

Firewall Evasion Lab Report: Bypassing Firewalls using VPN

Table of Content

1. Overview -----	01
2. Lab Tasks -----	01
2.1. Task 1: VM Setup -----	01
2.2. Task 2: Set up Firewall -----	06
2.3. Task 3: Bypassing Firewall using VPN -----	08
2.3.1. Step 1: Run VPN Server -----	08
2.3.2. Step 2: Run VPN Client -----	10
2.3.3. Step 3: Set Up Routing on Client and Server VMs -----	13
2.3.4. Step 4: Set Up NAT on Server VM -----	16
2.3.5. Step 5: Bypass firewall with VPN tunnel tun0 -----	17
3. References -----	19

1. Overview

This lab report on Firewall Evasion: Bypassing Firewalls using VPN. Firewalls are a common security measure used to protect computer networks from external threats. However, these firewalls can be bypassed by attackers using various methods, such as using a Virtual Private Network (VPN). This lab report explores the concept of Firewall Evasion using VPN, and provides a practical demonstration of how to bypass firewalls using a VPN. The report provides step by step instructions for setting up a VPN connection and testing its effectiveness in bypassing a firewall.

2. Lab Tasks

2.1. Task 1: VM Setup

VM1 (VPN Client) IP address: 10.0.2.4

VM2 (VPN Server) IP address: 10.0.2.5

Target domain: telehack.com (64.13.139.230)

VirtualBox NAT Network setting as shown in the below figure (Figure. 01).

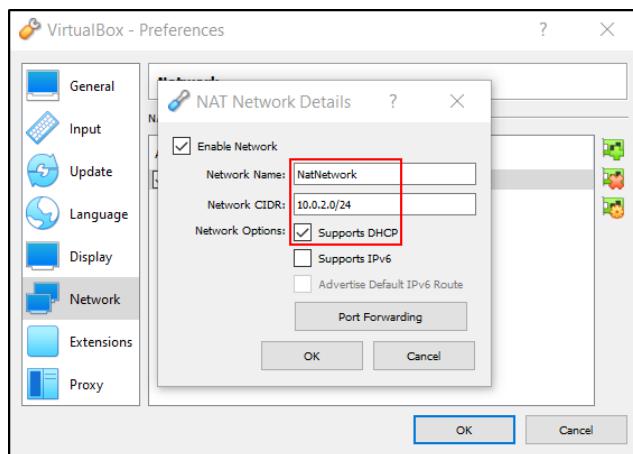


Figure. 01

Client and server network setting as shown in the below figures Figure. 02 and Figure. 03 respectively.

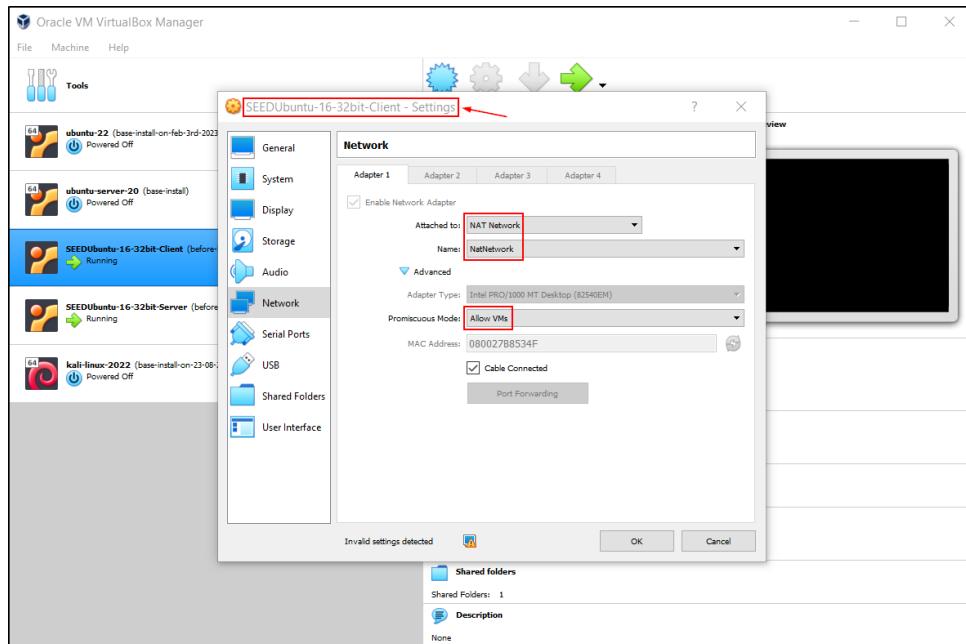


Figure. 02

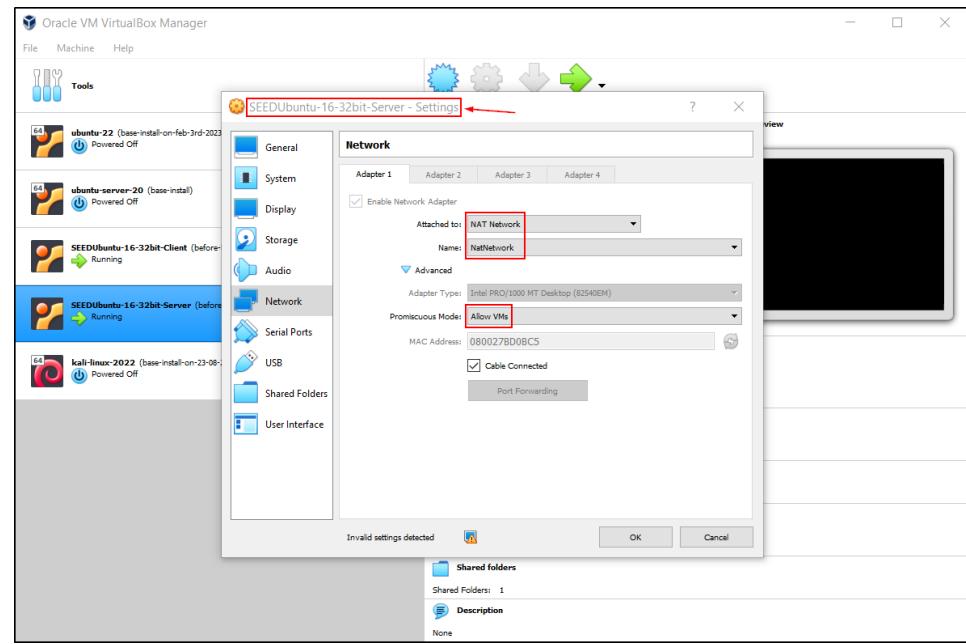
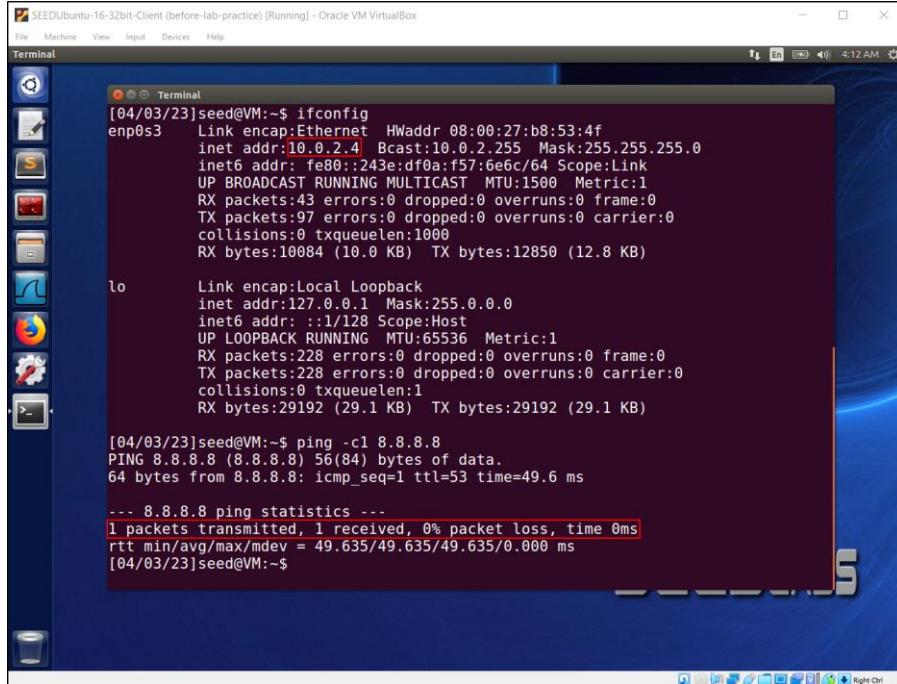


