SEED LAB7

57117111 蒋涛

VPN Tunneling Lab

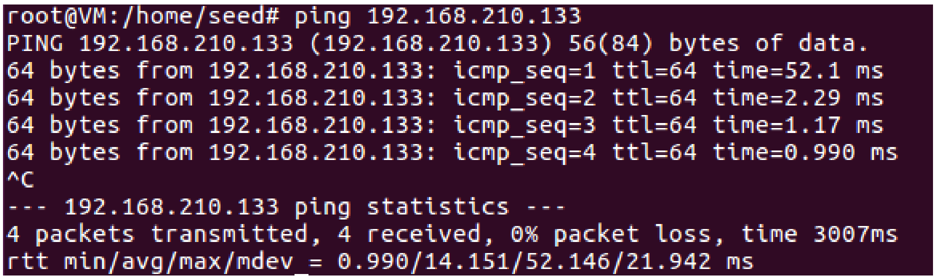
Task1: Network Setup

- Host U: 192.168.210.132

  VPN Server: 192.168.60.1 (192.168.210.133)

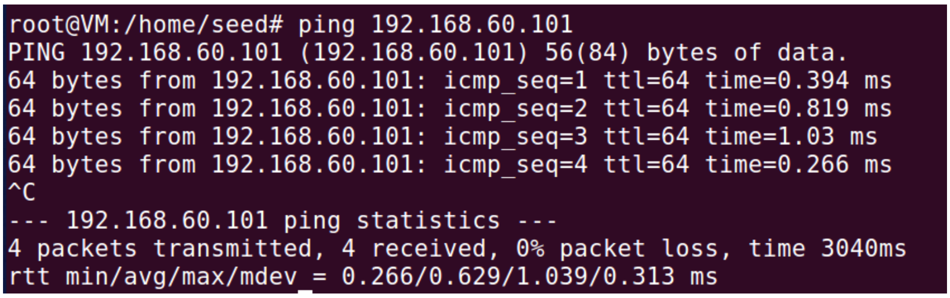
  Host V: 192.168.60.101

- Host U can communicate with VPN Server



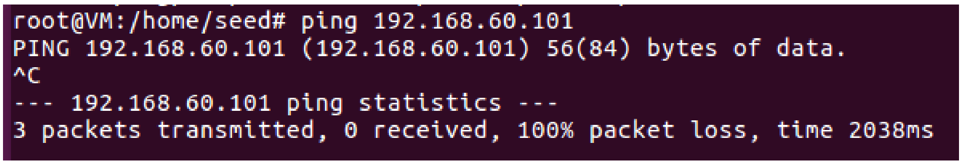
```
root@VM:/home/seed# ping 192.168.210.133
PING 192.168.210.133 (192.168.210.133) 56(84) bytes of data.
64 bytes from 192.168.210.133: icmp_seq=1 ttl=64 time=52.1 ms
64 bytes from 192.168.210.133: icmp_seq=2 ttl=64 time=2.29 ms
64 bytes from 192.168.210.133: icmp_seq=3 ttl=64 time=1.17 ms
64 bytes from 192.168.210.133: icmp_seq=4 ttl=64 time=0.990 ms
^C
--- 192.168.210.133 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3007ms
rtt min/avg/max/mdev = 0.990/14.151/52.146/21.942 ms
```

- VPN Server can communicate with Host V



```
root@VM:/home/seed# ping 192.168.60.101
PING 192.168.60.101 (192.168.60.101) 56(84) bytes of data.
64 bytes from 192.168.60.101: icmp_seq=1 ttl=64 time=0.394 ms
64 bytes from 192.168.60.101: icmp_seq=2 ttl=64 time=0.819 ms
64 bytes from 192.168.60.101: icmp_seq=3 ttl=64 time=1.03 ms
64 bytes from 192.168.60.101: icmp_seq=4 ttl=64 time=0.266 ms
^C
--- 192.168.60.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3040ms
rtt min/avg/max/mdev = 0.266/0.629/1.039/0.313 ms
```

