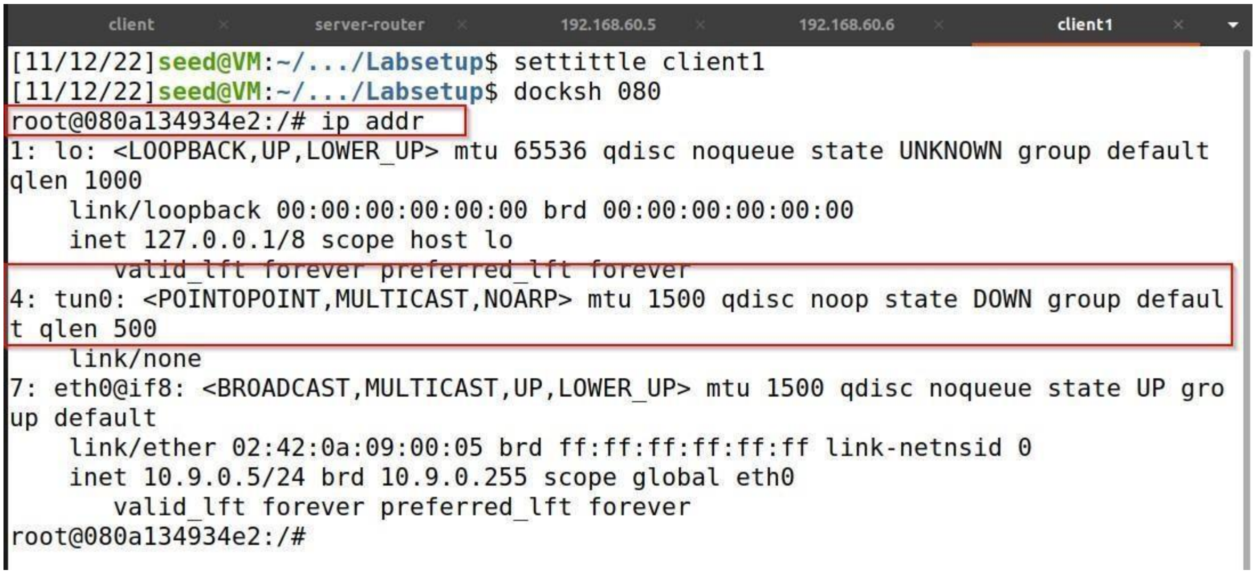
```
root@55d187e99a13:/# tcpdump -i eth1 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
04:52:43.605843 IP 192.168.60.5 > 10.9.0.11: ICMP echo request, id 32, seq 1, leng
th 64
04:52:43.605898 IP 10.9.0.11 > 192.168.60.5: ICMP echo reply, id 32, seq 1, length
 64
04:52:44.607706 IP 192.168.60.5 > 10.9.0.11: ICMP echo request, id 32, seq 2, leng
th 64
04:52:44.607825 IP 10.9.0.11 > 192.168.60.5: ICMP echo reply, id 32, seq 2, length
 64
04:52:48.786400 ARP, Request who-has 192.168.60.5 tell 192.168.60.11, length 28
04:52:48.786465 ARP, Request who-has 192.168.60.11 tell 192.168.60.5, length 28
04:52:48.786471 ARP, Reply 192.168.60.11 is-at 02:42:c0:a8:3c:0b, length 28
04:52:48.786480 ARP, Reply 192.168.60.5 is-at 02:42:c0:a8:3c:05, length 28
```

Task 2: Create and Configure TUN Interface

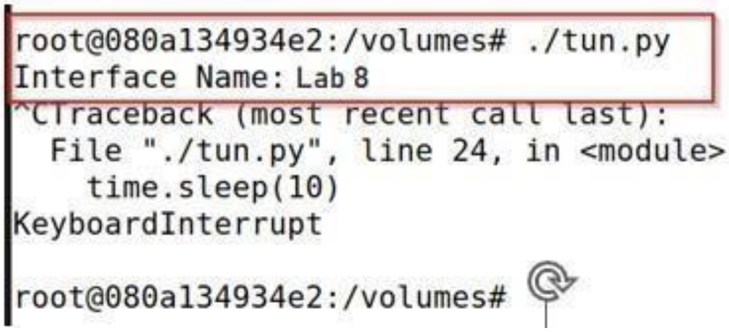
Task 2.a: Name of the Interface



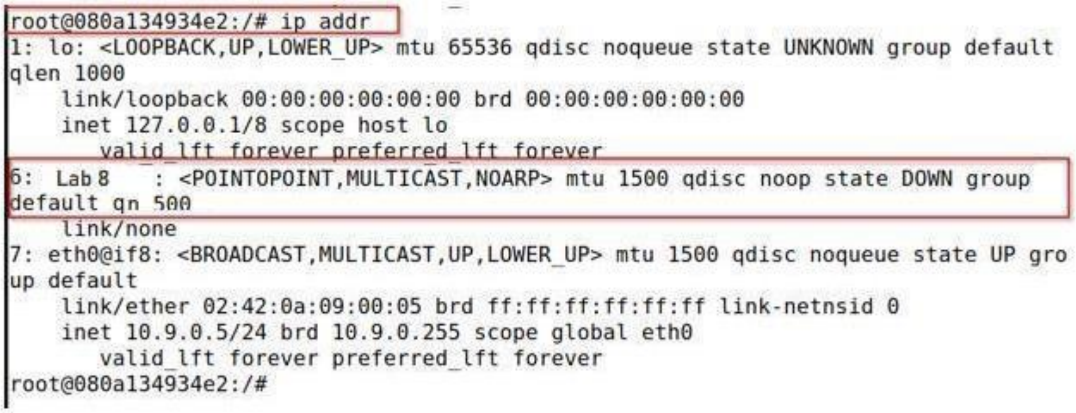
```
root@080a134934e2:/volumes# ls
tun.py
root@080a134934e2:/volumes# chmod a+x tun.py
root@080a134934e2:/volumes# ls
tun.py
root@080a134934e2:/volumes# ./tun.py
Interface Name: tun0
^CTraceback (most recent call last):
  File "./tun.py", line 24, in <module>
    time.sleep(10)
KeyboardInterrupt

root@080a134934e2:/volumes#
```



```
[11/12/22]seed@VM:~/.../Labsetup$ settittle client1
[11/12/22]seed@VM:~/.../Labsetup$ docksh 080
root@080a134934e2:/# 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: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group defaul
t qlen 500
    link/none
7: eth0@if8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP gro
up 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@080a134934e2:/#
```



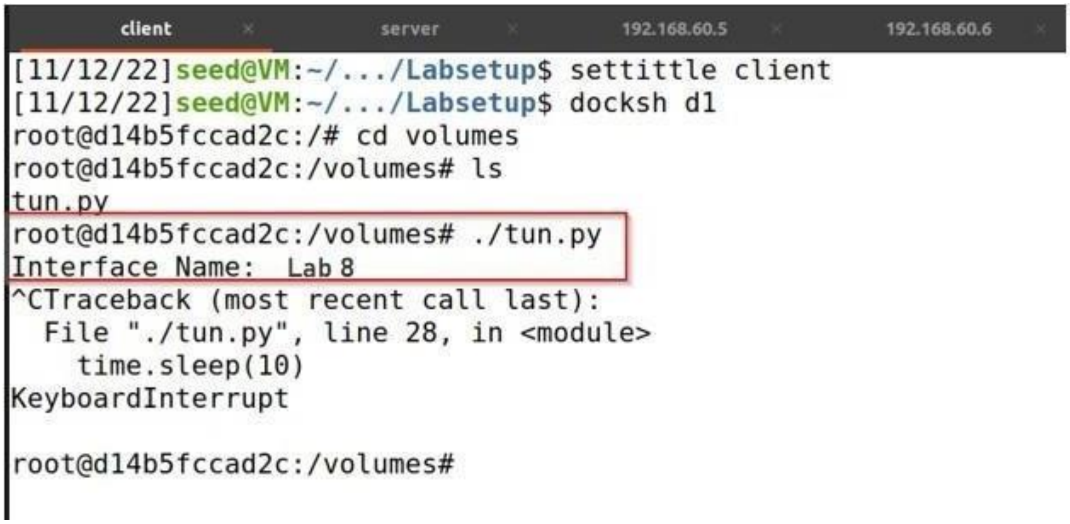
```
root@080a134934e2:/volumes# ./tun.py
Interface Name: Lab 8
^CTraceback (most recent call last):
  File "./tun.py", line 24, in <module>
    time.sleep(10)
KeyboardInterrupt

root@080a134934e2:/volumes#
```



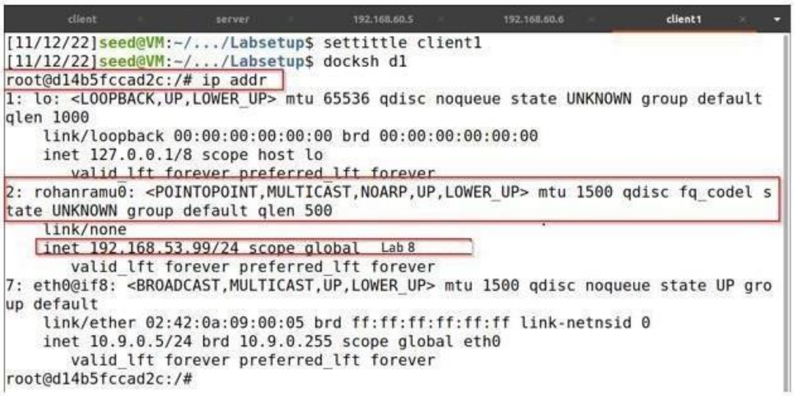
```
root@080a134934e2:/# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
6:  Lab 8    : <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group
default qn 500
    link/none
7: eth0@if8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP gro
up 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@080a134934e2:/#
```

Task 2.b: Set up the TUN Interface

Assign the ip address and bringing up the interface



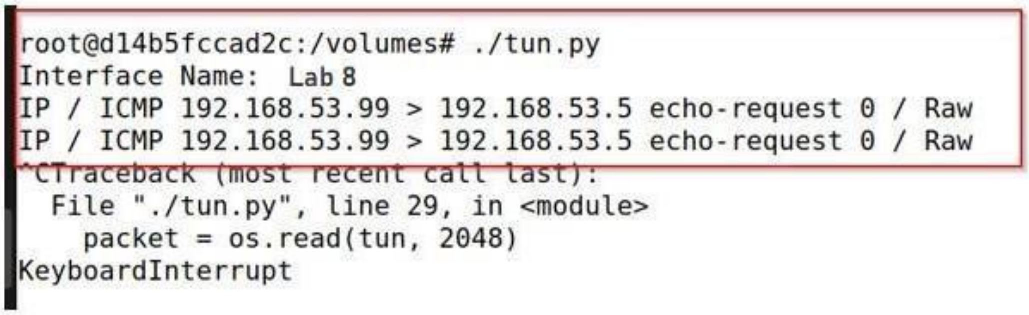
```
[11/12/22]seed@VM:~/.../Labsetup$ settittle client
[11/12/22]seed@VM:~/.../Labsetup$ docksh d1
root@d14b5fccad2c:/# cd volumes
root@d14b5fccad2c:/volumes# ls
tun.py
root@d14b5fccad2c:/volumes# ./tun.py
Interface Name:  Lab 8
^CTraceback (most recent call last):
  File "./tun.py", line 28, in <module>
    time.sleep(10)
KeyboardInterrupt

root@d14b5fccad2c:/volumes#
```



```
[11/12/22]seed@VM:~/.../Labsetup$ settittle client1
[11/12/22]seed@VM:~/.../Labsetup$ docksh d1
root@d14b5fccad2c:/# 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: rohanramu0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel s
tate UNKNOWN group default qlen 500
    link/none
    inet 192.168.53.99/24 scope global  Lab 8
       valid_lft forever preferred_lft forever
7: eth0@if8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP gro
up 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@d14b5fccad2c:/#
```

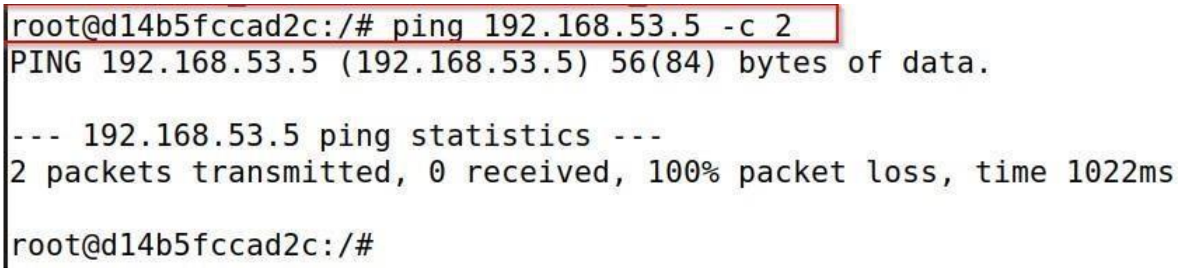
Task 2.c: Read from the TUN Interface

Reading from tunnel interface



```
root@d14b5fccad2c:/volumes# ./tun.py
Interface Name:  Lab 8
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
^CTraceback (most recent call last):
  File "./tun.py", line 29, in <module>
    packet = os.read(tun, 2048)
KeyboardInterrupt
```

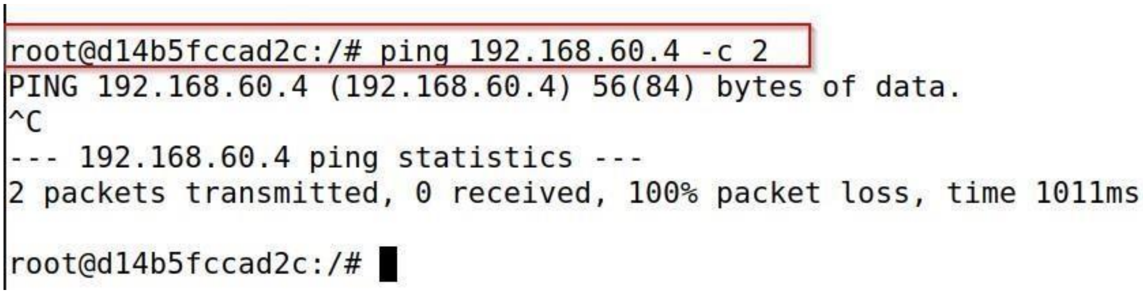
Ping 192.168.53.5 from user u



```
root@d14b5fccad2c:/# ping 192.168.53.5 -c 2
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.

--- 192.168.53.5 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1022ms

root@d14b5fccad2c:/#
```

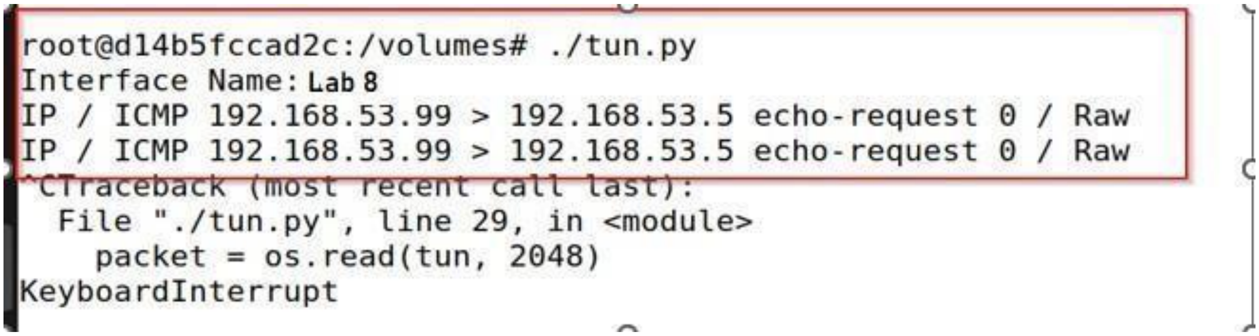
because it is with in the subnet.