Figure. 03

Client and server ifconfig output and ping to google DNS (8.8.8.8) output as shown in the below figures Figure. 04 and Figure. 05 respectively. Client and server can reach to outside internet without any disruption.



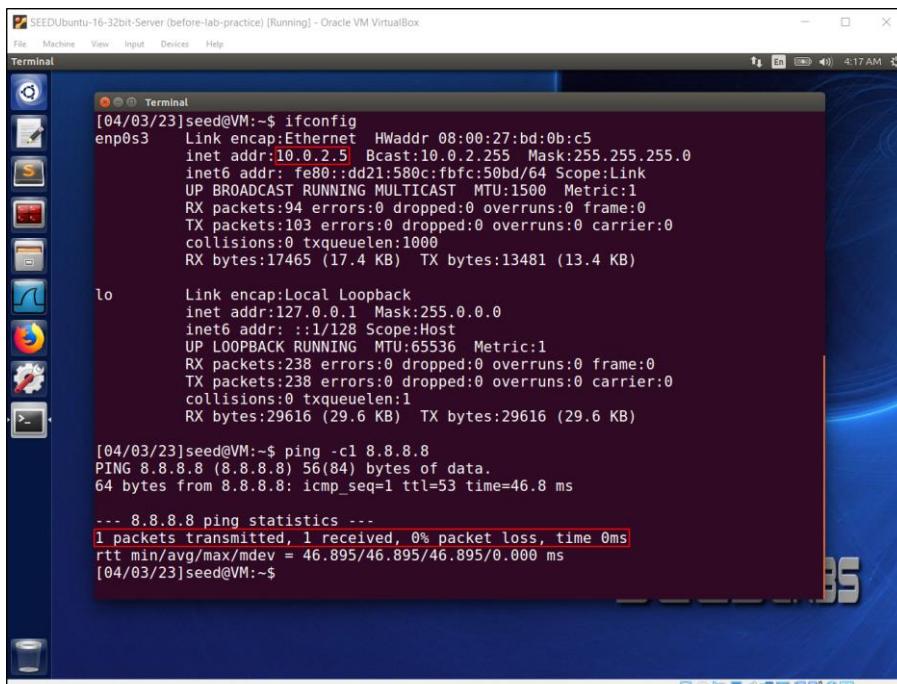
```
[04/03/23]seed@VM:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:b8:53:4f
inet addr:10.0.2.4 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::243e:df0a:f57:6e6c/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:43 errors:0 dropped:0 overruns:0 frame:0
TX packets:97 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:10084 (10.0 KB) TX bytes:12850 (12.8 KB)

lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:228 errors:0 dropped:0 overruns:0 frame:0
TX packets:228 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:29192 (29.1 KB) TX bytes:29192 (29.1 KB)

[04/03/23]seed@VM:~$ ping -c1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=53 time=49.6 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 49.635/49.635/49.635/0.000 ms
[04/03/23]seed@VM:~$
```

Figure. 04



```
[04/03/23]seed@VM:~$ ifconfig
enp0s3 Link encap:Ethernet HWaddr 08:00:27:bd:0b:c5
inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::dd21:580c:fbfc:50bd/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:94 errors:0 dropped:0 overruns:0 frame:0
TX packets:103 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:17465 (17.4 KB) TX bytes:13481 (13.4 KB)

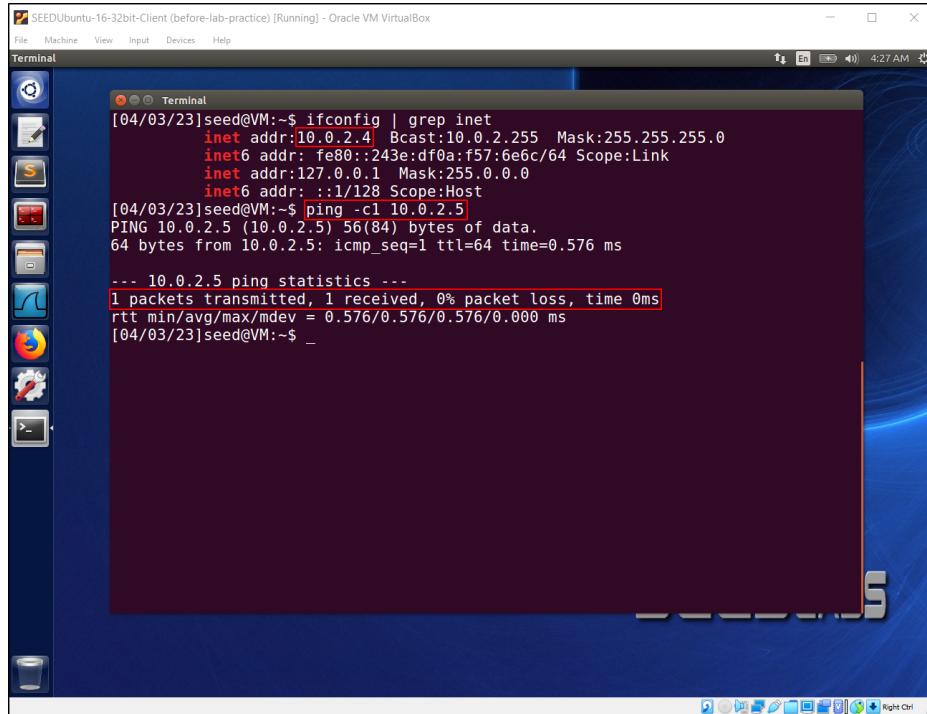
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:238 errors:0 dropped:0 overruns:0 frame:0
TX packets:238 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:29616 (29.6 KB) TX bytes:29616 (29.6 KB)

[04/03/23]seed@VM:~$ ping -c1 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=53 time=46.8 ms

--- 8.8.8.8 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 46.895/46.895/46.895/0.000 ms
[04/03/23]seed@VM:~$
```