- Host U can’t communicate with Host V

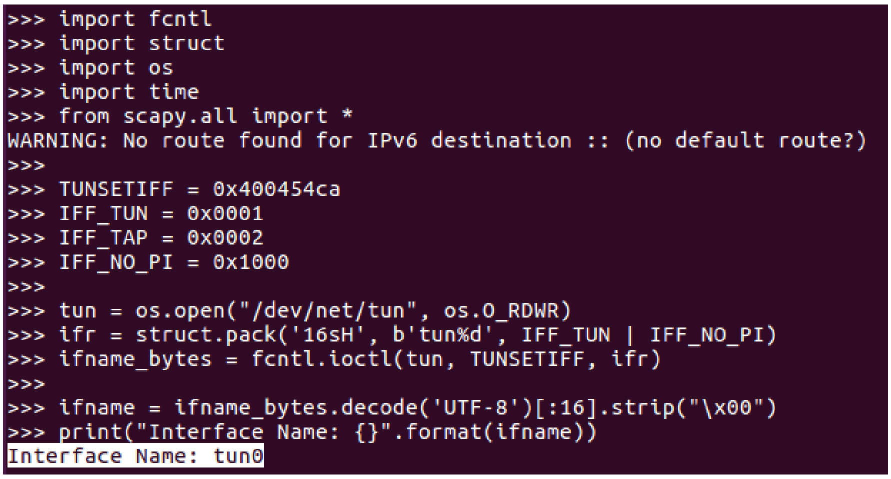


```
root@VM:/home/seed# ping 192.168.60.101
PING 192.168.60.101 (192.168.60.101) 56(84) bytes of data.
^C
--- 192.168.60.101 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2038ms
```

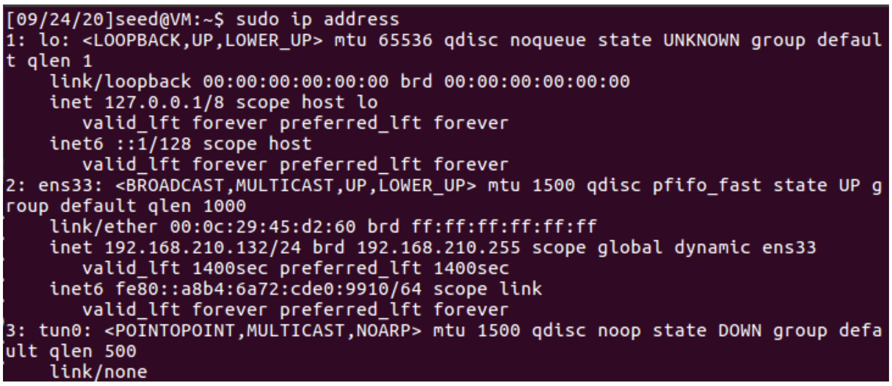
Task2: Create and Configure TUN Interface

Task 2.a: Name of the Interface

- Run the `tun.py` program on Host U and find `tun0` interface

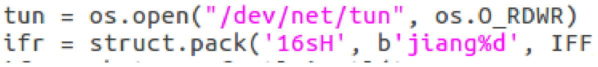


```
>>> import fcntl
>>> import struct
>>> import os
>>> import time
>>> from scapy.all import *
WARNING: No route found for IPv6 destination :: (no default route?)
>>>
>>> TUNSETIFF = 0x400454ca
>>> IFF_TUN = 0x0001
>>> IFF_TAP = 0x0002
>>> IFF_NO_PI = 0x1000
>>>
>>> tun = os.open("/dev/net/tun", os.O_RDWR)
>>> ifr = struct.pack('16sH', b'tun%d', IFF_TUN | IFF_NO_PI)
>>> ifname_bytes = fcntl.ioctl(tun, TUNSETIFF, ifr)
>>>
>>> ifname = ifname_bytes.decode('UTF-8')[:16].strip("\x00")
>>> print("Interface Name: {}".format(ifname))
Interface Name: tun0
```