Ping 192.168.60.4 from user u



```
root@d14b5fccad2c:/# ping 192.168.60.4 -c 2
PING 192.168.60.4 (192.168.60.4) 56(84) bytes of data.
^C
--- 192.168.60.4 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1011ms

root@d14b5fccad2c:/#
```

We cannot find any data in tunnel

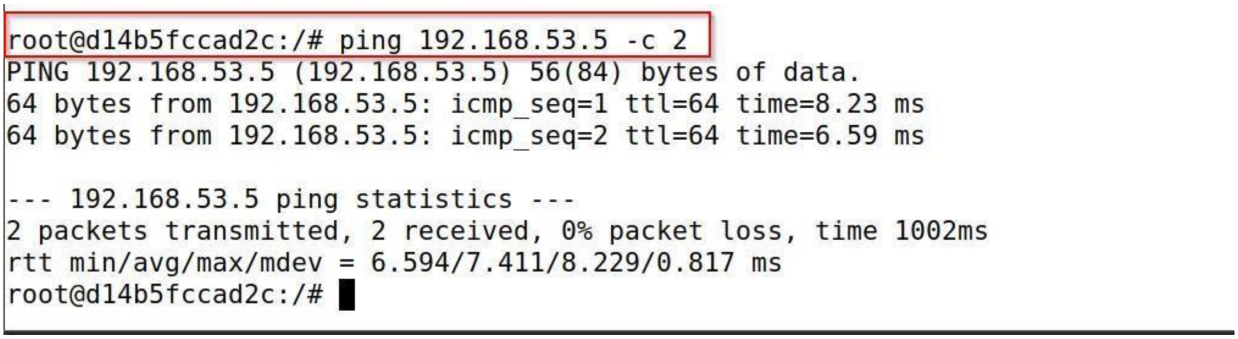


```
root@d14b5fccad2c:/volumes# ./tun.py
Interface Name: Lab 8
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
^CTraceback (most recent call last):
  File "./tun.py", line 29, in <module>
    packet = os.read(tun, 2048)
KeyboardInterrupt
```

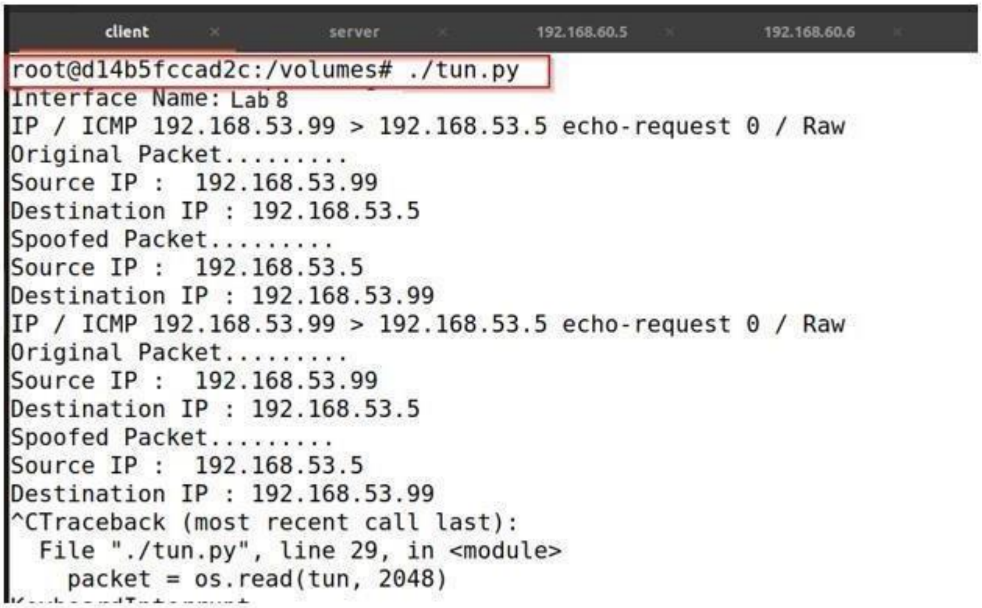
Routing table to determine interface based on the destination ip address, if we going to ping the ip address within the subnet my tun interface going to pick it up

Task 2.d: Write to the TUN Interface

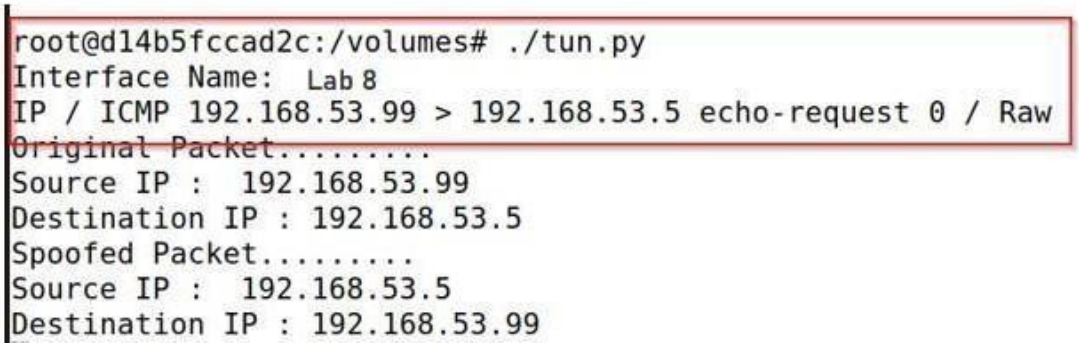
Ping 192.168.53.5 from user u



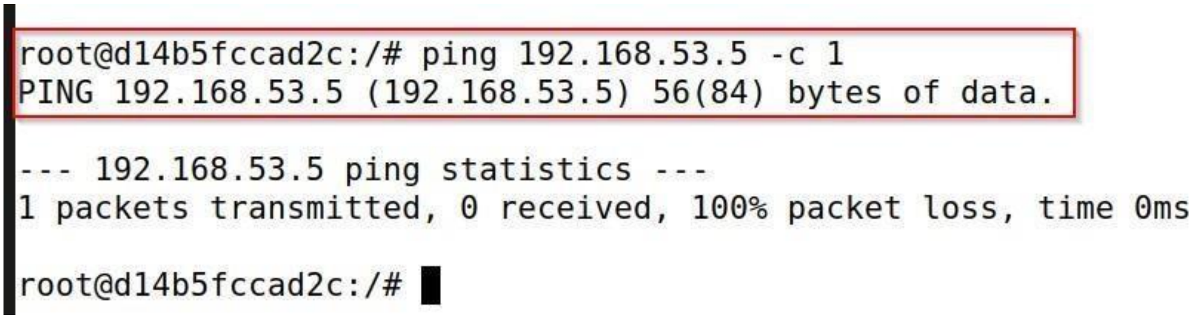
```
root@d14b5fccad2c:/# ping 192.168.53.5 -c 2
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.
64 bytes from 192.168.53.5: icmp_seq=1 ttl=64 time=8.23 ms
64 bytes from 192.168.53.5: icmp_seq=2 ttl=64 time=6.59 ms

--- 192.168.53.5 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 6.594/7.411/8.229/0.817 ms
root@d14b5fccad2c:/#
```



```
root@d14b5fccad2c:/volumes# ./tun.py
Interface Name: Lab 8
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
Original Packet.........
Source IP :  192.168.53.99
Destination IP : 192.168.53.5
Spoofed Packet.........
Source IP :  192.168.53.5
Destination IP : 192.168.53.99
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
Original Packet.........
Source IP :  192.168.53.99
Destination IP : 192.168.53.5
Spoofed Packet.........
Source IP :  192.168.53.5
Destination IP : 192.168.53.99
^CTraceback (most recent call last):
  File "./tun.py", line 29, in <module>
    packet = os.read(tun, 2048)
KeyboardInterrupt
```



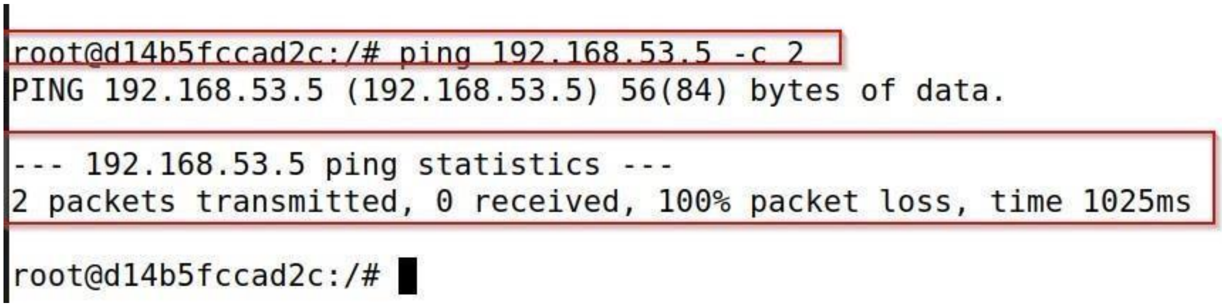
```
root@d14b5fccad2c:/volumes# ./tun.py
Interface Name:  Lab 8
IP / ICMP 192.168.53.99 > 192.168.53.5 echo-request 0 / Raw
Original Packet.........
Source IP :  192.168.53.99
Destination IP : 192.168.53.5
Spoofed Packet.........
Source IP :  192.168.53.5
Destination IP : 192.168.53.99
```



```
root@d14b5fccad2c:/# ping 192.168.53.5 -c 1
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.

--- 192.168.53.5 ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

root@d14b5fccad2c:/# 
```

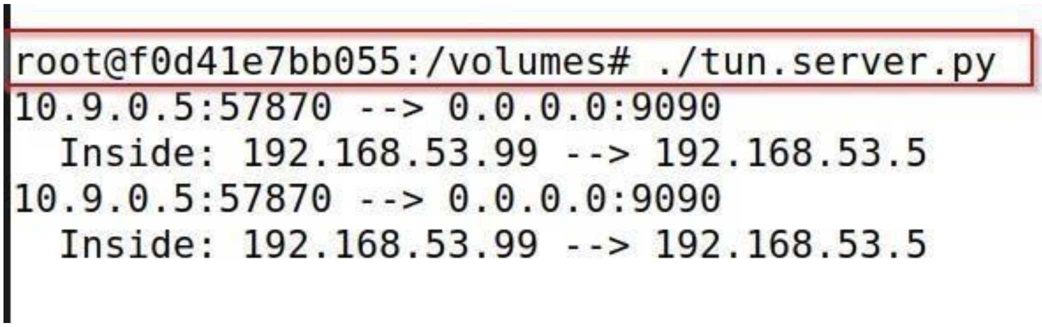
Task 3: Send the IP Packet to VPN Server Through a Tunnel

Ping 192.168.53.5 from host u



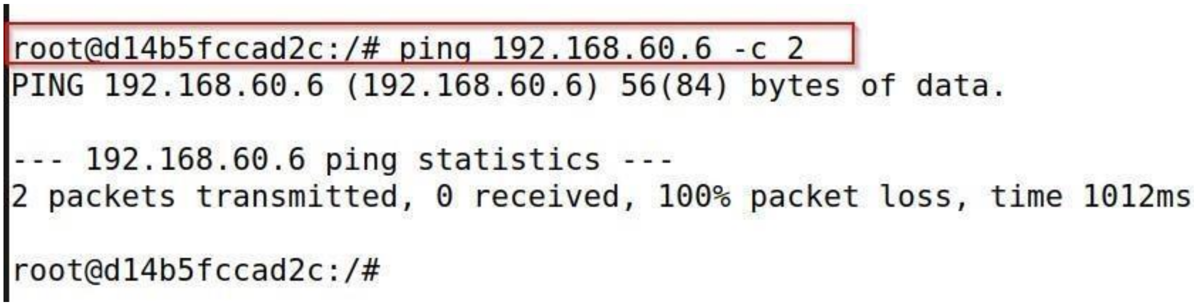
```
root@d14b5fccad2c:/# ping 192.168.53.5 -c 2
PING 192.168.53.5 (192.168.53.5) 56(84) bytes of data.

--- 192.168.53.5 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1025ms

root@d14b5fccad2c:/# 
```

Checking the output in vpn server



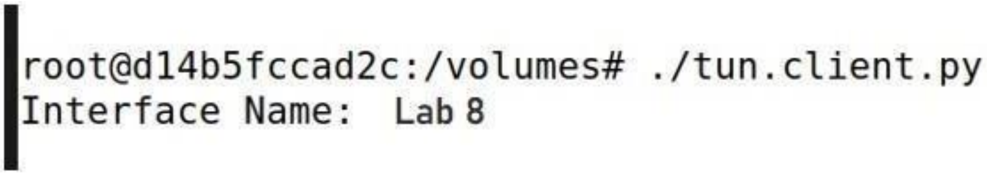
```
root@f0d41e7bb055:/volumes# ./tun.server.py
10.9.0.5:57870 --> 0.0.0.0:9090
  Inside: 192.168.53.99 --> 192.168.53.5
10.9.0.5:57870 --> 0.0.0.0:9090
  Inside: 192.168.53.99 --> 192.168.53.5
```