Figure. 05

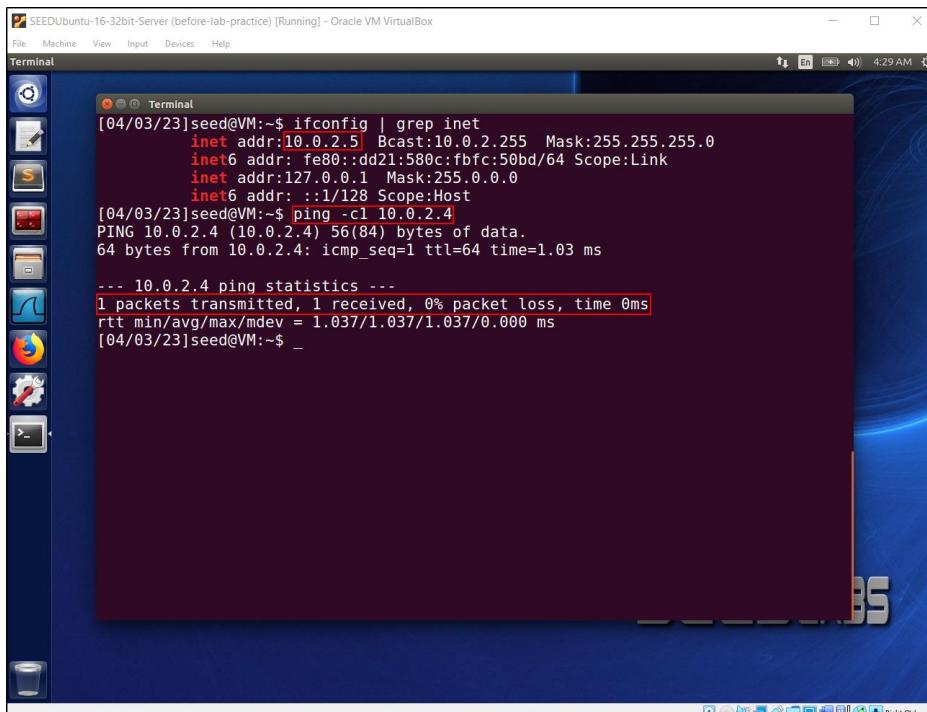
Client and server can communicate (ping) each other as shown in the below figures Figure. 06 and Figure. 07 respectively.



```
[04/03/23]seed@VM:~$ ifconfig | grep inet
    inet addr:10.0.2.4 Bcast:10.0.2.255 Mask:255.255.255.0
    inet6 addr: fe80::243e:df0a:f57:6e6c/64 Scope:Link
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
[04/03/23]seed@VM:~$ ping -c1 10.0.2.5
PING 10.0.2.5 (10.0.2.5) 56(84) bytes of data.
64 bytes from 10.0.2.5: icmp_seq=1 ttl=64 time=0.576 ms

--- 10.0.2.5 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.576/0.576/0.576/0.000 ms
[04/03/23]seed@VM:~$ _
```

Figure. 06



```
[04/03/23]seed@VM:~$ ifconfig | grep inet
    inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
    inet6 addr: fe80::dd21:580c:fbfc:50bd/64 Scope:Link
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
[04/03/23]seed@VM:~$ ping -c1 10.0.2.4
PING 10.0.2.4 (10.0.2.4) 56(84) bytes of data.
64 bytes from 10.0.2.4: icmp_seq=1 ttl=64 time=1.03 ms

--- 10.0.2.4 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.037/1.037/1.037/0.000 ms
[04/03/23]seed@VM:~$ _
```

Figure. 07

Client and server communicate (ping) to telehack.com as shown in the below figures Figure. 08 and Figure. 09 respectively.

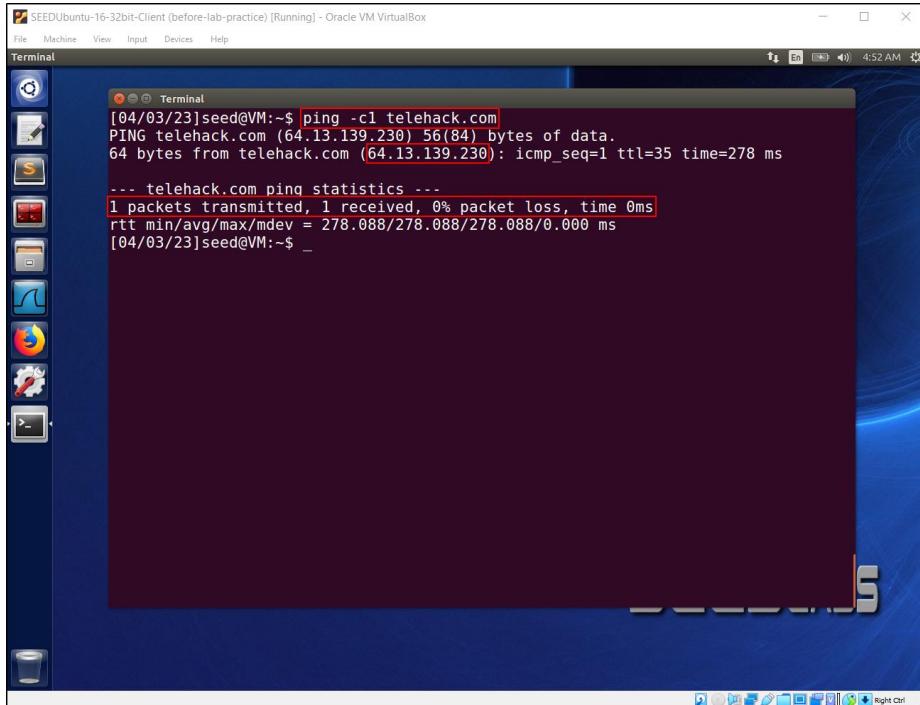


Figure. 08

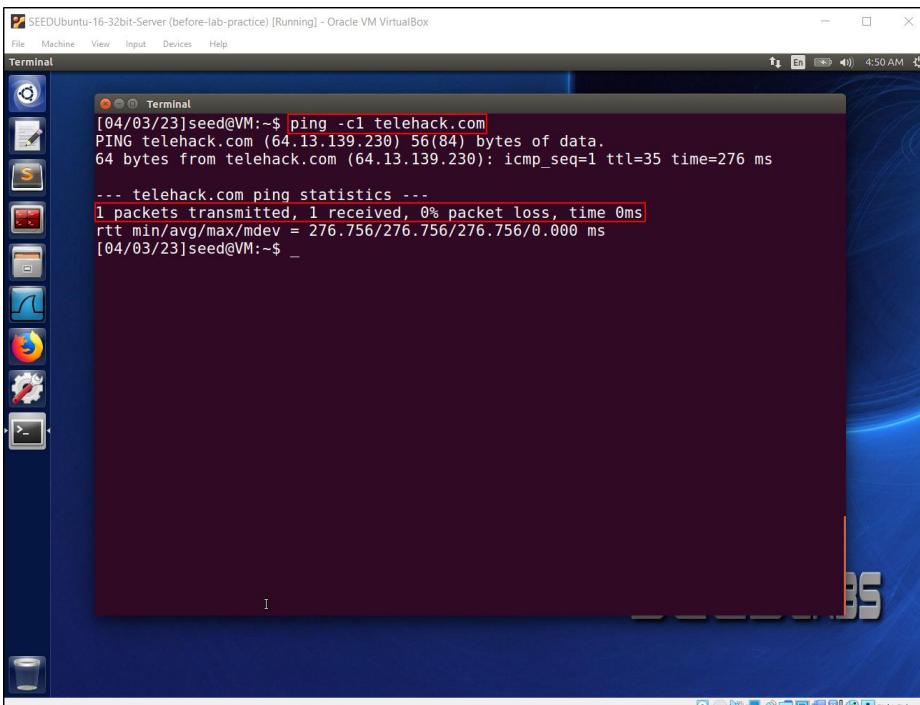


Figure. 09

2.2. Task 2: Set up Firewall

Client connection with target telehack.com via curl, ping and telnet tools with Wireshark packet capture as show in the below figures Figure. 10 and Figure. 11.