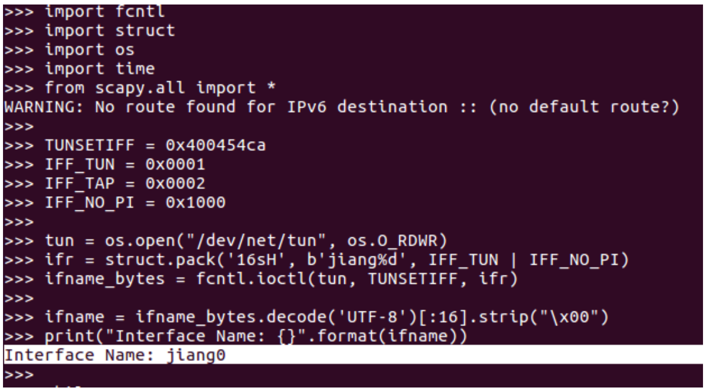


```
[09/24/20]seed@VM:~$ sudo ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP g
roup default qlen 1000
    link/ether 00:0c:29:45:d2:60 brd ff:ff:ff:ff:ff:ff
    inet 192.168.210.132/24 brd 192.168.210.255 scope global dynamic ens33
       valid_lft 1400sec preferred_lft 1400sec
    inet6 fe80::a8b4:6a72:cde0:9910/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group defa
ult qlen 500
    link/none
```

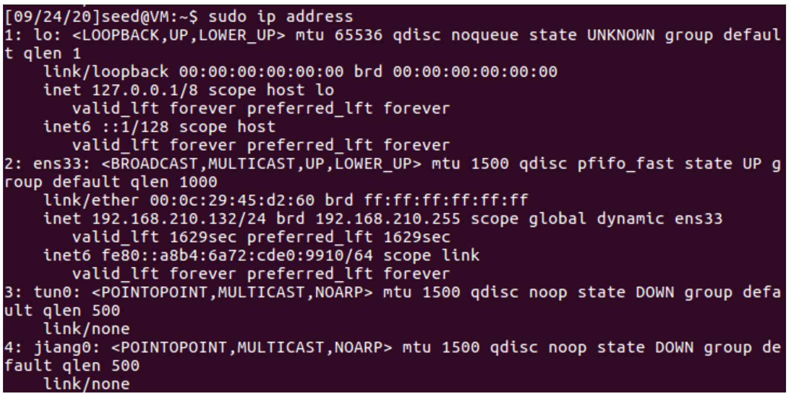
- Modify the prefix of the interface name



```
tun = os.open("/dev/net/tun", os.O_RDWR)
ifr = struct.pack('16sH', b'jiang%d', IFF
```



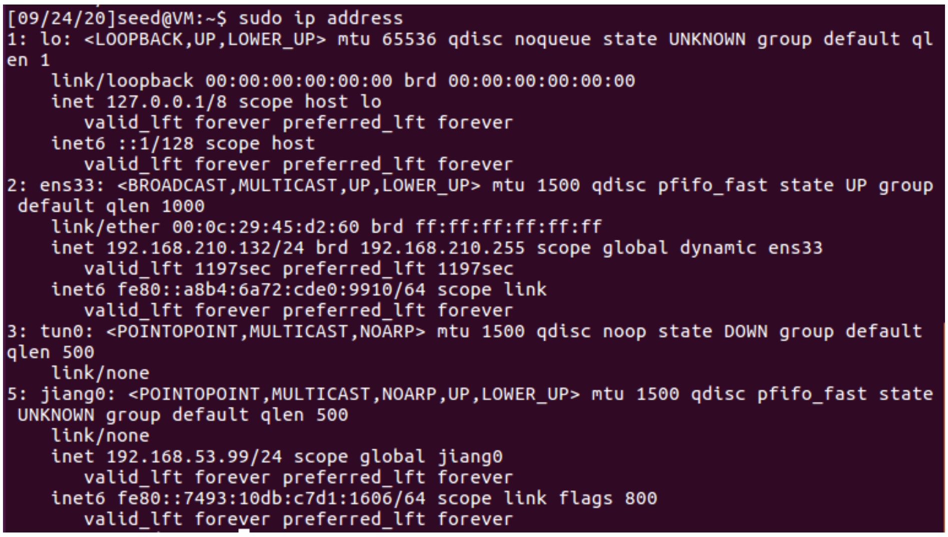
```
>>> import fcntl
>>> import struct
>>> import os
>>> import time
>>> from scapy.all import *
WARNING: No route found for IPv6 destination :: (no default route?)
>>>
>>> TUNSETIFF = 0x400454ca
>>> IFF_TUN = 0x0001
>>> IFF_TAP = 0x0002
>>> IFF_NO_PI = 0x1000
>>>
>>> tun = os.open("/dev/net/tun", os.O_RDWR)
>>> ifr = struct.pack('16sH', b'jiang%d', IFF_TUN | IFF_NO_PI)
>>> ifname_bytes = fcntl.ioctl(tun, TUNSETIFF, ifr)
>>>
>>> ifname = ifname_bytes.decode('UTF-8')[:16].strip("\x00")
>>> print("Interface Name: {}".format(ifname))
Interface Name: jiang0
>>>
```



```
[09/24/20]seed@VM:~$ sudo ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1
    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
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP g
roup default qlen 1000
    link/ether 00:0c:29:45:d2:60 brd ff:ff:ff:ff:ff:ff
    inet 192.168.210.132/24 brd 192.168.210.255 scope global dynamic ens33
       valid_lft 1629sec preferred_lft 1629sec
    inet6 fe80::a8b4:6a72:cde0:9910/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group defa
ult qlen 500
    link/none
4: jiang0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group de
fault qlen 500
    link/none
```