If we ping 192.168.60.6

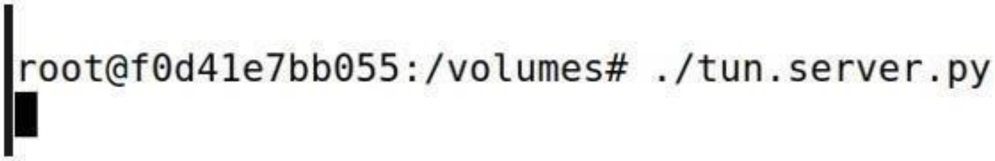


```
root@d14b5fccad2c:/# ping 192.168.60.6 -c 2
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.

--- 192.168.60.6 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1012ms

root@d14b5fccad2c:/#
```

There is no change in host u and vpn server



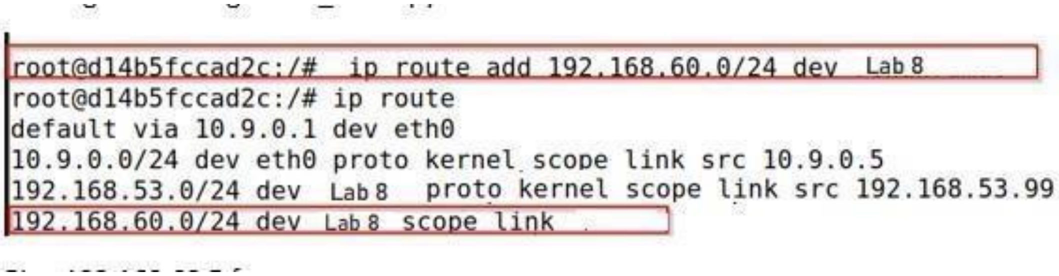
```
root@d14b5fccad2c:/volumes# ./tun.client.py
Interface Name:  Lab 8
```



```
root@f0d41e7bb055:/volumes# ./tun.server.py
```

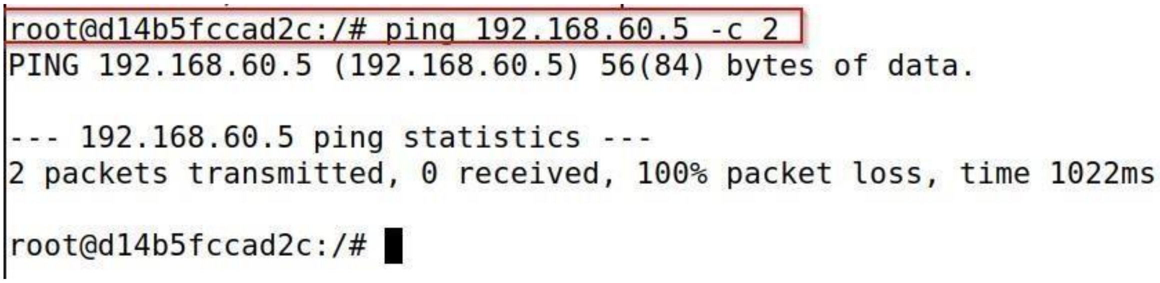
because routing entry doesn’t set properly, in the current routing entry all the packets are forwarded to eth0, in order to reach to the private network, we need to go tun interface.

Adding the routing in tun_client.py



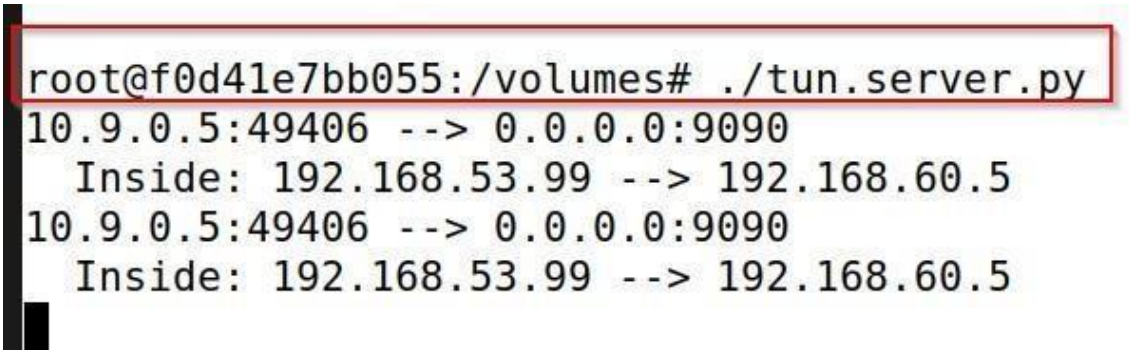
```
root@d14b5fccad2c:/#  ip route add 192.168.60.0/24 dev  Lab 8
root@d14b5fccad2c:/# ip route
default via 10.9.0.1 dev eth0
10.9.0.0/24 dev eth0 proto kernel scope link src 10.9.0.5
192.168.53.0/24 dev  Lab 8   proto kernel scope link src 192.168.53.99
192.168.60.0/24 dev  Lab 8  scope link
```

Ping 192.168.60.5 from user u



```
root@d14b5fccad2c:/# ping 192.168.60.5 -c 2
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.

--- 192.168.60.5 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1022ms

root@d14b5fccad2c:/#
```

Output at vpn server

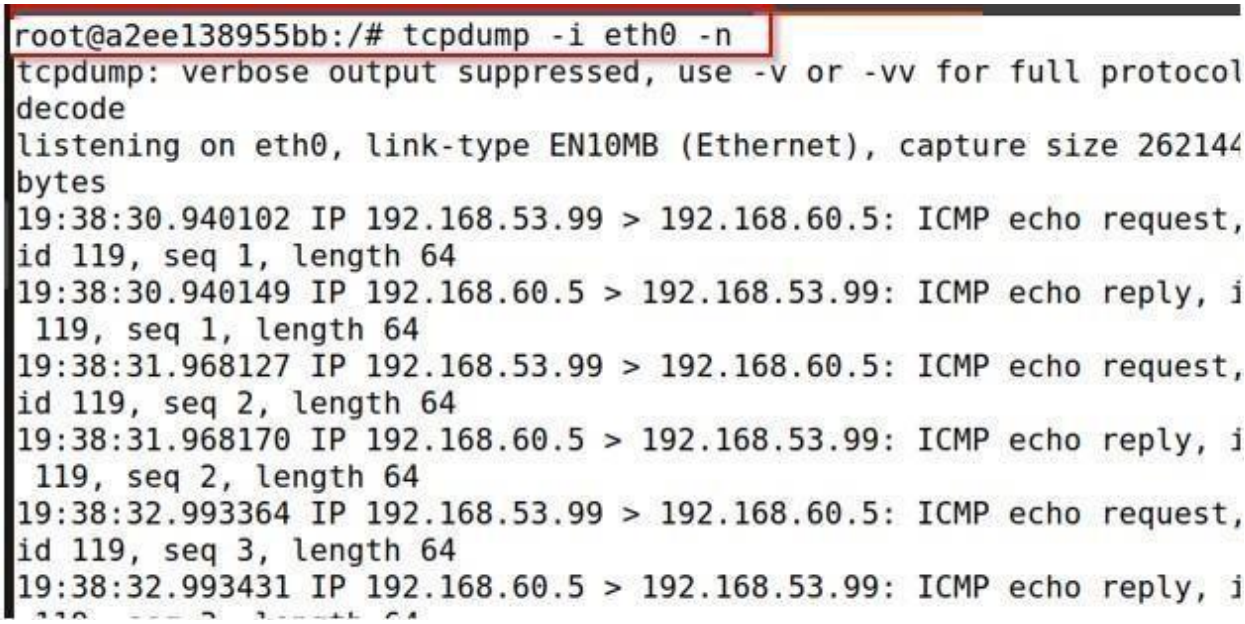


```
root@f0d41e7bb055:/volumes# ./tun.server.py
10.9.0.5:49406 --> 0.0.0.0:9090
  Inside: 192.168.53.99 --> 192.168.60.5
10.9.0.5:49406 --> 0.0.0.0:9090
  Inside: 192.168.53.99 --> 192.168.60.5
```

the problem here it is only one way, we can read tun interface and send it out but cannot write because it is only one way direction.

Task 4: Set Up the VPN Server

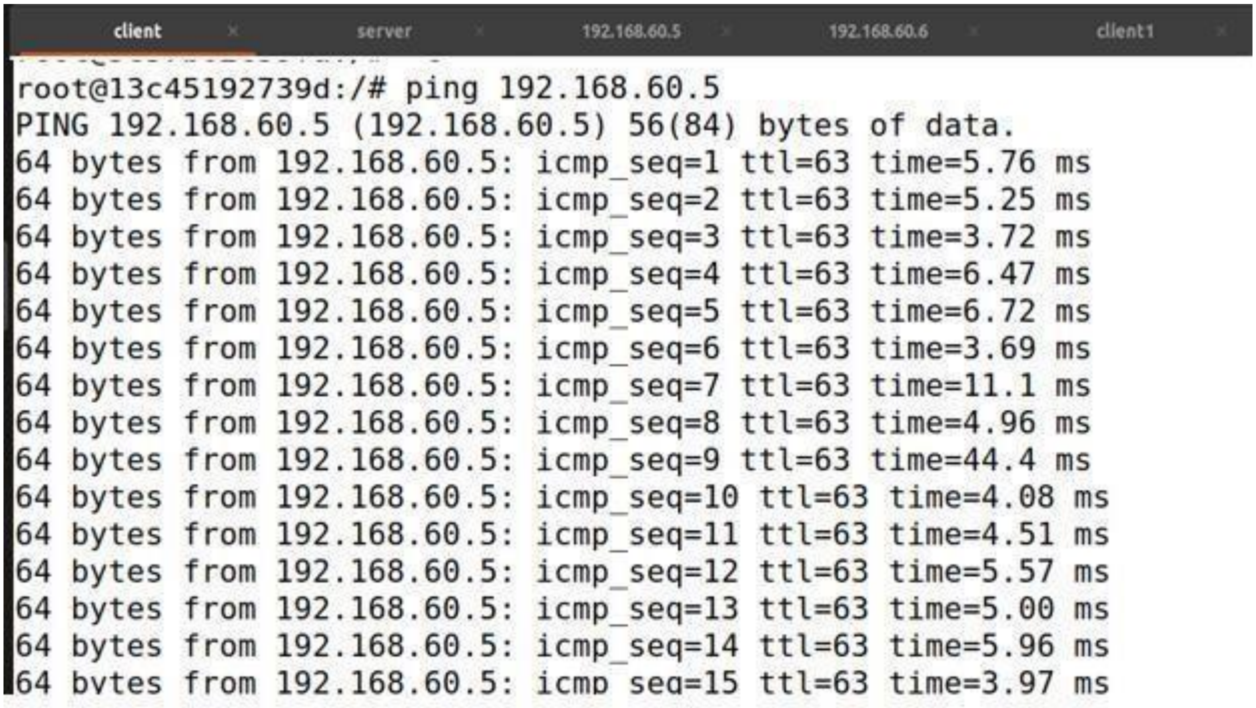
Ping 192.168.60.5 on user u and execute tcpdump on host v to check whether we have icmp packets on Host V



```
root@a2ee138955bb:/# 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 26214
bytes
19:38:30.940102 IP 192.168.53.99 > 192.168.60.5: ICMP echo request,
id 119, seq 1, length 64
19:38:30.940149 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, i
 119, seq 1, length 64
19:38:31.968127 IP 192.168.53.99 > 192.168.60.5: ICMP echo request,
id 119, seq 2, length 64
19:38:31.968170 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, i
 119, seq 2, length 64
19:38:32.993364 IP 192.168.53.99 > 192.168.60.5: ICMP echo request,
id 119, seq 3, length 64
19:38:32.993431 IP 192.168.60.5 > 192.168.53.99: ICMP echo reply, i
 119, seq 3, length 64
```

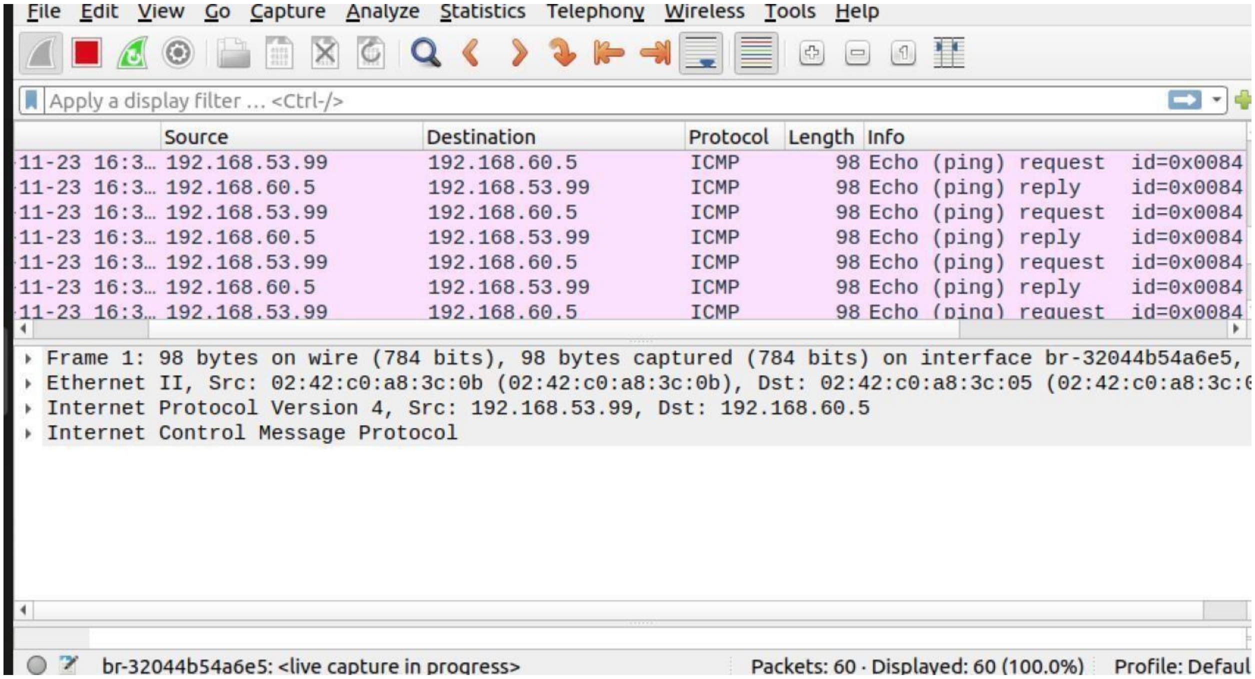
Task 5: Handling Traffic in Both Directions