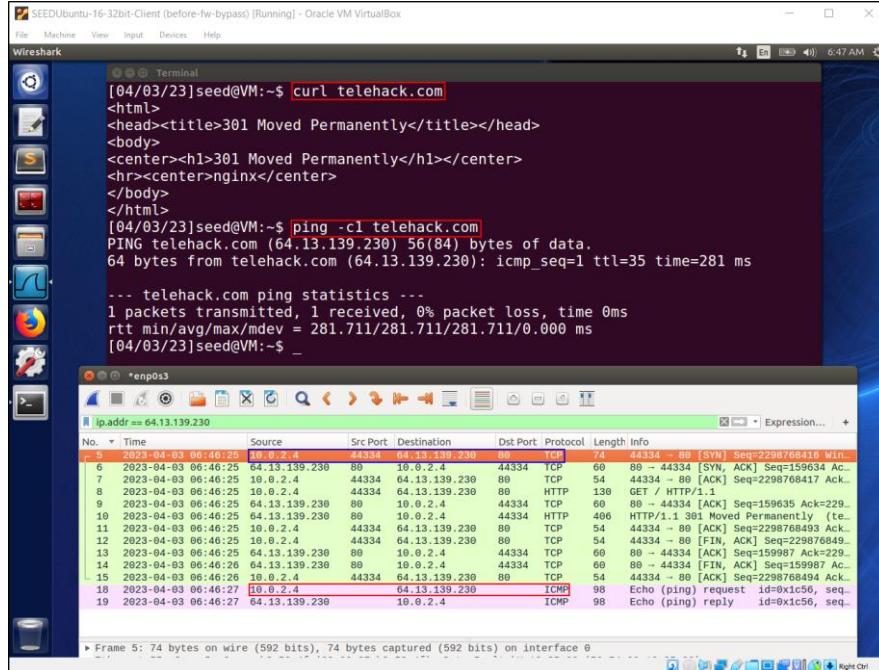


Figure. 10

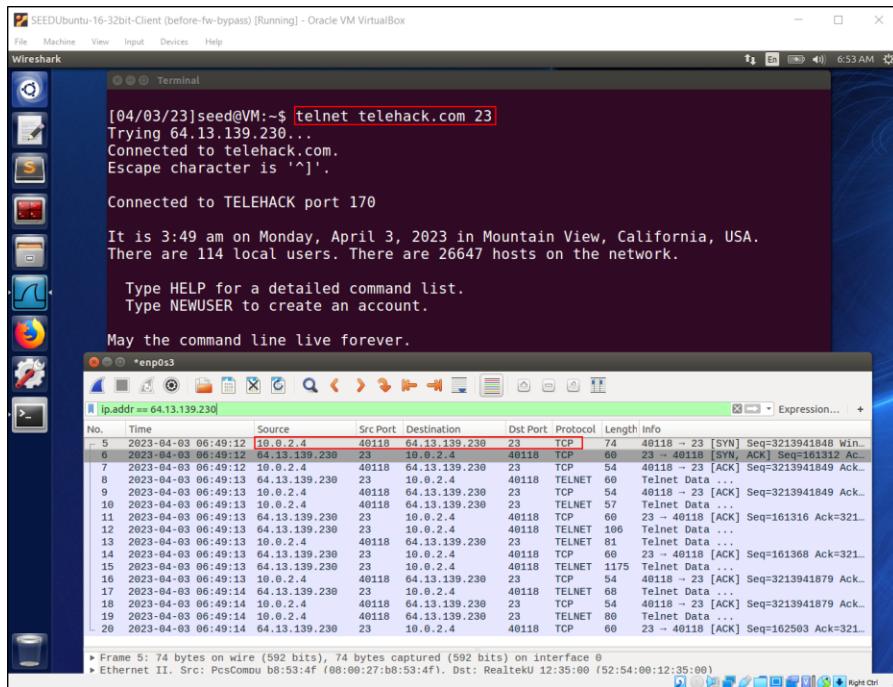


Figure. 11

Ufw tool used to set firewall rules as deny from client to access target domain telehack.com and try to connect via ping to target but 100% packet loss as shown below.

```
[04/03/23]seed@VM:~$ sudo ufw status
Status: inactive
[04/03/23]seed@VM:~$ sudo ufw enable
Firewall is active and enabled on system startup
[04/03/23]seed@VM:~$ sudo ufw status
Status: active
[04/03/23]seed@VM:~$ sudo ufw deny out on enp0s3 to 64.13.139.230
Rule added
[04/03/23]seed@VM:~$ sudo ufw status
Status: active
To           Action      From
--           ----      ---
64.13.139.230    DENY OUT  Anywhere on enp0s3

[04/03/23]seed@VM:~$ ping -c1 telehack.com
PING telehack.com (64.13.139.230) 56(84) bytes of data.
ping: sendmsg: Operation not permitted

... telehack.com ping statistics ...
1 packets transmitted, 0 received, 100% packet loss, time 0ms
```

A red box highlights the line "100% packet loss" in the ping statistics output. An arrow points from this box to a red annotation "100% packetloss" located above the terminal window.

Figure. 12

Curl command used to check whether telehack.com can accessible from client or not. But network is unreachable as shown below.

```
[04/03/23]seed@VM:~$ sudo ufw status
Status: active
To           Action      From
--           ----      ---
64.13.139.230    DENY OUT  Anywhere on enp0s3

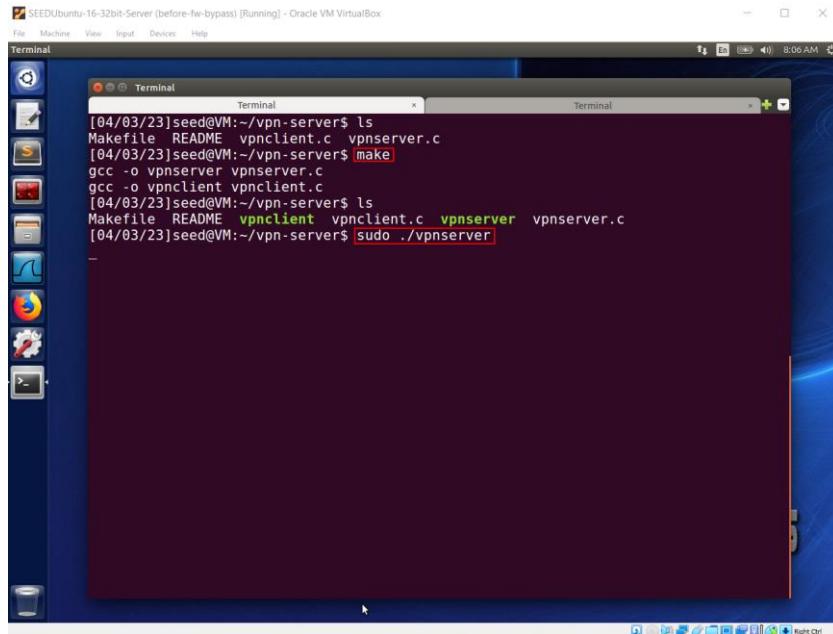
[04/03/23]seed@VM:~$ curl telehack.com -v
* Rebuilt URL to: telehack.com/
*   Trying 64.13.139.230...
*   Trying 2001:470:67:c0::1337...
* Immediate connect fail for 2001:470:67:c0::1337: Network is unreachable
*   Trying 2001:470:67:c0::1337...
* Immediate connect fail for 2001:470:67:c0::1337: Network is unreachable
*   Trying 2001:470:67:c0::1337...
* Immediate connect fail for 2001:470:67:c0::1337: Network is unreachable
*   Trying 2001:470:67:c0::1337...
* Immediate connect fail for 2001:470:67:c0::1337: Network is unreachable
^C
```

Figure. 13

2.3. Task 3: Bypassing Firewall using VPN

Step 1: Run VPN Server

Run VPN server program `vpnserver.c` in one terminal window on server as shown below.