Task 2.b: Set up the TUN Interface

- The number ahead of `jiang0` changes to 5. `UP` and `LOWER_UP` are added to flag and 192.168.53.99/24 added to `inet`.

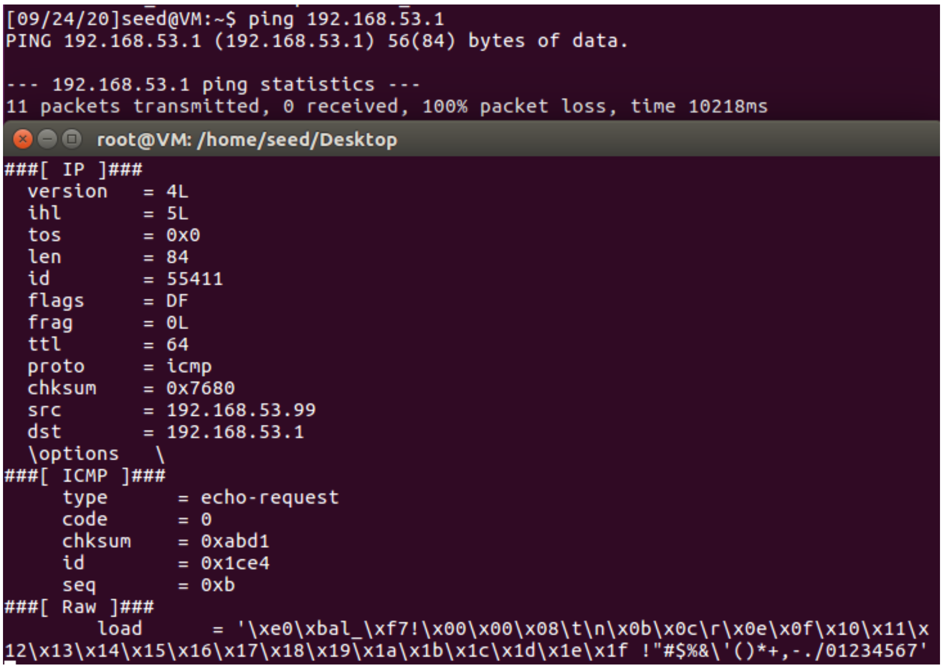


```
[09/24/20]seed@VM:~$ sudo ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default ql
en 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group
 default qlen 1000
    link/ether 00:0c:29:45:d2:60 brd ff:ff:ff:ff:ff:ff
    inet 192.168.210.132/24 brd 192.168.210.255 scope global dynamic ens33
       valid_lft 1197sec preferred_lft 1197sec
    inet6 fe80::a8b4:6a72:cde0:9910/64 scope link
       valid_lft forever preferred_lft forever
3: tun0: <POINTOPOINT,MULTICAST,NOARP> mtu 1500 qdisc noop state DOWN group default
qlen 500
    link/none
5: jiang0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state
 UNKNOWN group default qlen 500
    link/none
    inet 192.168.53.99/24 scope global jiang0
       valid_lft forever preferred_lft forever
    inet6 fe80::7493:10db:c7d1:1606/64 scope link flags 800
       valid_lft forever preferred_lft forever
```

Task 2.c: Read from the TUN Interface

- Modify the code. `ping` 192.168.53.1 and 192.168.60.1

  Because `jiang0` is in the subnet 192.168.53.99/24, the ICMP request is sent out through interface `jiang0` while not in the subnet 192.168.60.0/24, nothing is printed out by `tun.py`