Ping 192.168.60.5 to check the traffic in two directions



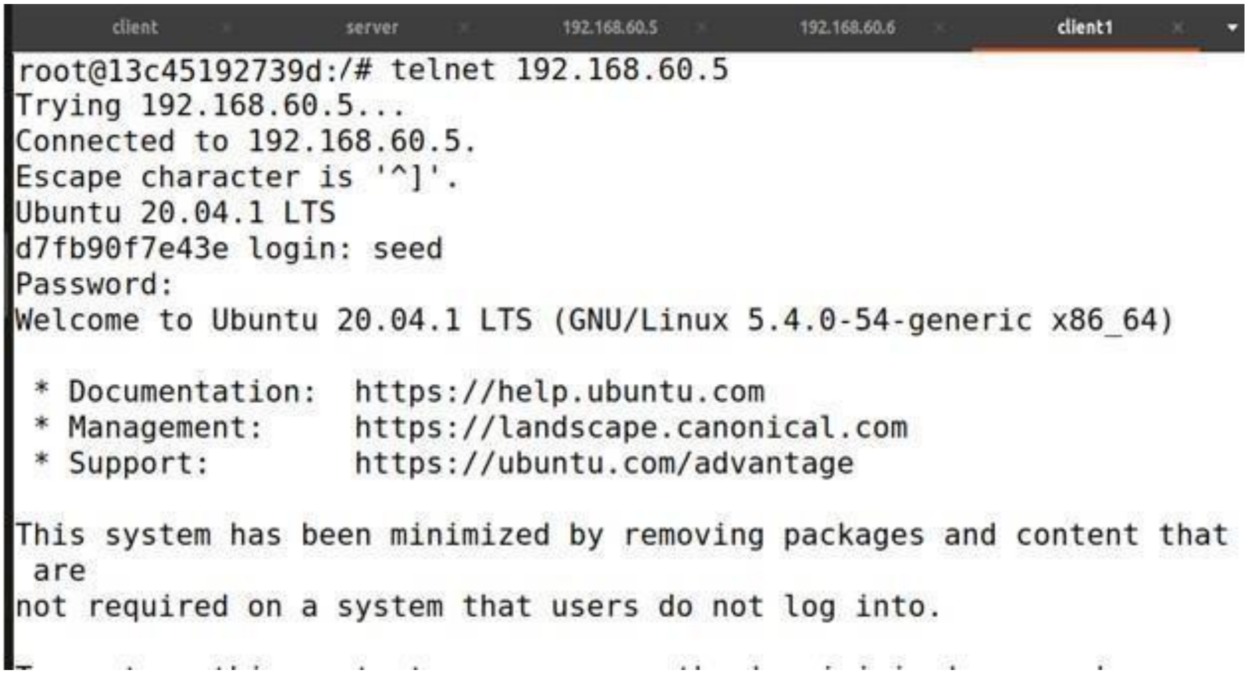
```
root@13c45192739d:/# ping 192.168.60.5
PING 192.168.60.5 (192.168.60.5) 56(84) bytes of data.
64 bytes from 192.168.60.5: icmp_seq=1 ttl=63 time=5.76 ms
64 bytes from 192.168.60.5: icmp_seq=2 ttl=63 time=5.25 ms
64 bytes from 192.168.60.5: icmp_seq=3 ttl=63 time=3.72 ms
64 bytes from 192.168.60.5: icmp_seq=4 ttl=63 time=6.47 ms
64 bytes from 192.168.60.5: icmp_seq=5 ttl=63 time=6.72 ms
64 bytes from 192.168.60.5: icmp_seq=6 ttl=63 time=3.69 ms
64 bytes from 192.168.60.5: icmp_seq=7 ttl=63 time=11.1 ms
64 bytes from 192.168.60.5: icmp_seq=8 ttl=63 time=4.96 ms
64 bytes from 192.168.60.5: icmp_seq=9 ttl=63 time=44.4 ms
64 bytes from 192.168.60.5: icmp_seq=10 ttl=63 time=4.08 ms
64 bytes from 192.168.60.5: icmp_seq=11 ttl=63 time=4.51 ms
64 bytes from 192.168.60.5: icmp_seq=12 ttl=63 time=5.57 ms
64 bytes from 192.168.60.5: icmp_seq=13 ttl=63 time=5.00 ms
64 bytes from 192.168.60.5: icmp_seq=14 ttl=63 time=5.96 ms
64 bytes from 192.168.60.5: icmp_seq=15 ttl=63 time=3.97 ms
```

wireshark results


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

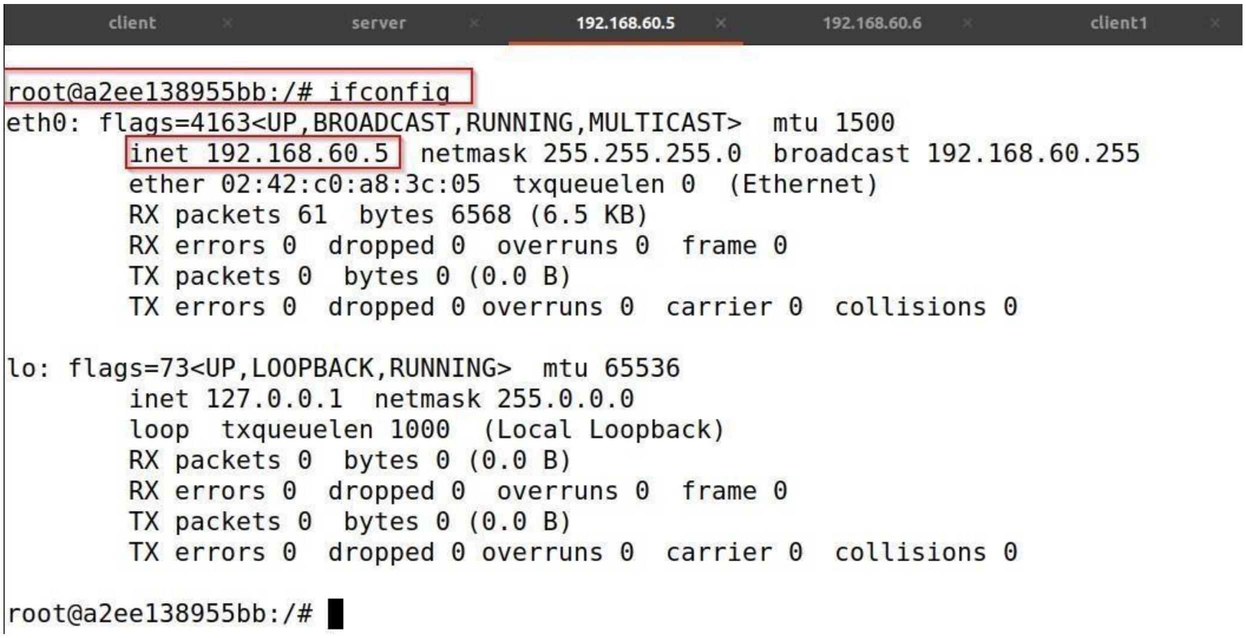
```
                Source           Destination      Protocol Length Info
11-23 16:3… 192.168.53.99    192.168.60.5     ICMP        98 Echo (ping) request  id=0x0084
11-23 16:3… 192.168.60.5     192.168.53.99    ICMP        98 Echo (ping) reply    id=0x0084
11-23 16:3… 192.168.53.99    192.168.60.5     ICMP        98 Echo (ping) request  id=0x0084
11-23 16:3… 192.168.60.5     192.168.53.99    ICMP        98 Echo (ping) reply    id=0x0084
11-23 16:3… 192.168.53.99    192.168.60.5     ICMP        98 Echo (ping) request  id=0x0084
11-23 16:3… 192.168.60.5     192.168.53.99    ICMP        98 Echo (ping) reply    id=0x0084
11-23 16:3… 192.168.53.99    192.168.60.5     ICMP        98 Echo (ping) request  id=0x0084

▸ Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface br-32044b54a6e5,
▸ Ethernet II, Src: 02:42:c0:a8:3c:0b (02:42:c0:a8:3c:0b), Dst: 02:42:c0:a8:3c:05 (02:42:c0:a8:3c:0
▸ Internet Protocol Version 4, Src: 192.168.53.99, Dst: 192.168.60.5
▸ Internet Control Message Protocol

br-32044b54a6e5: <live capture in progress>      Packets: 60 · Displayed: 60 (100.0%)    Profile: Defaul
```

Telnet 192.168.60.5



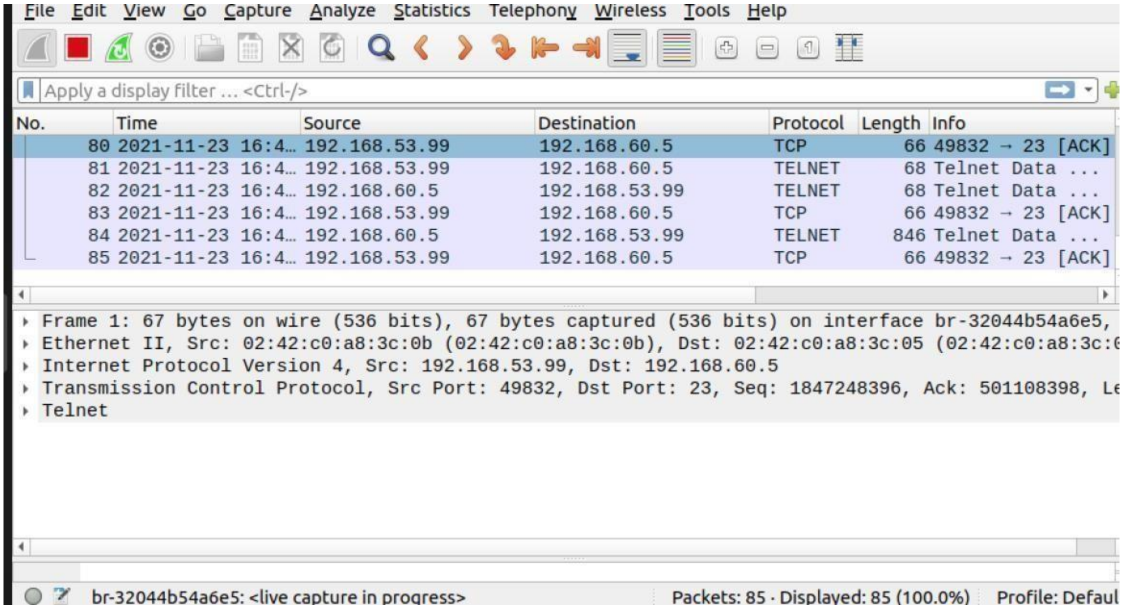
```
root@13c45192739d:/# telnet 192.168.60.5
Trying 192.168.60.5...
Connected to 192.168.60.5.
Escape character is '^]'.
Ubuntu 20.04.1 LTS
d7fb90f7e43e login: seed
Password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-54-generic x86_64)

 * 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.
```

Verifying the ip



```
root@a2ee138955bb:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.60.5  netmask 255.255.255.0  broadcast 192.168.60.255
        ether 02:42:c0:a8:3c:05  txqueuelen 0  (Ethernet)
        RX packets 61  bytes 6568 (6.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@a2ee138955bb:/#
```

wireshark results

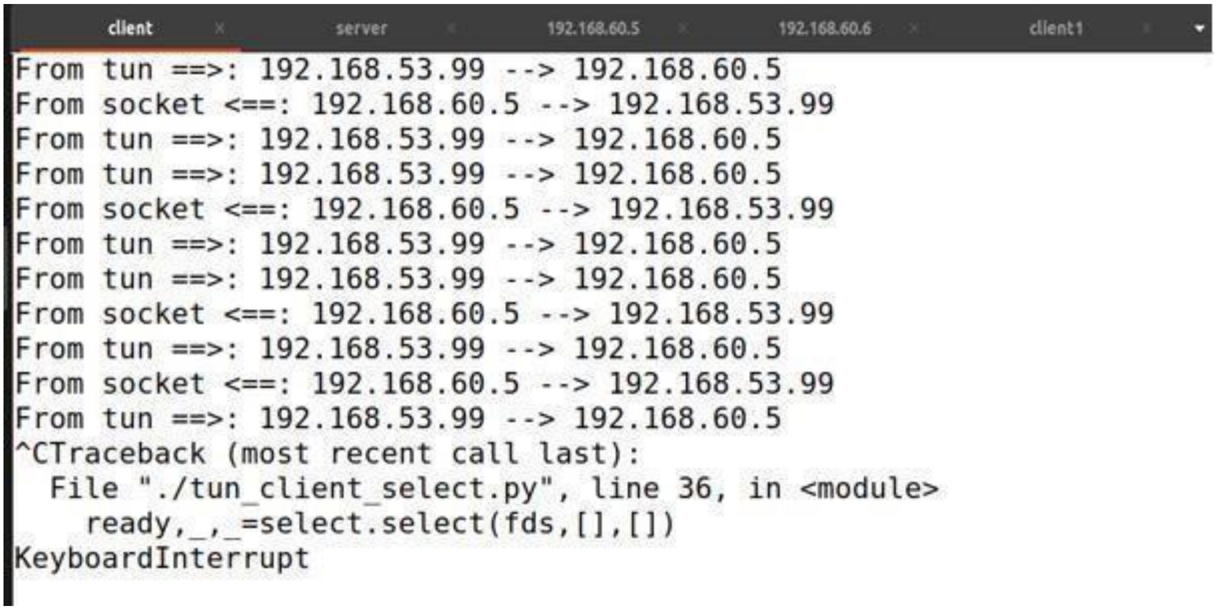

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>

```
No.   Time              Source          Destination     Protocol  Length Info
   80 2021-11-23 16:4… 192.168.53.99   192.168.60.5    TCP           66 49832 → 23 [ACK]
   81 2021-11-23 16:4… 192.168.53.99   192.168.60.5    TELNET        68 Telnet Data ...
   82 2021-11-23 16:4… 192.168.60.5    192.168.53.99   TELNET        68 Telnet Data ...
   83 2021-11-23 16:4… 192.168.53.99   192.168.60.5    TCP           66 49832 → 23 [ACK]
   84 2021-11-23 16:4… 192.168.60.5    192.168.53.99   TELNET       846 Telnet Data ...
   85 2021-11-23 16:4… 192.168.53.99   192.168.60.5    TCP           66 49832 → 23 [ACK]

› Frame 1: 67 bytes on wire (536 bits), 67 bytes captured (536 bits) on interface br-32044b54a6e5,
› Ethernet II, Src: 02:42:c0:a8:3c:0b (02:42:c0:a8:3c:0b), Dst: 02:42:c0:a8:3c:05 (02:42:c0:a8:3c:0
› Internet Protocol Version 4, Src: 192.168.53.99, Dst: 192.168.60.5
› Transmission Control Protocol, Src Port: 49832, Dst Port: 23, Seq: 1847248396, Ack: 501108398, Le
› Telnet

br-32044b54a6e5: <live capture in progress>    Packets: 85 · Displayed: 85 (100.0%)    Profile: Defaul
```

the packets are running successfully in two directions.