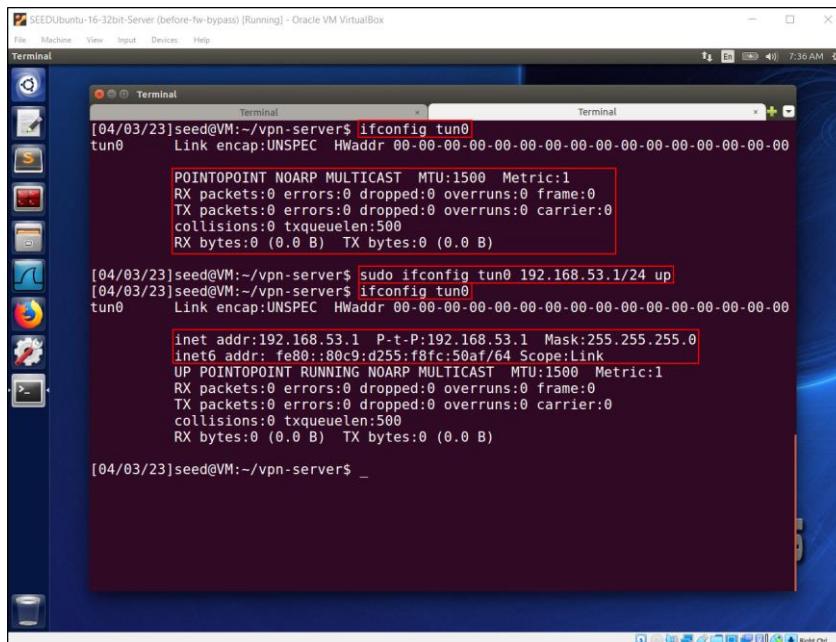


The screenshot shows a terminal window titled "Terminal" with the command history:

```
[04/03/23]seed@VM:~/vpn-server$ ls
Makefile README vpncclient.c vpnserver.c
[04/03/23]seed@VM:~/vpn-server$ make
gcc -o vpnsrvr vpnsrvr.c
gcc -o vpnclient vpnclient.c
[04/03/23]seed@VM:~/vpn-server$ ls
Makefile README vpnclient vpnclient.c vpnsrvr.c
[04/03/23]seed@VM:~/vpn-server$ sudo ./vpnsrvr
```

Figure. 14

While running above `vpnserver.c` program assign IP address 192.168.53.1 to tun0 interface in new terminal window on server as shown below.



The screenshot shows a terminal window titled "Terminal" with the command history:

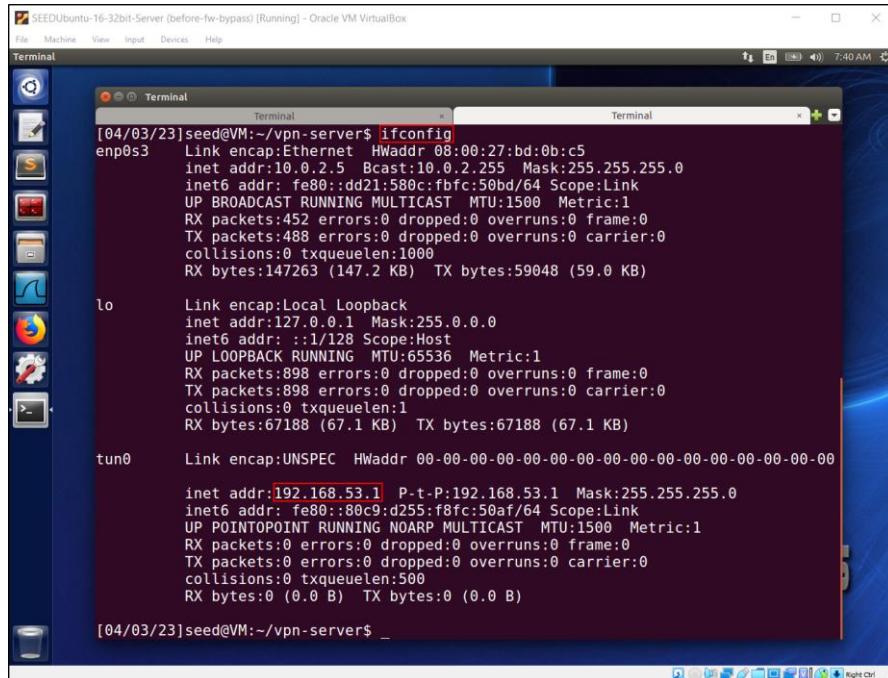
```
[04/03/23]seed@VM:~/vpn-server$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
          POINTOPOINT NOARP MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

[04/03/23]seed@VM:~/vpn-server$ sudo ifconfig tun0 192.168.53.1/24 up
[04/03/23]seed@VM:~/vpn-server$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
          inet addr:192.168.53.1 P-t-P:192.168.53.1 Mask:255.255.255.0
          inet6 addr: fe80::80c9:d255:8fc:50af/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

[04/03/23]seed@VM:~/vpn-server$ _
```

Figure. 15

Ifconfig command output on server after IP address assigned to tun0 interface as shown below.



```
[04/03/23]seed@VM:~/vpn-server$ ifconfig
enp0s3  Link encap:Ethernet HWaddr 08:00:27:bd:0b:c5
        inet addr:10.0.2.5 Bcast:10.0.2.255 Mask:255.255.255.0
        inet6 addr: fe80::dd21:580c:fbfc:50bd/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:452 errors:0 dropped:0 overruns:0 frame:0
              TX packets:488 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:147263 (147.2 KB) TX bytes:59048 (59.0 KB)

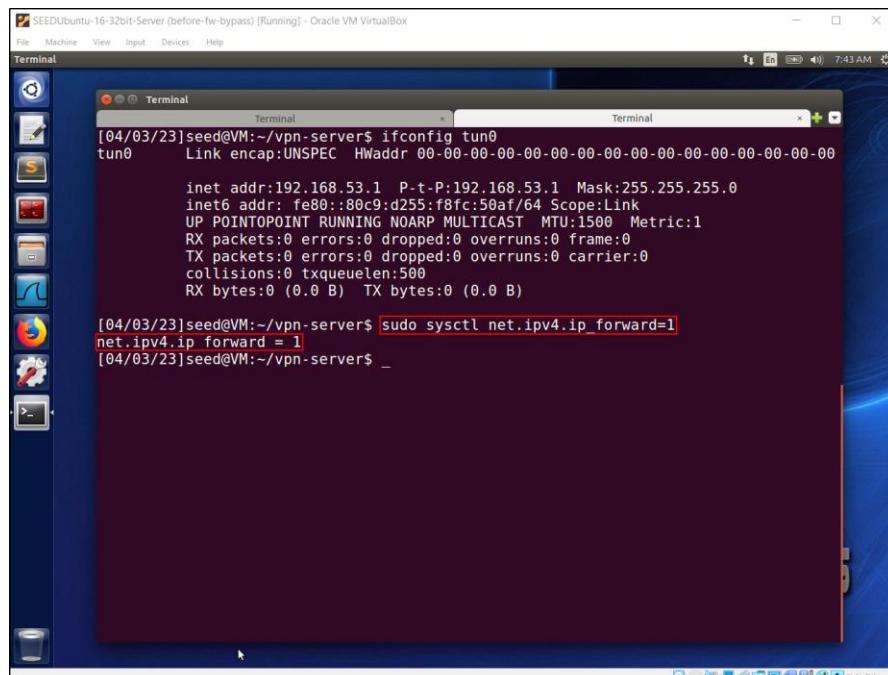
lo    Link encap:Local Loopback
        inet addr:127.0.0.1 Mask:255.0.0.0
        inet6 addr: ::1/128 Scope:Host
              UP LOOPBACK RUNNING MTU:65536 Metric:1
              RX packets:898 errors:0 dropped:0 overruns:0 frame:0
              TX packets:898 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1
              RX bytes:67188 (67.1 KB) TX bytes:67188 (67.1 KB)

tun0   Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
        inet addr:192.168.53.1 P-t-P:192.168.53.1 Mask:255.255.255.0
        inet6 addr: fe80::80c9:d255:f8fc:50af/64 Scope:Link
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:500
              RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

[04/03/23]seed@VM:~/vpn-server$ _
```