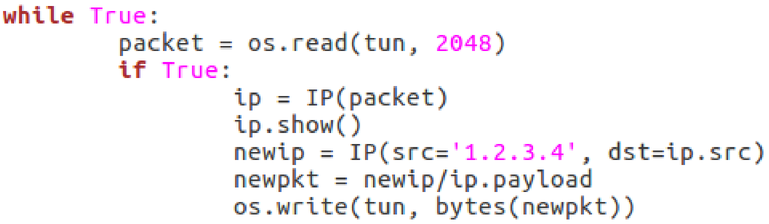
```
[09/24/20]seed@VM:~$ ping 192.168.53.1
PING 192.168.53.1 (192.168.53.1) 56(84) bytes of data.

--- 192.168.53.1 ping statistics ---
11 packets transmitted, 0 received, 100% packet loss, time 10218ms
```

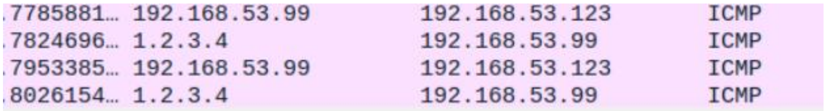
```
###[ IP ]###
  version   = 4L
  ihl       = 5L
  tos       = 0x0
  len       = 84
  id        = 55411
  flags     = DF
  frag      = 0L
  ttl       = 64
  proto     = icmp
  chksum    = 0x7680
  src       = 192.168.53.99
  dst       = 192.168.53.1
  \options   \
###[ ICMP ]###
     type      = echo-request
     code      = 0
     chksum    = 0xabd1
     id        = 0x1ce4
     seq       = 0xb
###[ Raw ]###
        load      = '\xe0\xbal_\xf7!\x00\x00\x08\t\n\x0b\x0c\r\x0e\x0f\x10\x11\x
12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f !"#$%&\'()*+,-./01234567'
```

Task 2.d: Write to the TUN Interface

- Write an IP packet to the interface

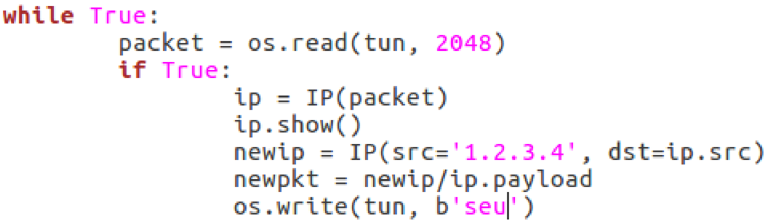


```
while True:
        packet = os.read(tun, 2048)
        if True:
                ip = IP(packet)
                ip.show()
                newip = IP(src='1.2.3.4', dst=ip.src)
                newpkt = newip/ip.payload
                os.write(tun, bytes(newpkt))
```

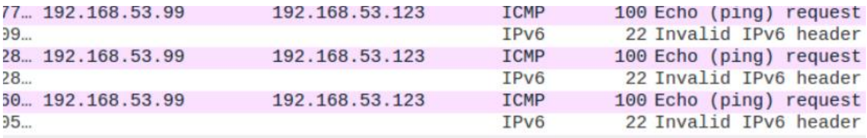


```
.7785881… 192.168.53.99        192.168.53.123       ICMP
.7824696… 1.2.3.4              192.168.53.99        ICMP
.7953385… 192.168.53.99        192.168.53.123       ICMP
.8026154… 1.2.3.4              192.168.53.99        ICMP
```

- Write arbitrary data to the interface



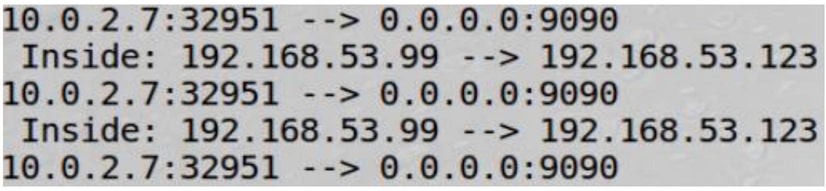
```
while True:
        packet = os.read(tun, 2048)
        if True:
                ip = IP(packet)
                ip.show()
                newip = IP(src='1.2.3.4', dst=ip.src)
                newpkt = newip/ip.payload
                os.write(tun, b'seu')
```



```
77… 192.168.53.99        192.168.53.123       ICMP      100 Echo (ping) request
99…                                           IPv6       22 Invalid IPv6 header
28… 192.168.53.99        192.168.53.123       ICMP      100 Echo (ping) request
28…                                           IPv6       22 Invalid IPv6 header
50… 192.168.53.99        192.168.53.123       ICMP      100 Echo (ping) request
95…                                           IPv6       22 Invalid IPv6 header
```