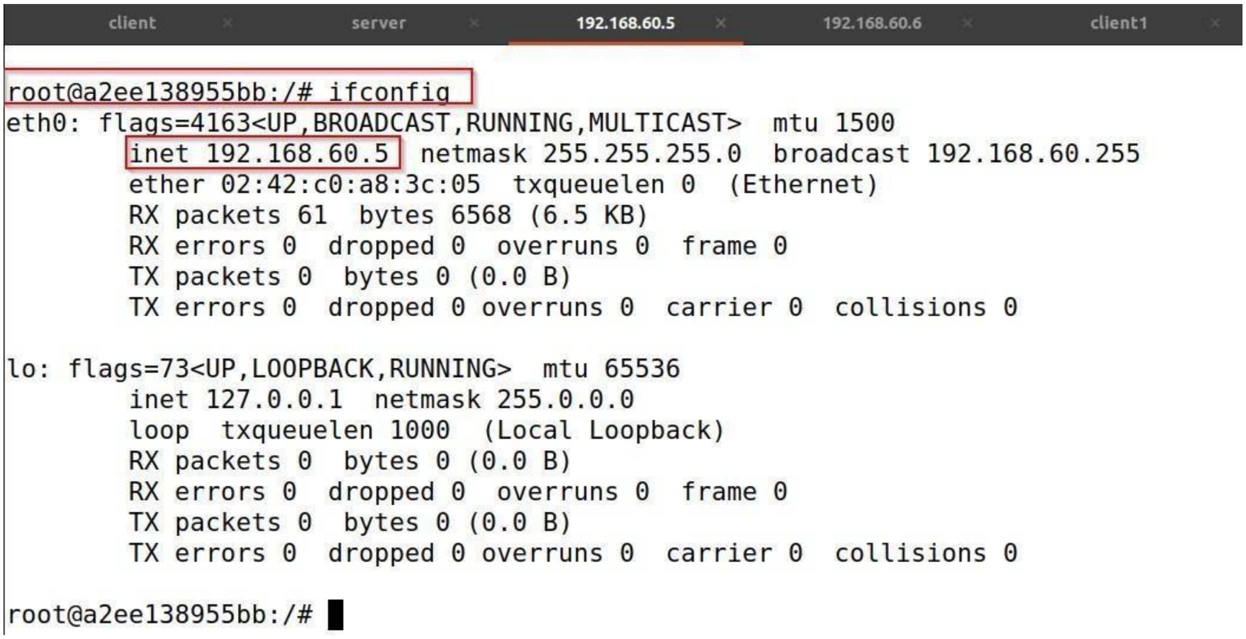
Task 6: Tunnel-Breaking Experiment

Breaking the tunnel



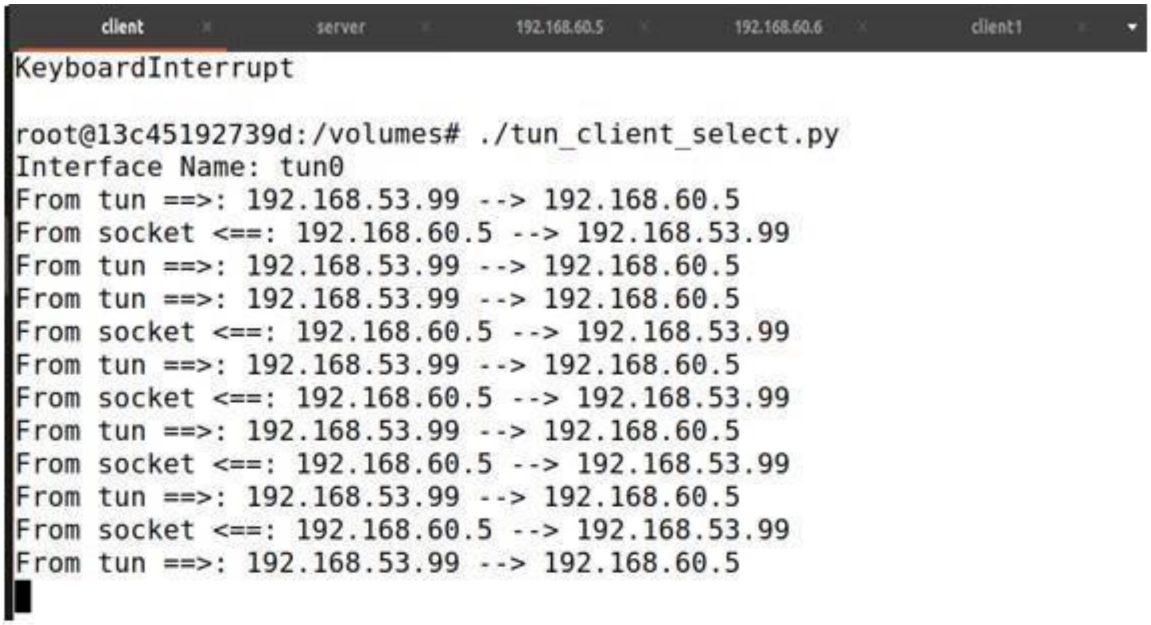
```
From tun ==>: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==>: 192.168.53.99 --> 192.168.60.5
From tun ==>: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==>: 192.168.53.99 --> 192.168.60.5
From tun ==>: 192.168.53.99 --> 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
^CTraceback (most recent call last):
  File "./tun_client_select.py", line 36, in <module>
    ready,_,_=select.select(fds,[],[])
KeyboardInterrupt
```

We are unable to get any data if we in telnet connection



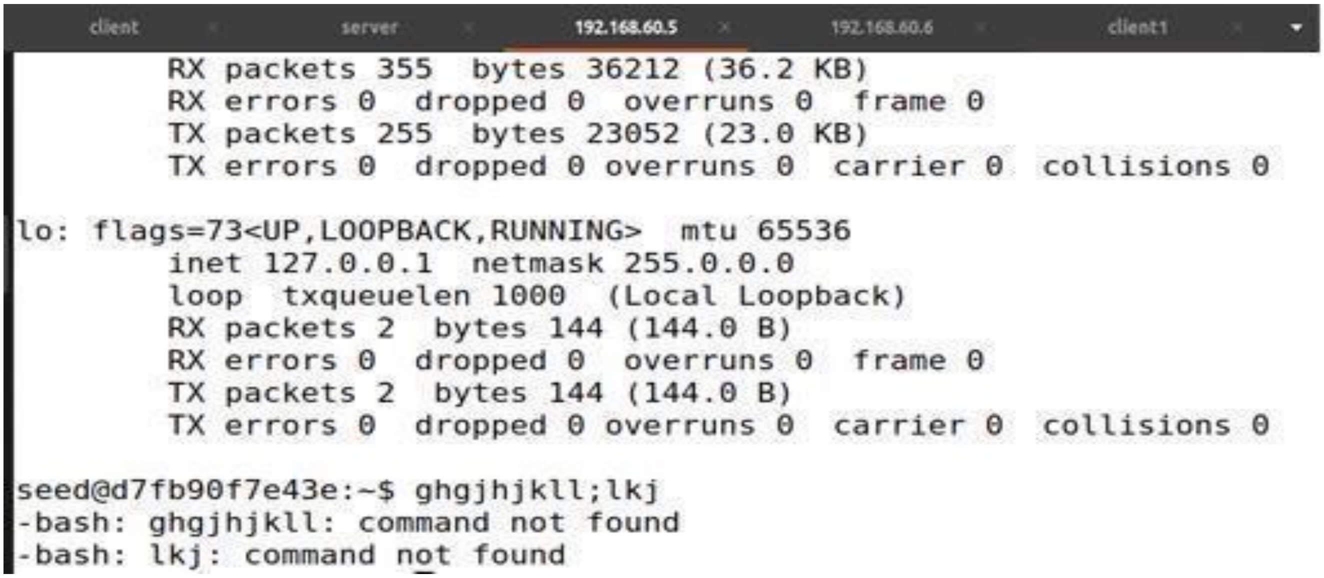
```
root@a2ee138955bb:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.60.5  netmask 255.255.255.0  broadcast 192.168.60.255
        ether 02:42:c0:a8:3c:05  txqueuelen 0  (Ethernet)
        RX packets 61  bytes 6568 (6.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@a2ee138955bb:/# 
```

Restart The tunnel



```
KeyboardInterrupt

root@13c45192739d:/volumes# ./tun_client_select.py
Interface Name: tun0
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 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
```

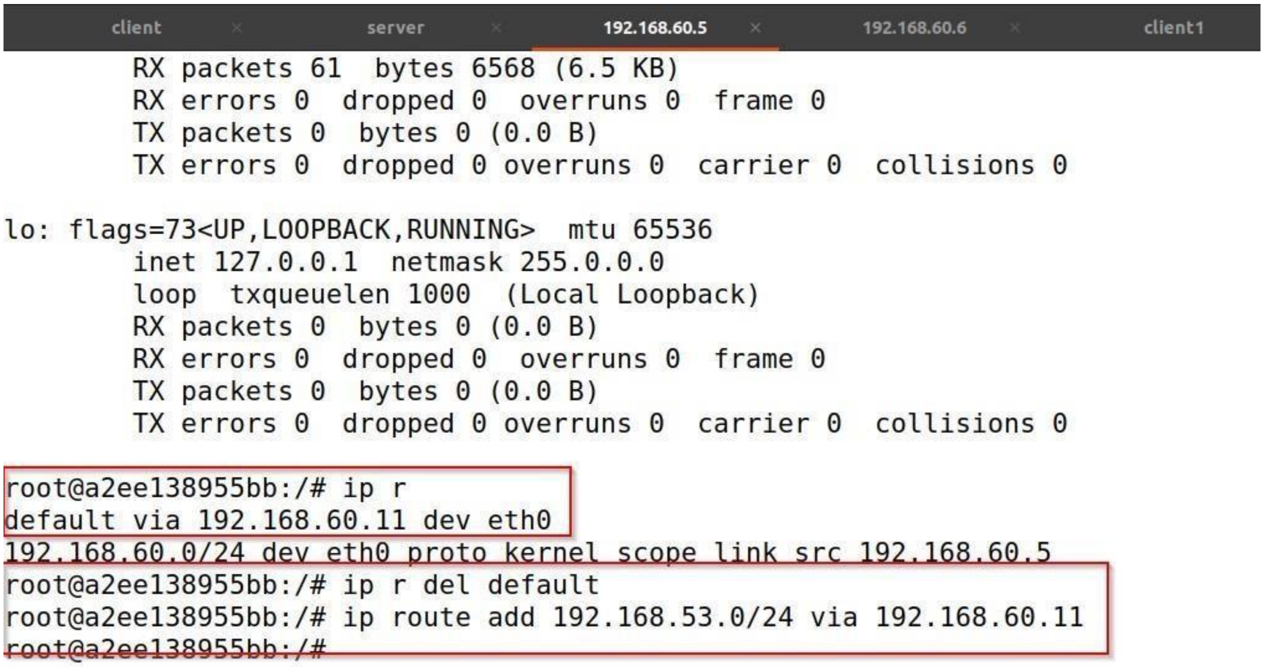
What ever be the data we type during the break of tunnel It comes back after we restart the tunnel



```
        RX packets 355  bytes 36212 (36.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 255  bytes 23052 (23.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 2  bytes 144 (144.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2  bytes 144 (144.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

seed@d7fb90f7e43e:~$ ghgjhjkll;lkj
-bash: ghgjhjkll: command not found
-bash: lkj: command not found
```

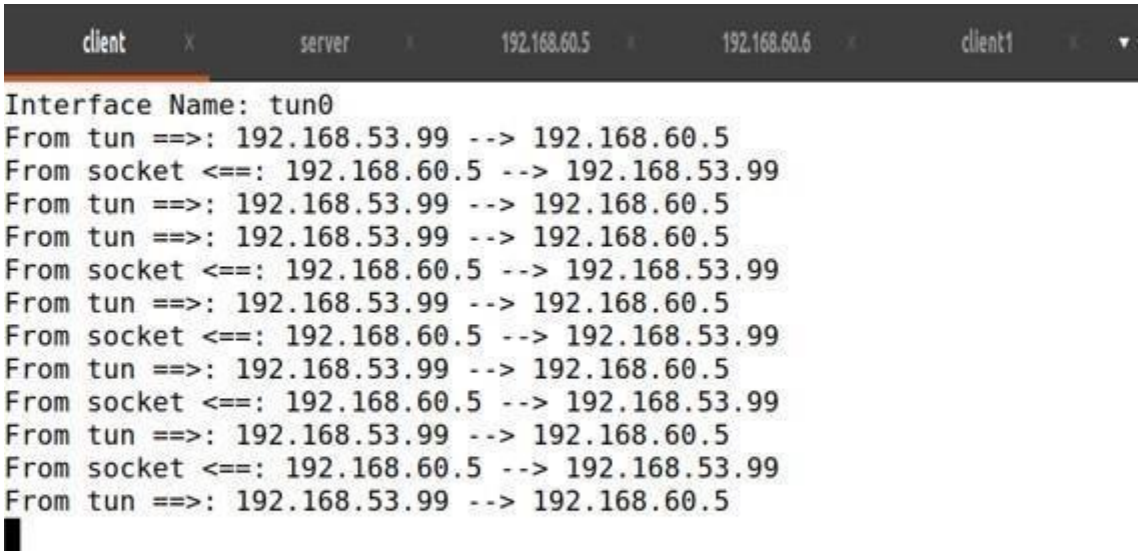
when the connection is lost, the telnet connection will buffer all the characters we type, when the channel re-connects, we will receive all those characters, these acrobats will be saved in buffer.