Figure. 16

IP forwarding enabled using the following command on server as shown below.



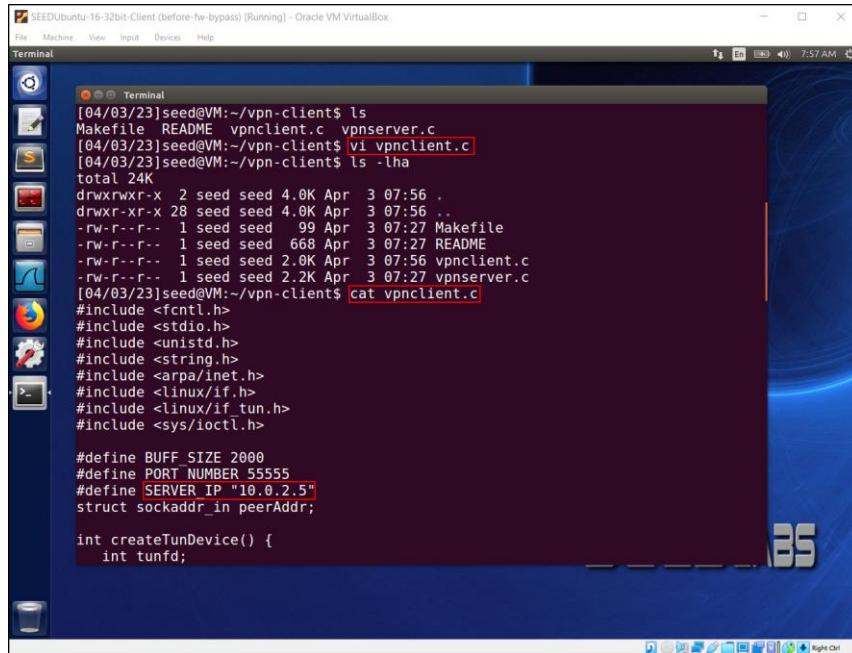
```
[04/03/23]seed@VM:~/vpn-server$ ifconfig tun0
tun0   Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
        inet addr:192.168.53.1 P-t-P:192.168.53.1 Mask:255.255.255.0
        inet6 addr: fe80::80c9:d255:f8fc:50af/64 Scope:Link
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
              RX packets:0 errors:0 dropped:0 overruns:0 frame:0
              TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:500
              RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

[04/03/23]seed@VM:~/vpn-server$ sudo sysctl net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
[04/03/23]seed@VM:~/vpn-server$ _
```

Figure. 17

Step 2: Run VPN Client

Change SERVER_IP address to server's IP address on vpncclient.c file as shown below.



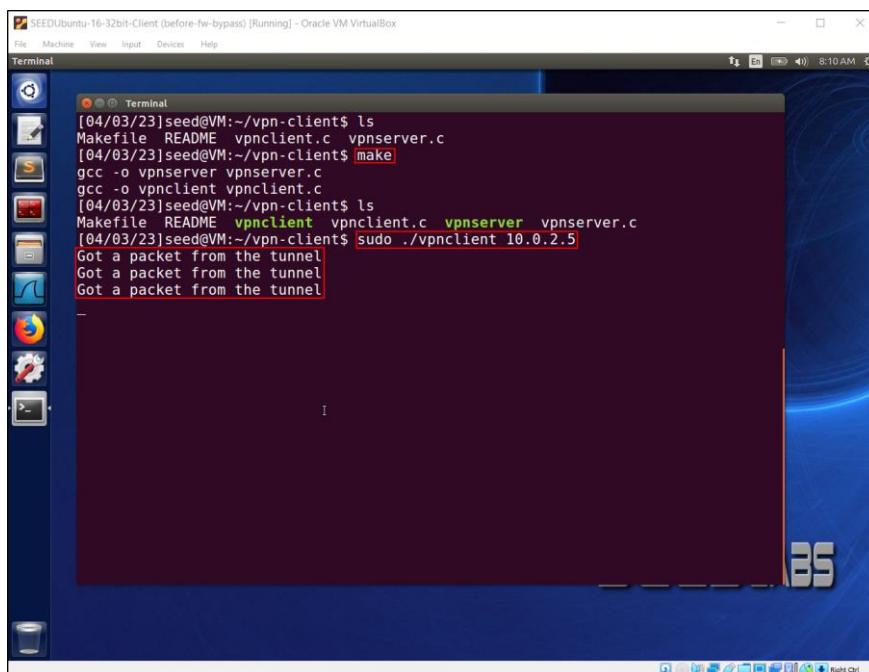
```
[04/03/23]seed@VM:~/vpn-client$ ls
Makefile README vpncclient.c vpnsrvr.c
[04/03/23]seed@VM:~/vpn-client$ vi vpncclient.c
[04/03/23]seed@VM:~/vpn-client$ ls -lha
total 24K
drwxrwxr-x  2 seed seed 4.0K Apr  3 07:56 .
drwxr-xr-x 28 seed seed 4.0K Apr  3 07:56 ..
-rw-r--r--  1 seed seed  99 Apr  3 07:27 Makefile
-rw-r--r--  1 seed seed 668 Apr  3 07:27 README
-rw-r--r--  1 seed seed 2.0K Apr  3 07:56 vpncclient.c
-rw-r--r--  1 seed seed 2.2K Apr  3 07:27 vpnsrvr.c
[04/03/23]seed@VM:~/vpn-client$ cat vpncclient.c
#include <fcntl.h>
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <arpa/inet.h>
#include <linux/if.h>
#include <linux/if_tun.h>
#include <sys/ioctl.h>

#define BUFF_SIZE 2000
#define PORT NUMBER 55555
#define SERVER IP "10.0.2.5"
struct sockaddr_in peerAddr;

int createTunDevice() {
    int tunfd;
```