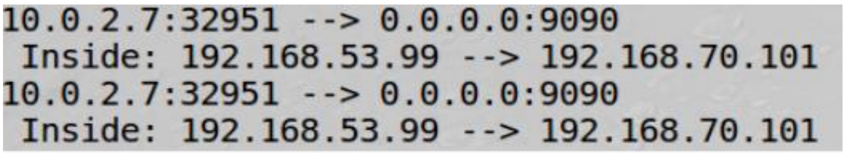
Task3: Send the IP Packet to VPN Server Through a Tunnel

- `ping` 192.168.53.123 on Host U



```
10.0.2.7:32951 --> 0.0.0.0:9090
 Inside: 192.168.53.99 --> 192.168.53.123
10.0.2.7:32951 --> 0.0.0.0:9090
 Inside: 192.168.53.99 --> 192.168.53.123
10.0.2.7:32951 --> 0.0.0.0:9090
```

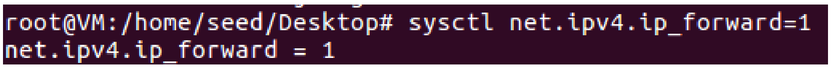
- Add route to the interface and `ping` 192.168.70.101



```
10.0.2.7:32951 --> 0.0.0.0:9090
 Inside: 192.168.53.99 --> 192.168.70.101
10.0.2.7:32951 --> 0.0.0.0:9090
 Inside: 192.168.53.99 --> 192.168.70.101
```

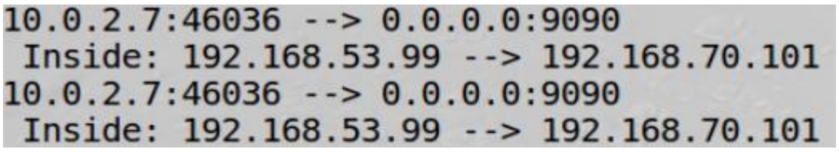
Task4: Set Up the VPN Server

- Enable the IP forwarding

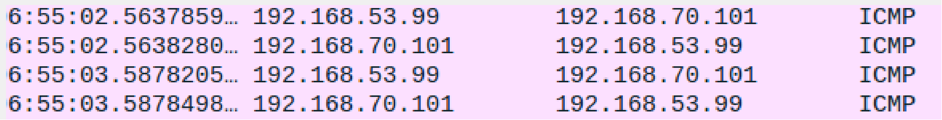


```
root@VM:/home/seed/Desktop# sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
```

- Modify `tun_server.py` and `ping` Host V on Host U



```
10.0.2.7:46036 --> 0.0.0.0:9090
 Inside: 192.168.53.99 --> 192.168.70.101
10.0.2.7:46036 --> 0.0.0.0:9090
 Inside: 192.168.53.99 --> 192.168.70.101
```

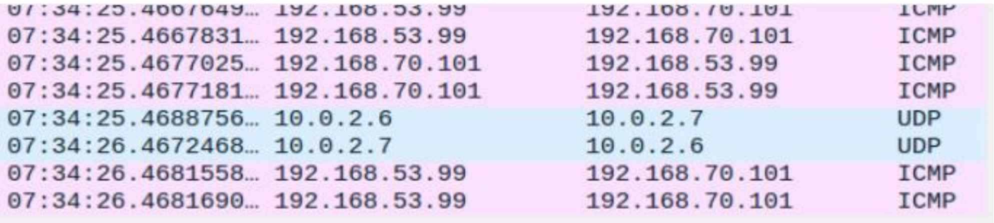


```
6:55:02.5637859… 192.168.53.99      192.168.70.101     ICMP
6:55:02.5638280… 192.168.70.101     192.168.53.99      ICMP
6:55:03.5878205… 192.168.53.99      192.168.70.101     ICMP
6:55:03.5878498… 192.168.70.101     192.168.53.99      ICMP
```