Task 7: Routing Experiment on Host V



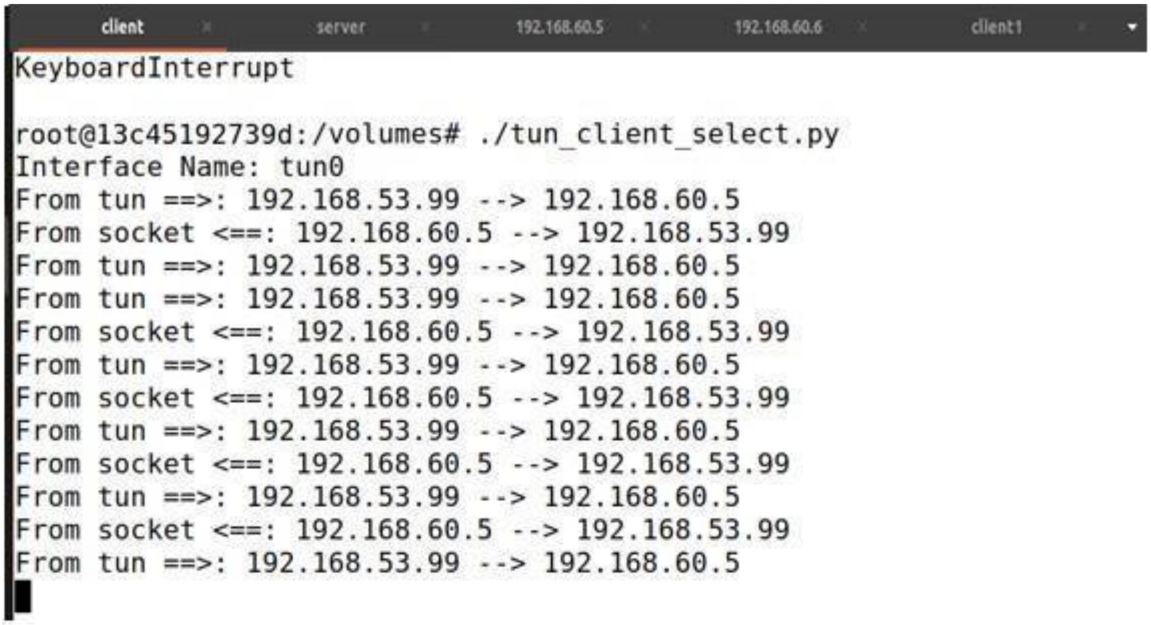
```
        RX packets 61  bytes 6568 (6.5 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@a2ee138955bb:/# ip r
default via 192.168.60.11 dev eth0
192.168.60.0/24 dev eth0 proto kernel scope link src 192.168.60.5
root@a2ee138955bb:/# ip r del default
root@a2ee138955bb:/# ip route add 192.168.53.0/24 via 192.168.60.11
root@a2ee138955bb:/#
```

Stopping the tunnel



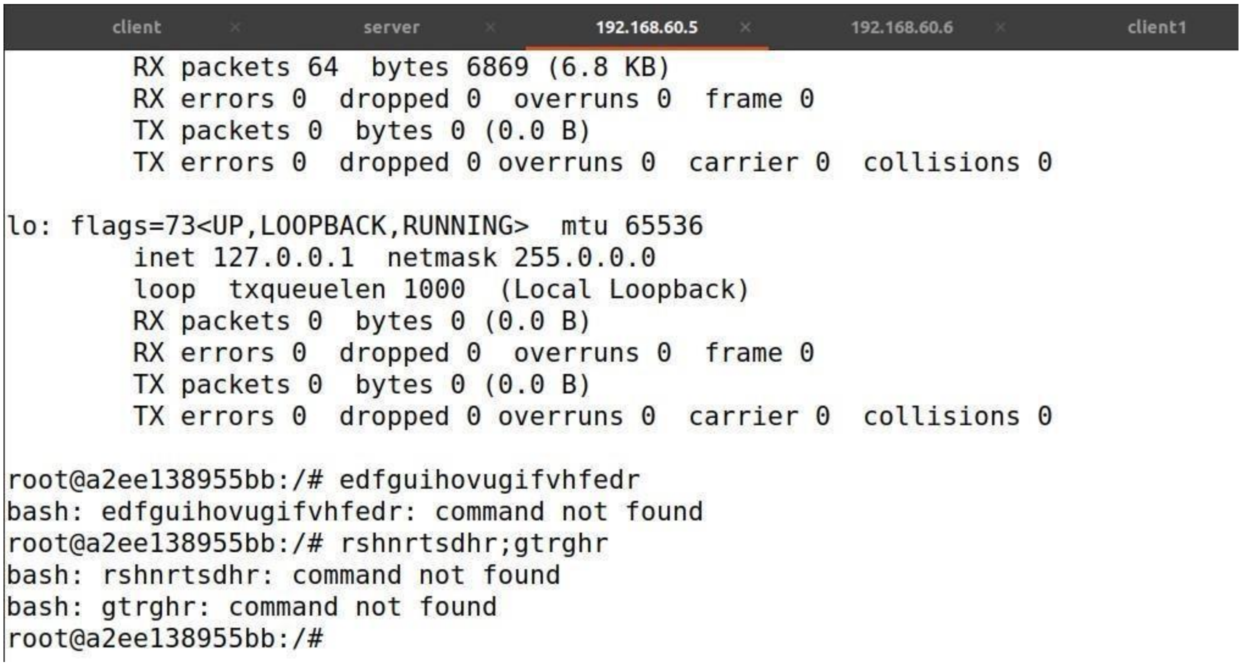
```
Interface Name: tun0
From tun ==>: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==>: 192.168.53.99 --> 192.168.60.5
From tun ==>: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==>: 192.168.53.99 --> 192.168.60.5
From socket <==: 192.168.60.5 --> 192.168.53.99
From tun ==>: 192.168.53.99 --> 192.168.60.5
From 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
```

Restart tunnel



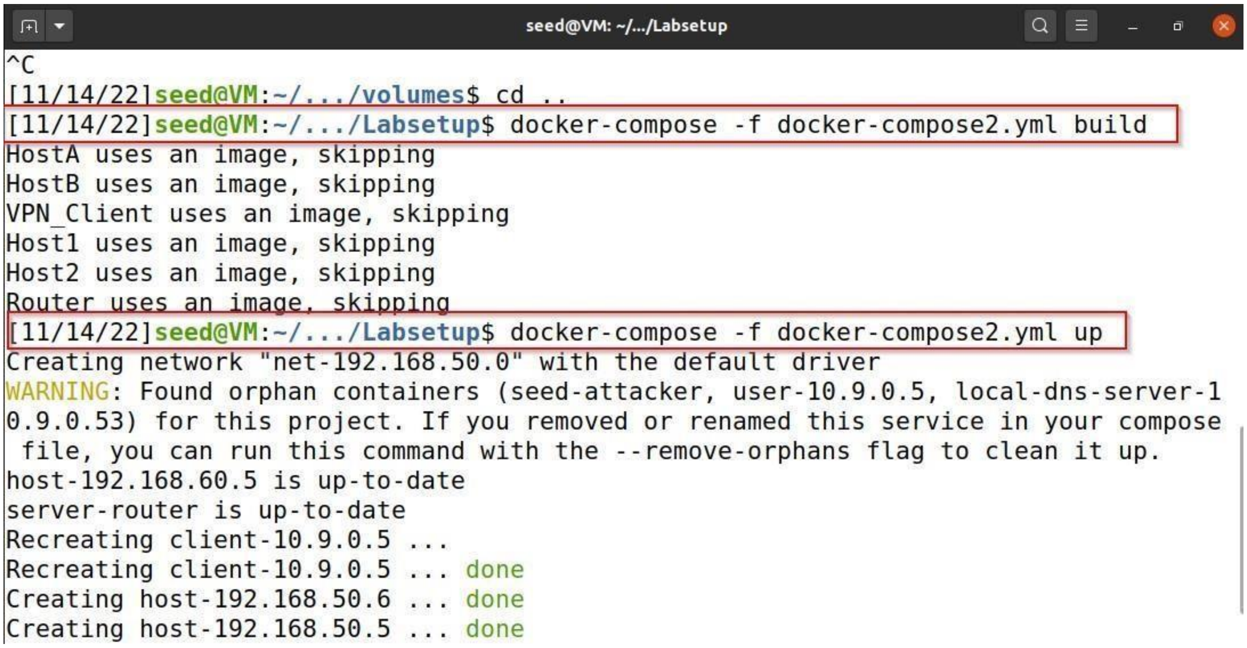
```
KeyboardInterrupt

root@13c45192739d:/volumes# ./tun_client_select.py
Interface Name: tun0
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 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
```

Telnet is working post restart of tunnel

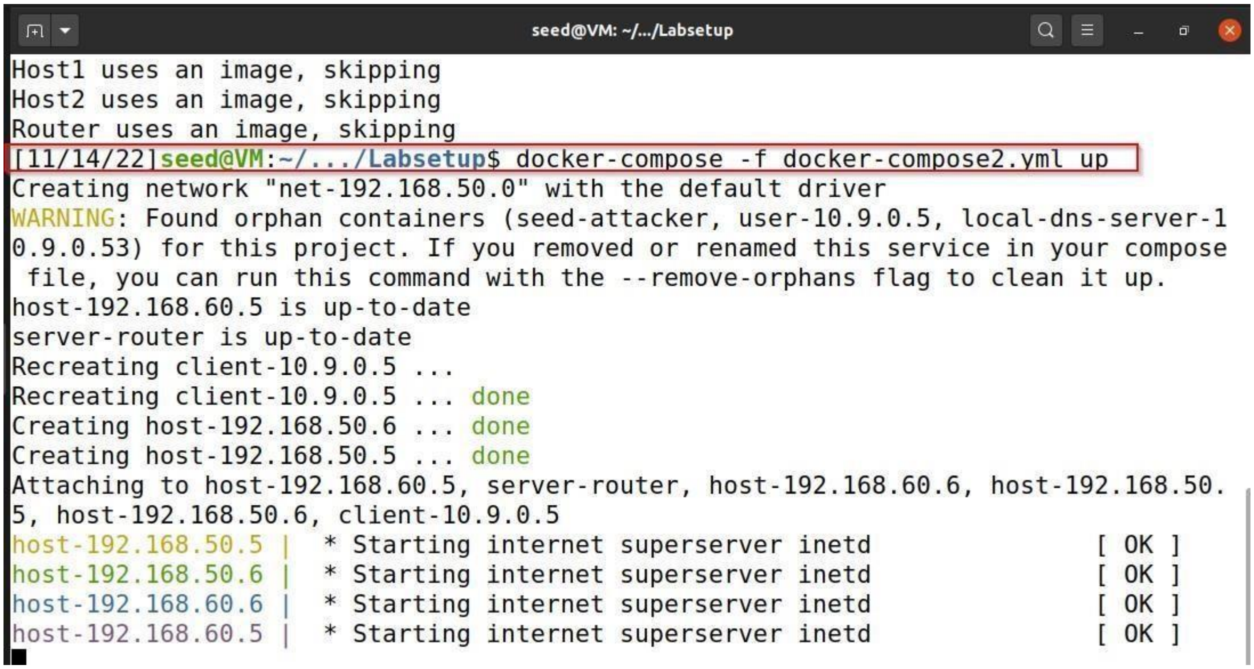


```
        RX packets 64  bytes 6869 (6.8 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@a2ee138955bb:/# edfguihovugifvhfedr
bash: edfguihovugifvhfedr: command not found
root@a2ee138955bb:/# rshnrtsdhr;gtrghr
bash: rshnrtsdhr: command not found
bash: gtrghr: command not found
root@a2ee138955bb:/#
```

Task 8: VPN Between Private Networks

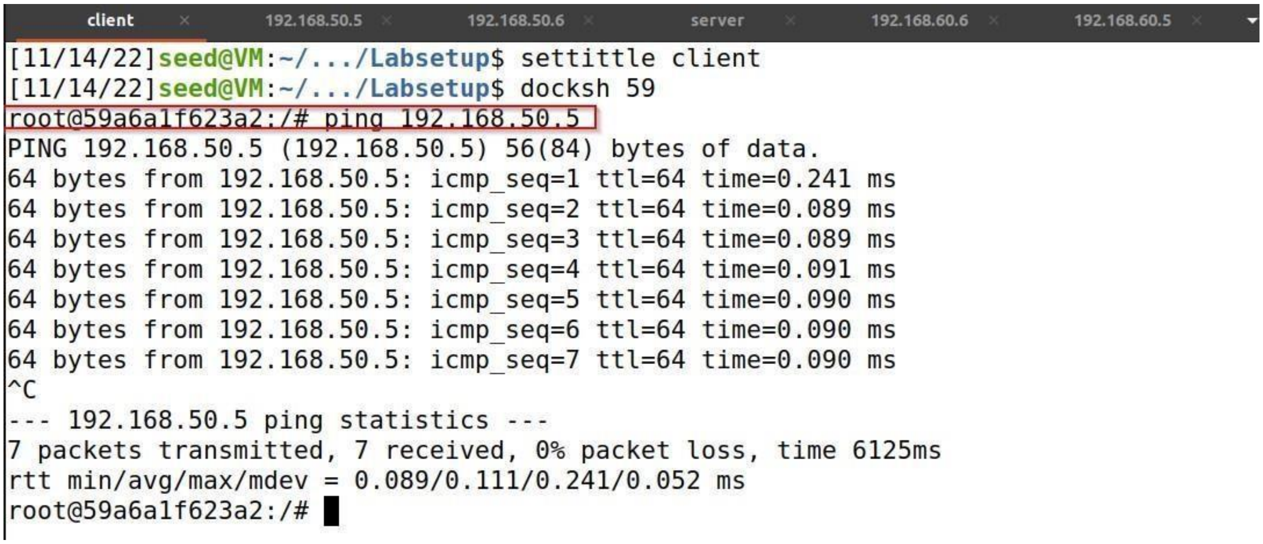


```
^C
[11/14/22]seed@VM:~/.../volumes$ cd ..
[11/14/22]seed@VM:~/.../Labsetup$ docker-compose -f docker-compose2.yml build
HostA uses an image, skipping
HostB uses an image, skipping
VPN_Client uses an image, skipping
Host1 uses an image, skipping
Host2 uses an image, skipping
Router uses an image, skipping
[11/14/22]seed@VM:~/.../Labsetup$ docker-compose -f docker-compose2.yml up
Creating network "net-192.168.50.0" with the default driver
WARNING: Found orphan containers (seed-attacker, user-10.9.0.5, local-dns-server-1
0.9.0.53) for this project. If you removed or renamed this service in your compose
 file, you can run this command with the --remove-orphans flag to clean it up.
host-192.168.60.5 is up-to-date
server-router is up-to-date
Recreating client-10.9.0.5 ...
Recreating client-10.9.0.5 ... done
Creating host-192.168.50.6 ... done
Creating host-192.168.50.5 ... done
```