Figure. 18

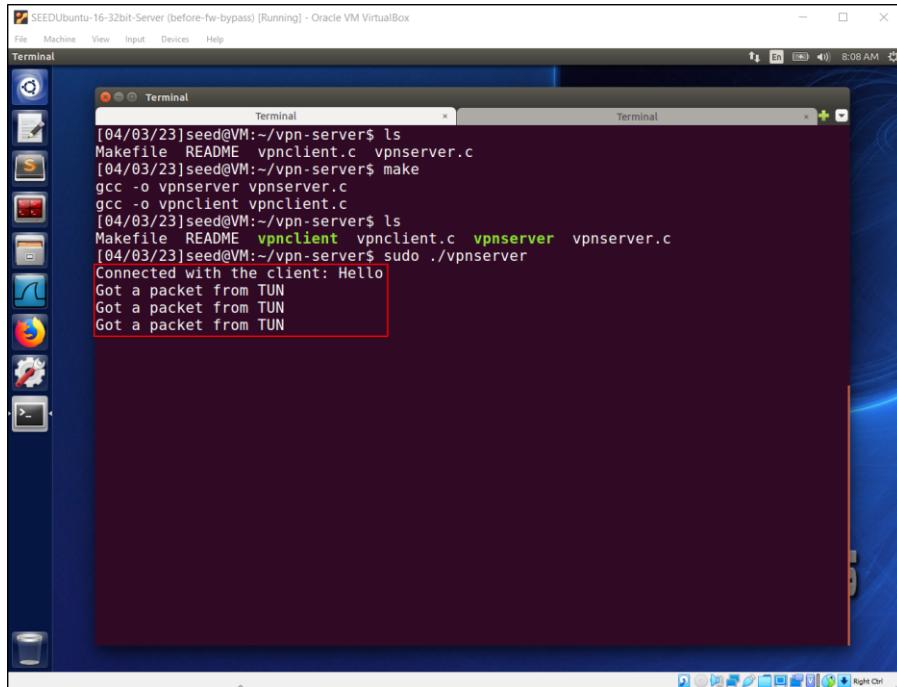
Run VPN client program vpncclient.c in one terminal window on client as shown below.



```
[04/03/23]seed@VM:~/vpn-client$ ls
Makefile README vpncclient.c vpnsrvr.c
[04/03/23]seed@VM:~/vpn-client$ make
gcc -o vpnsrvr vpnsrvr.c
gcc -o vpnclient vpncclient.c
[04/03/23]seed@VM:~/vpn-client$ ls
Makefile README vpncclient.c vpnsrvr.c
[04/03/23]seed@VM:~/vpn-client$ sudo ./vpncclient 10.0.2.5
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel
```

Figure. 19

Client connection message (Hello) on server as shown below.



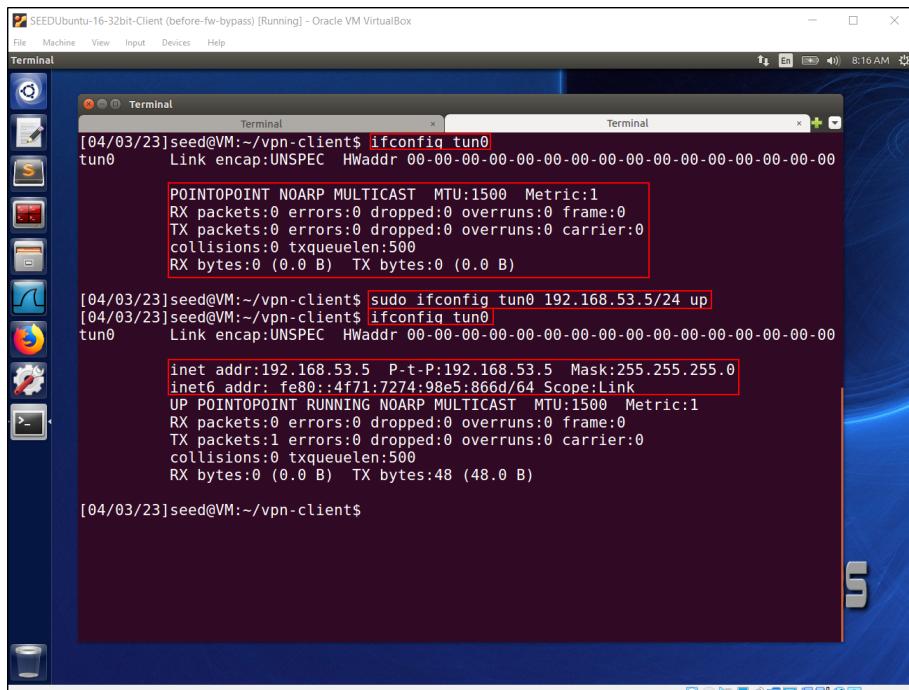
The screenshot shows a terminal window titled "Terminal" running on a Linux desktop environment. The window title bar says "SEEDUbuntu-16-32bit-Server (before-fw-bypass) [Running] - Oracle VM VirtualBox". The terminal content shows the following command-line session:

```
[04/03/23]seed@VM:~/vpn-server$ ls
Makefile README vpncclient.c vpnserver.c
[04/03/23]seed@VM:~/vpn-server$ make
gcc -o vpnserver vpnserver.c
gcc -o vpnclient vpncclient.c
[04/03/23]seed@VM:~/vpn-server$ ls
Makefile README vpncclient.c vpnserver.c
[04/03/23]seed@VM:~/vpn-server$ sudo ./vpnserver
Connected with the client: Hello
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
```

A red box highlights the line "Connected with the client: Hello".

Figure. 20

While running above vpncclient.c program assign IP address 192.168.53.5 to tun0 interface in new terminal window on client as shown below.



The screenshot shows a terminal window titled "Terminal" running on a Linux desktop environment. The window title bar says "SEEDUbuntu-16-32bit-Client (before-fw-bypass) [Running] - Oracle VM VirtualBox". The terminal content shows the following command-line session:

```
[04/03/23]seed@VM:~/vpn-client$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00:00:00:00:00:00
          MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

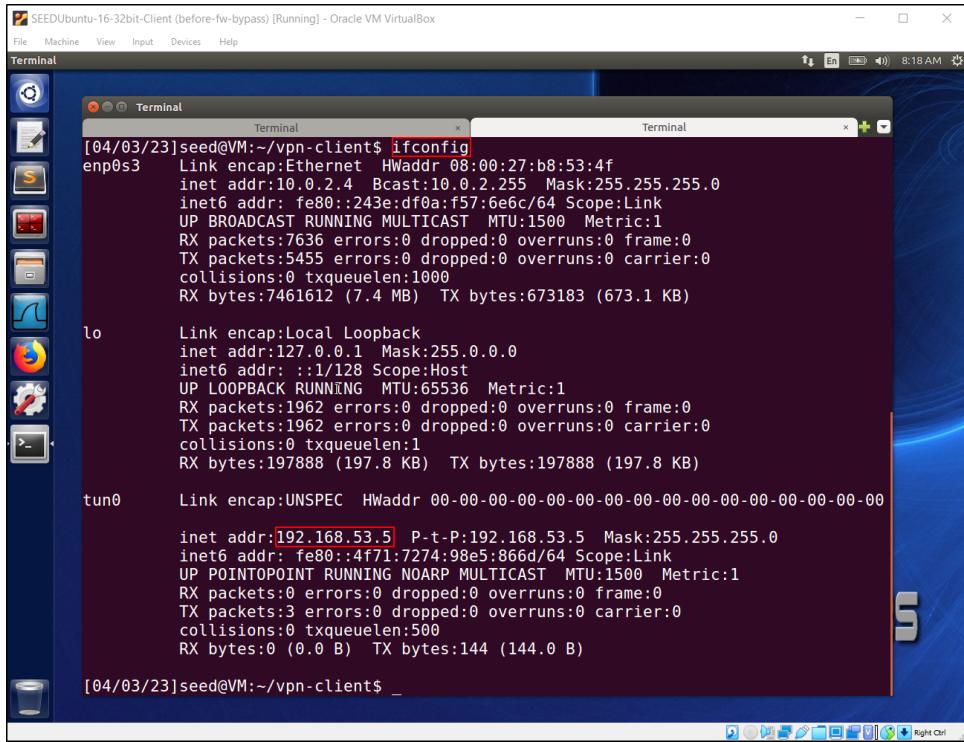
[04/03/23]seed@VM:~/vpn-client$ sudo ifconfig tun0 192.168.53.5/24 up
[04/03/23]seed@VM:~/vpn-client$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00:00:00:00:00:00
          inet addr:192.168.53.5 P-t-P:192.168.53.5 Mask:255.255.255.0
          inet6 addr: fe80::4f71:7274:98e5:866d/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:500
          RX bytes:0 (0.0 B) TX bytes:48 (48.0 B)

[04/03/23]seed@VM:~/vpn-client$
```