Task5: Handling Traffic in Both Directions

- Modify tun_server.py and tun_client.py

- `ping` Host V on Host U successfully

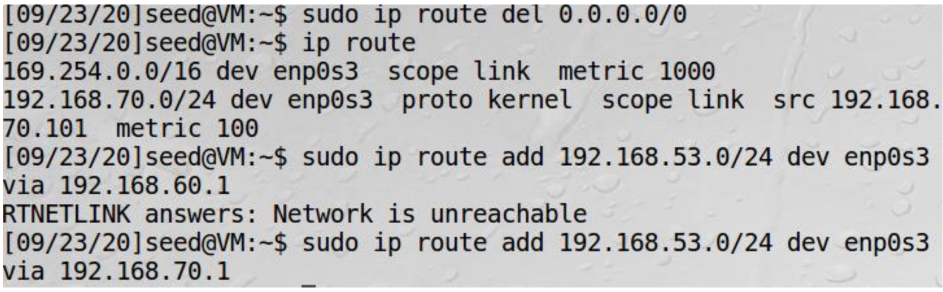


```
07:34:25.4667649… 192.168.53.99       192.168.70.101     ICMP
07:34:25.4667831… 192.168.53.99       192.168.70.101     ICMP
07:34:25.4677025… 192.168.70.101      192.168.53.99      ICMP
07:34:25.4677181… 192.168.70.101      192.168.53.99      ICMP
07:34:25.4688756… 10.0.2.6            10.0.2.7           UDP
07:34:26.4672468… 10.0.2.7            10.0.2.6           UDP
07:34:26.4681558… 192.168.53.99       192.168.70.101     ICMP
07:34:26.4681690… 192.168.53.99       192.168.70.101     ICMP
```

Task6: Tunnel-Breaking Experiment

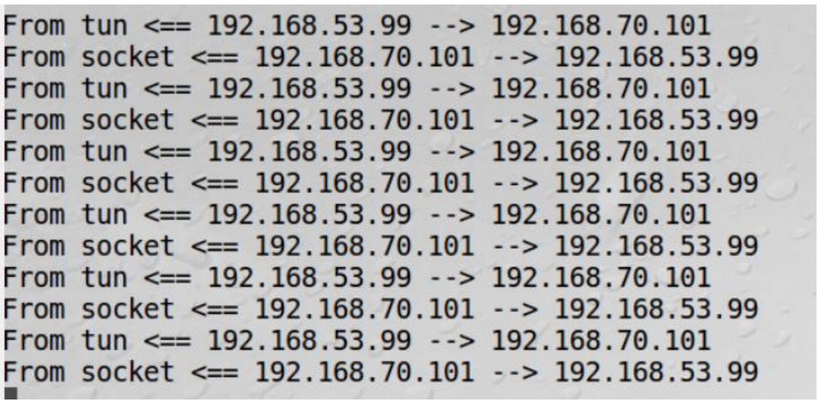
Task7: Routing Experiment on Host V

- Add route on Host V



```
[09/23/20]seed@VM:~$ sudo ip route del 0.0.0.0/0
[09/23/20]seed@VM:~$ ip route
169.254.0.0/16 dev enp0s3  scope link  metric 1000
192.168.70.0/24 dev enp0s3  proto kernel  scope link  src 192.168.
70.101  metric 100
[09/23/20]seed@VM:~$ sudo ip route add 192.168.53.0/24 dev enp0s3
via 192.168.60.1
RTNETLINK answers: Network is unreachable
[09/23/20]seed@VM:~$ sudo ip route add 192.168.53.0/24 dev enp0s3
via 192.168.70.1
```

- `ping` Host V on Host U successfully



```
From tun <== 192.168.53.99 --> 192.168.70.101
From socket <== 192.168.70.101 --> 192.168.53.99
From tun <== 192.168.53.99 --> 192.168.70.101
From socket <== 192.168.70.101 --> 192.168.53.99
From tun <== 192.168.53.99 --> 192.168.70.101
From socket <== 192.168.70.101 --> 192.168.53.99
From tun <== 192.168.53.99 --> 192.168.70.101
From socket <== 192.168.70.101 --> 192.168.53.99
From tun <== 192.168.53.99 --> 192.168.70.101
From socket <== 192.168.70.101 --> 192.168.53.99
From tun <== 192.168.53.99 --> 192.168.70.101
From socket <== 192.168.70.101 --> 192.168.53.99
```