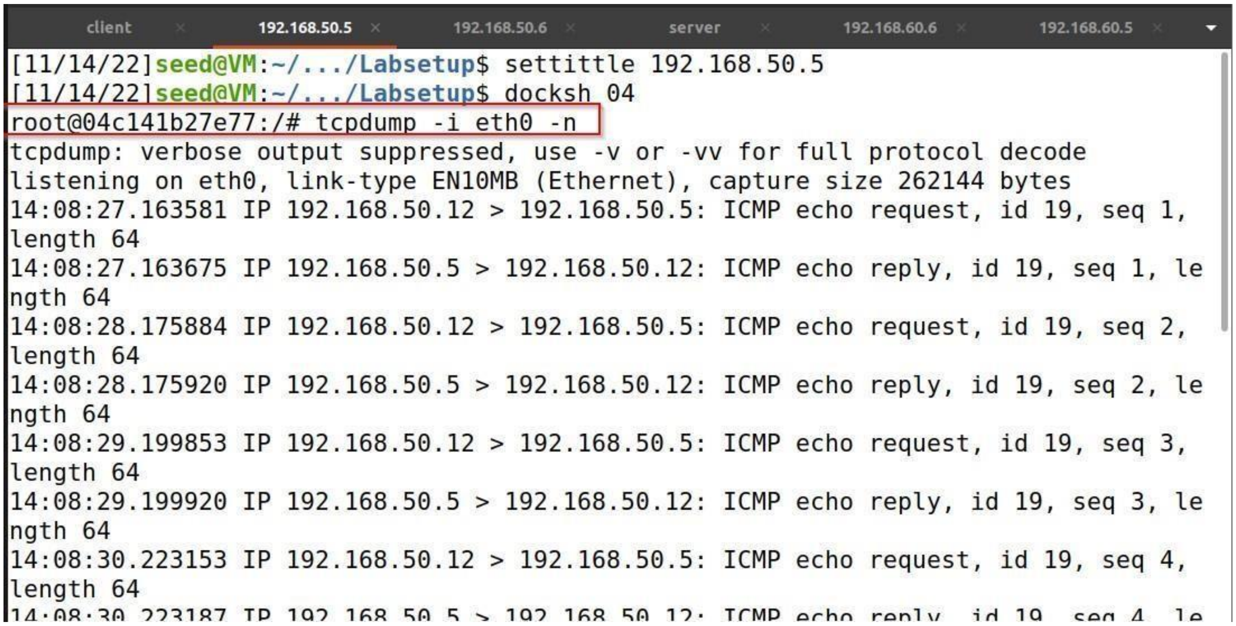
```
Host1 uses an image, skipping
Host2 uses an image, skipping
Router uses an image, skipping
[11/14/22]seed@VM:~/.../Labsetup$ docker-compose -f docker-compose2.yml up
Creating network "net-192.168.50.0" with the default driver
WARNING: Found orphan containers (seed-attacker, user-10.9.0.5, local-dns-server-1
0.9.0.53) for this project. If you removed or renamed this service in your compose
 file, you can run this command with the --remove-orphans flag to clean it up.
host-192.168.60.5 is up-to-date
server-router is up-to-date
Recreating client-10.9.0.5 ...
Recreating client-10.9.0.5 ... done
Creating host-192.168.50.6 ... done
Creating host-192.168.50.5 ... done
Attaching to host-192.168.60.5, server-router, host-192.168.60.6, host-192.168.50.
5, host-192.168.50.6, client-10.9.0.5
host-192.168.50.5 |  * Starting internet superserver inetd                 [ OK ]
host-192.168.50.6 |  * Starting internet superserver inetd                 [ OK ]
host-192.168.60.6 |  * Starting internet superserver inetd                 [ OK ]
host-192.168.60.5 |  * Starting internet superserver inetd                 [ OK ]
```

Ping 192.168.50.5 from client

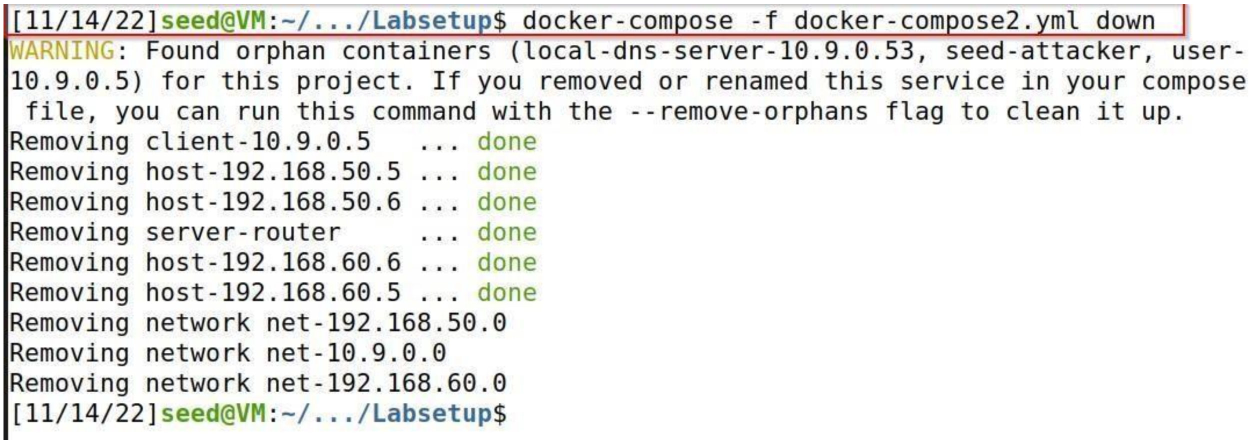


```
[11/14/22]seed@VM:~/.../Labsetup$ settittle client
[11/14/22]seed@VM:~/.../Labsetup$ docksh 59
root@59a6a1f623a2:/# ping 192.168.50.5
PING 192.168.50.5 (192.168.50.5) 56(84) bytes of data.
64 bytes from 192.168.50.5: icmp_seq=1 ttl=64 time=0.241 ms
64 bytes from 192.168.50.5: icmp_seq=2 ttl=64 time=0.089 ms
64 bytes from 192.168.50.5: icmp_seq=3 ttl=64 time=0.089 ms
64 bytes from 192.168.50.5: icmp_seq=4 ttl=64 time=0.091 ms
64 bytes from 192.168.50.5: icmp_seq=5 ttl=64 time=0.090 ms
64 bytes from 192.168.50.5: icmp_seq=6 ttl=64 time=0.090 ms
64 bytes from 192.168.50.5: icmp_seq=7 ttl=64 time=0.090 ms
^C
--- 192.168.50.5 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6125ms
rtt min/avg/max/mdev = 0.089/0.111/0.241/0.052 ms
root@59a6a1f623a2:/#
```

Successful we are getting icmp packets in 192.168.50.5



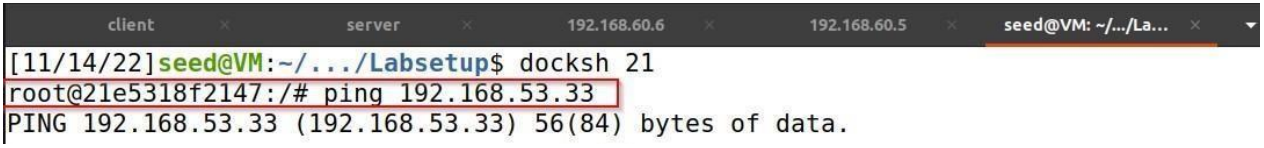
```
[11/14/22]seed@VM:~/.../Labsetup$ settittle 192.168.50.5
[11/14/22]seed@VM:~/.../Labsetup$ docksh 04
root@04c141b27e77:/# tcpdump -i eth0 -n
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
14:08:27.163581 IP 192.168.50.12 > 192.168.50.5: ICMP echo request, id 19, seq 1,
length 64
14:08:27.163675 IP 192.168.50.5 > 192.168.50.12: ICMP echo reply, id 19, seq 1, le
ngth 64
14:08:28.175884 IP 192.168.50.12 > 192.168.50.5: ICMP echo request, id 19, seq 2,
length 64
14:08:28.175920 IP 192.168.50.5 > 192.168.50.12: ICMP echo reply, id 19, seq 2, le
ngth 64
14:08:29.199853 IP 192.168.50.12 > 192.168.50.5: ICMP echo request, id 19, seq 3,
length 64
14:08:29.199920 IP 192.168.50.5 > 192.168.50.12: ICMP echo reply, id 19, seq 3, le
ngth 64
14:08:30.223153 IP 192.168.50.12 > 192.168.50.5: ICMP echo request, id 19, seq 4,
length 64
14:08:30.223187 IP 192.168.50.5 > 192.168.50.12: ICMP echo reply, id 19, seq 4, le
```



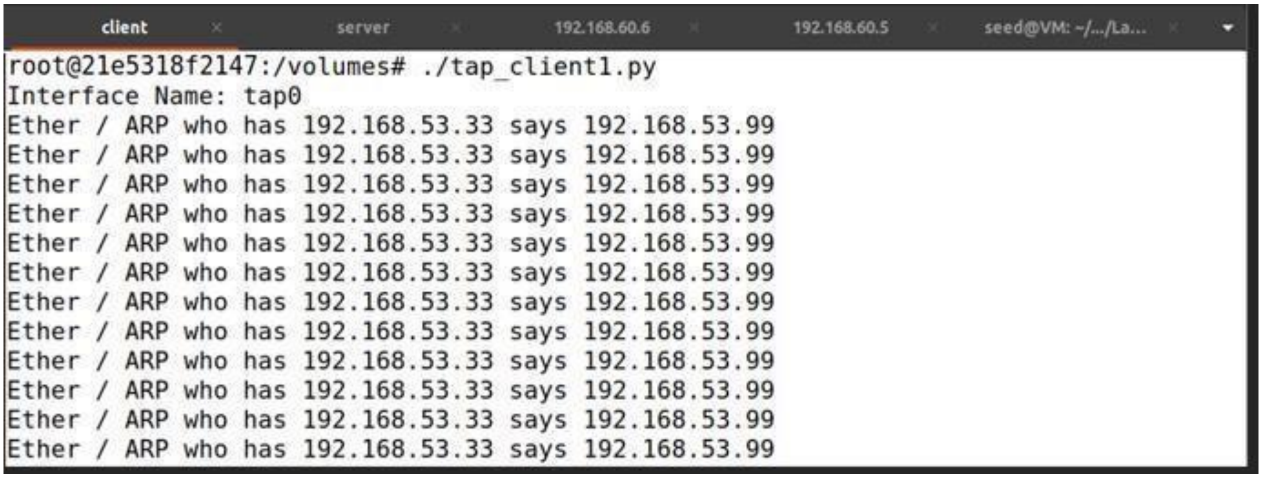
```
[11/14/22]seed@VM:~/.../Labsetup$ docker-compose -f docker-compose2.yml down
WARNING: Found orphan containers (local-dns-server-10.9.0.53, seed-attacker, user-
10.9.0.5) for this project. If you removed or renamed this service in your compose
 file, you can run this command with the --remove-orphans flag to clean it up.
Removing client-10.9.0.5    ... done
Removing host-192.168.50.5 ... done
Removing host-192.168.50.6 ... done
Removing server-router      ... done
Removing host-192.168.60.6 ... done
Removing host-192.168.60.5 ... done
Removing network net-192.168.50.0
Removing network net-10.9.0.0
Removing network net-192.168.60.0
[11/14/22]seed@VM:~/.../Labsetup$
```

Task 9: Experiment with the TAP Interface

Ping 192.168.53.33 from host u

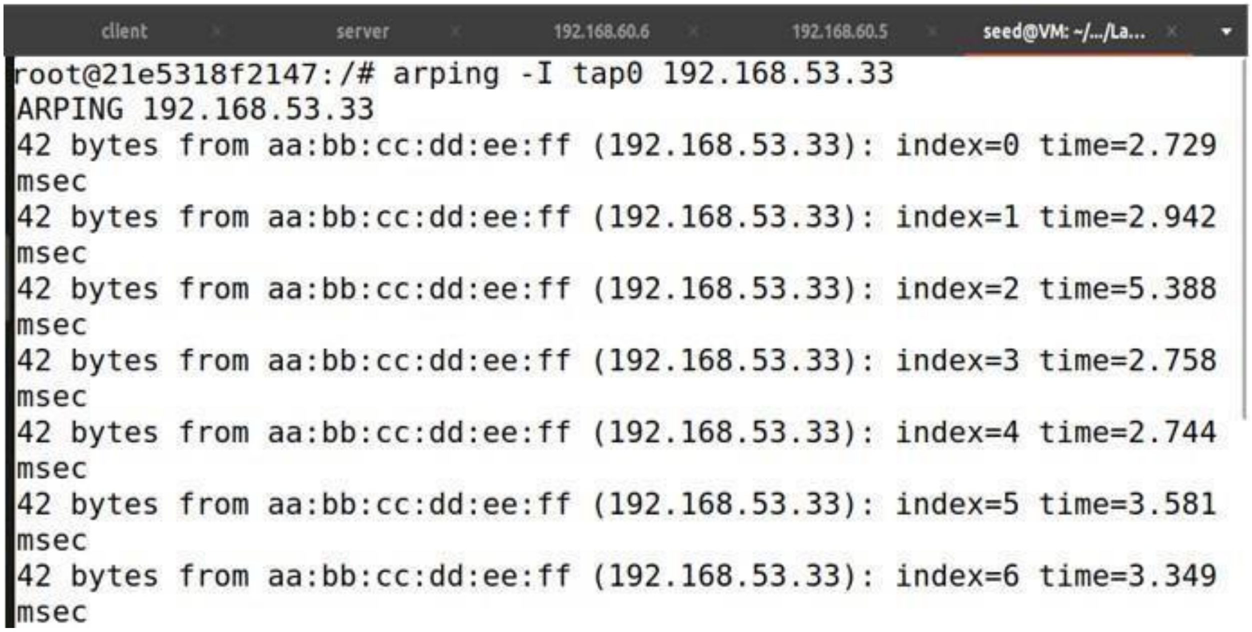


```
[11/14/22]seed@VM:~/.../Labsetup$ docksh 21
root@21e5318f2147:/# ping 192.168.53.33
PING 192.168.53.33 (192.168.53.33) 56(84) bytes of data.
```

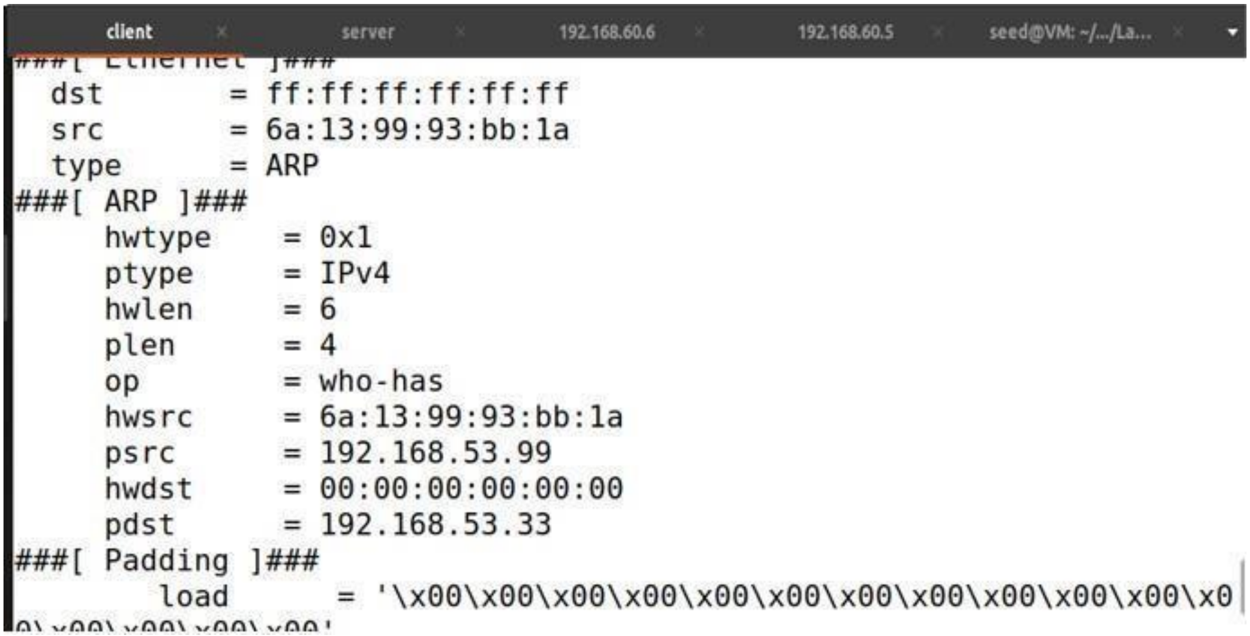


```
root@21e5318f2147:/volumes# ./tap_client1.py
Interface Name: tap0
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
Ether / ARP who has 192.168.53.33 says 192.168.53.99
```

arping -I tap0 192.168.53.33

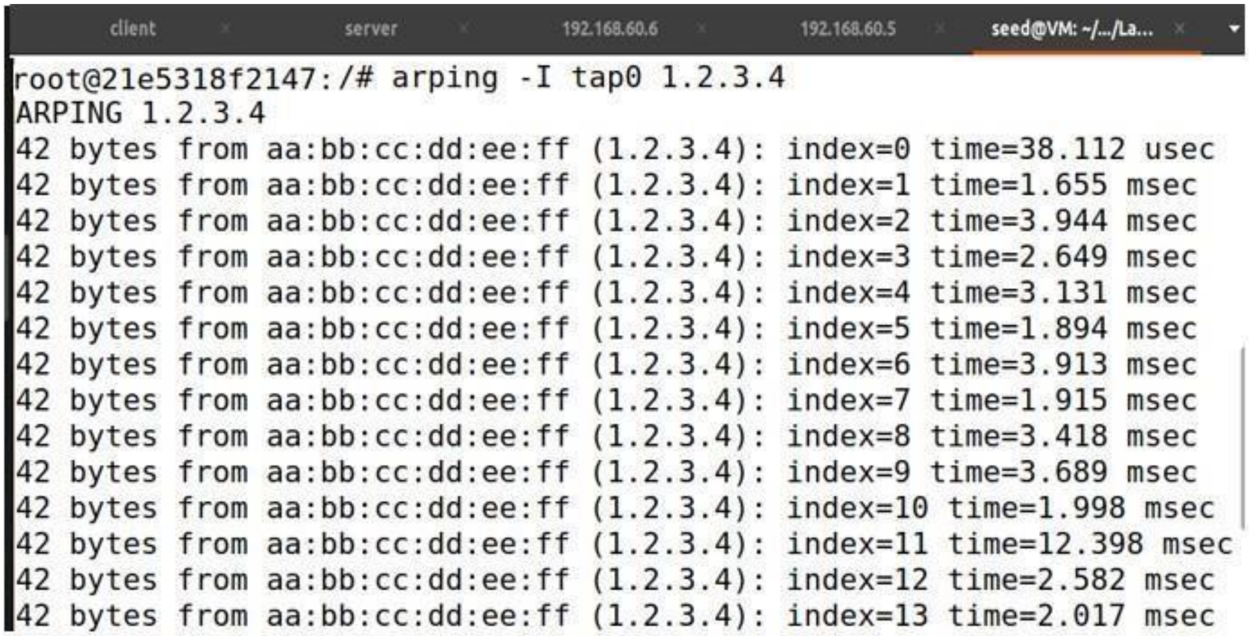


```
root@21e5318f2147:/# arping -I tap0 192.168.53.33
ARPING 192.168.53.33
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=0 time=2.729
msec
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=1 time=2.942
msec
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=2 time=5.388
msec
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=3 time=2.758
msec
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=4 time=2.744
msec
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=5 time=3.581
msec
42 bytes from aa:bb:cc:dd:ee:ff (192.168.53.33): index=6 time=3.349
msec
```

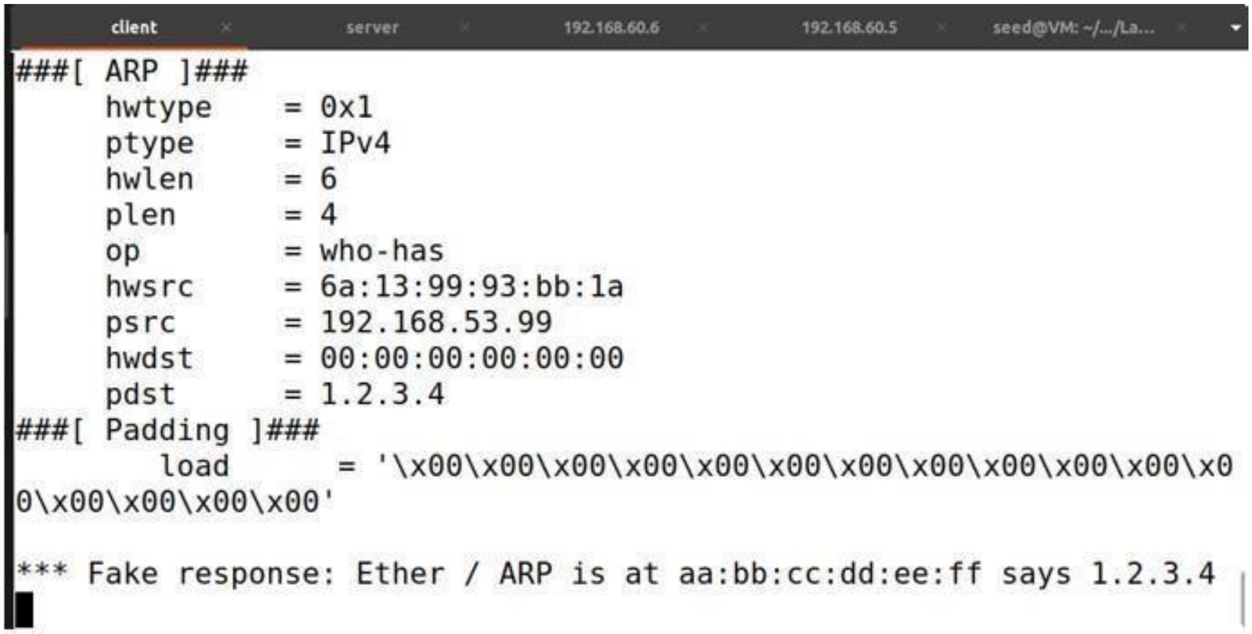


```
###[ Ethernet ]###
  dst       = ff:ff:ff:ff:ff:ff
  src       = 6a:13:99:93:bb:1a
  type      = ARP
###[ ARP ]###
     hwtype    = 0x1
     ptype     = IPv4
     hwlen     = 6
     plen      = 4
     op        = who-has
     hwsrc     = 6a:13:99:93:bb:1a
     psrc      = 192.168.53.99
     hwdst     = 00:00:00:00:00:00
     pdst      = 192.168.53.33
###[ Padding ]###
        load      = '\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00'
```

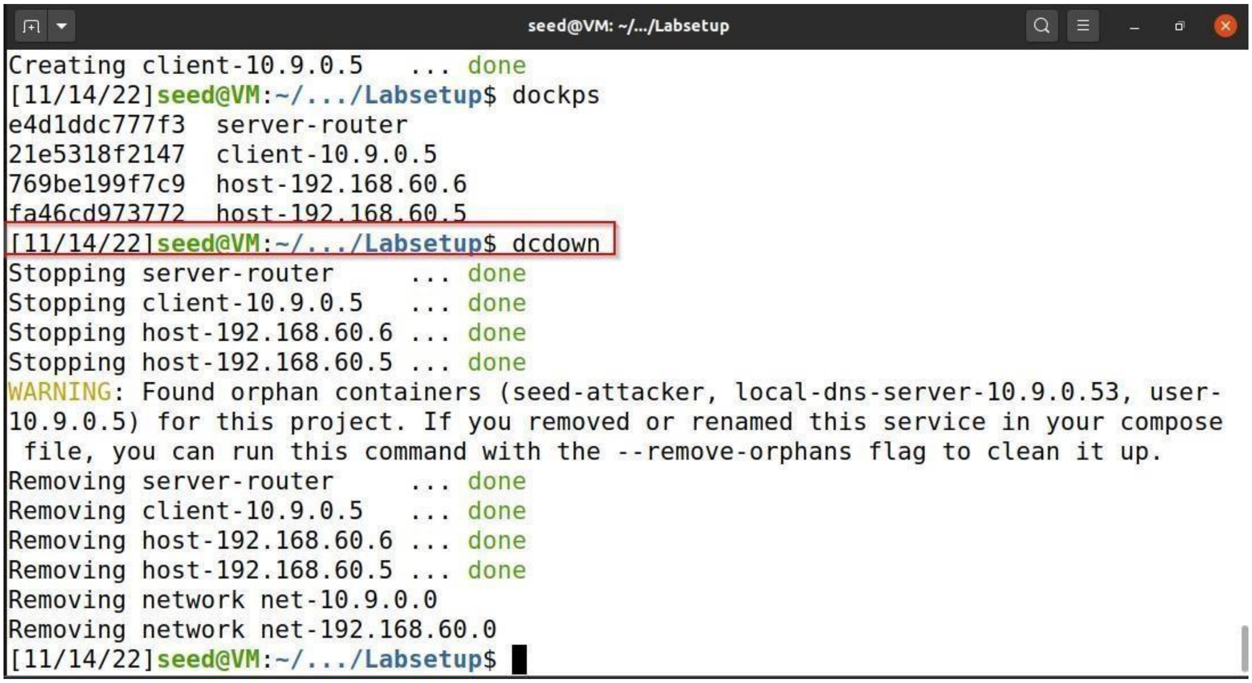
arping -I tap0 1.2.3.4



```
root@21e5318f2147:/# arping -I tap0 1.2.3.4
ARPING 1.2.3.4
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=0 time=38.112 usec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=1 time=1.655 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=2 time=3.944 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=3 time=2.649 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=4 time=3.131 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=5 time=1.894 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=6 time=3.913 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=7 time=1.915 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=8 time=3.418 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=9 time=3.689 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=10 time=1.998 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=11 time=12.398 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=12 time=2.582 msec
42 bytes from aa:bb:cc:dd:ee:ff (1.2.3.4): index=13 time=2.017 msec
```



```
###[ ARP ]###
     hwtype    = 0x1
     ptype     = IPv4
     hwlen     = 6
     plen      = 4
     op        = who-has
     hwsrc     = 6a:13:99:93:bb:1a
     psrc      = 192.168.53.99
     hwdst     = 00:00:00:00:00:00
     pdst      = 1.2.3.4
###[ Padding ]###
        load      = '\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00'

*** Fake response: Ether / ARP is at aa:bb:cc:dd:ee:ff says 1.2.3.4
```



```
Creating client-10.9.0.5   ... done
[11/14/22]seed@VM:~/.../Labsetup$ dockps
e4d1ddc777f3  server-router
21e5318f2147  client-10.9.0.5
769be199f7c9  host-192.168.60.6
fa46cd973772  host-192.168.60.5
[11/14/22]seed@VM:~/.../Labsetup$ dcdown
Stopping server-router     ... done
Stopping client-10.9.0.5   ... done
Stopping host-192.168.60.6 ... done
Stopping host-192.168.60.5 ... done
WARNING: Found orphan containers (seed-attacker, local-dns-server-10.9.0.53, user-
10.9.0.5) for this project. If you removed or renamed this service in your compose
 file, you can run this command with the --remove-orphans flag to clean it up.
Removing server-router     ... done
Removing client-10.9.0.5   ... done
Removing host-192.168.60.6 ... done
Removing host-192.168.60.5 ... done
Removing network net-10.9.0.0
Removing network net-192.168.60.0
[11/14/22]seed@VM:~/.../Labsetup$ 
```