Red boxes highlight the line "Connected with the client: Hello" and the line "inet addr:192.168.53.5 P-t-P:192.168.53.5 Mask:255.255.255.0".

Figure. 21

Ifconfig command output on client after IP address assigned to tun0 interface as shown below.



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "Terminal". The command "ifconfig" is run, and its output is displayed. The output shows three network interfaces: enp0s3, lo, and tun0. The tun0 interface has been assigned an IP address of 192.168.53.5. The terminal window is part of a desktop environment with a blue background, and there are other windows and icons visible in the background.

```
[04/03/23]seed@VM:~/vpn-client$ ifconfig
enp0s3      Link encap:Ethernet HWaddr 08:00:27:b8:53:4f
            inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0
              inet6 addr: fe80::2a8:ff%enp0s3/128 Scope:Link
                  UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
                  RX packets:7636 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:5455 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:7461612 (7.4 MB)  TX bytes:673183 (673.1 KB)

lo          Link encap:Local Loopback
            inet addr:127.0.0.1  Mask:255.0.0.0
              inet6 addr: ::1/128 Scope:Host
                  UP LOOPBACK RUNNING MTU:65536 Metric:1
                  RX packets:1962 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:1962 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1
                  RX bytes:197888 (197.8 KB)  TX bytes:197888 (197.8 KB)

tun0        Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
            inet addr:192.168.53.5  P-t-P:192.168.53.5  Mask:255.255.255.0
              inet6 addr: fe80::4f71:7274:98e5:866d/64 Scope:Link
                  UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
                  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:500
                  RX bytes:0 (0.0 B)  TX bytes:144 (144.0 B)

[04/03/23]seed@VM:~/vpn-client$ _
```

Figure. 22

Step 3: Set Up Routing on Client and Server VMs

VPN tunnel tun0 established between client and server after the above two steps. Set up routing paths on both client and server machines to direct the intended traffic through the tunnel as shown in the below figures Figure. 23 and Figure. 24 respectively.

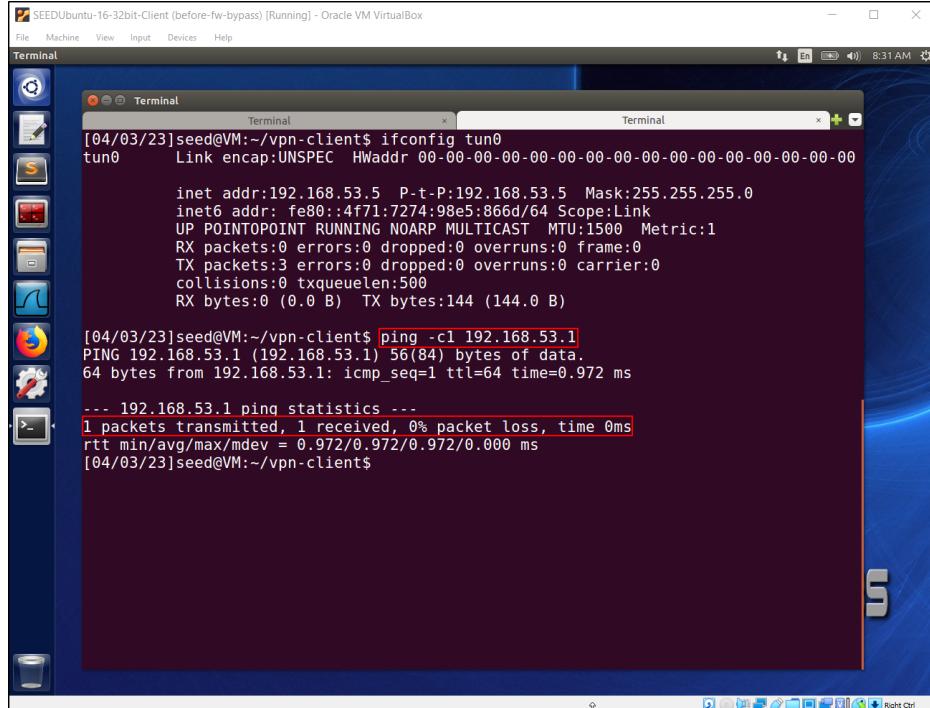
```
[04/03/23]seed@VM:~/vpn-client$ route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         10.0.2.1       0.0.0.0        UG    100    0      0 enp0s3
10.0.2.0        *              255.255.255.0   U     100    0      0 enp0s3
link-local      *              255.255.0.0     U     1000   0      0 enp0s3
192.168.53.0   *              255.255.255.0   U     0      0      0 tun0
[04/03/23]seed@VM:~/vpn-client$ sudo route add -net 192.168.53.0 netmask 255.255.255.255 5.0 dev tun0
[04/03/23]seed@VM:~/vpn-client$ route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         10.0.2.1       0.0.0.0        UG    100    0      0 enp0s3
10.0.2.0        *              255.255.255.0   U     100    0      0 enp0s3
link-local      *              255.255.0.0     U     1000   0      0 enp0s3
192.168.53.0   *              255.255.255.0   U     0      0      0 tun0
192.168.53.0   *              255.255.255.0   U     0      0      0 tun0
[04/03/23]seed@VM:~/vpn-client$ _
```

Figure. 23

```
[04/03/23]seed@VM:~/vpn-server$ route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         10.0.2.1       0.0.0.0        UG    100    0      0 enp0s3
10.0.2.0        *              255.255.255.0   U     100    0      0 enp0s3
link-local      *              255.255.0.0     U     1000   0      0 enp0s3
192.168.53.0   *              255.255.255.0   U     0      0      0 tun0
[04/03/23]seed@VM:~/vpn-server$ sudo route add -net 192.168.53.0 netmask 255.255.255.255 5.0 dev tun0
[04/03/23]seed@VM:~/vpn-server$ route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref  Use Iface
default         10.0.2.1       0.0.0.0        UG    100    0      0 enp0s3
10.0.2.0        *              255.255.255.0   U     100    0      0 enp0s3
link-local      *              255.255.0.0     U     1000   0      0 enp0s3
192.168.53.0   *              255.255.255.0   U     0      0      0 tun0
192.168.53.0   *              255.255.255.0   U     0      0      0 tun0
[04/03/23]seed@VM:~/vpn-server$ _
```

Figure. 24

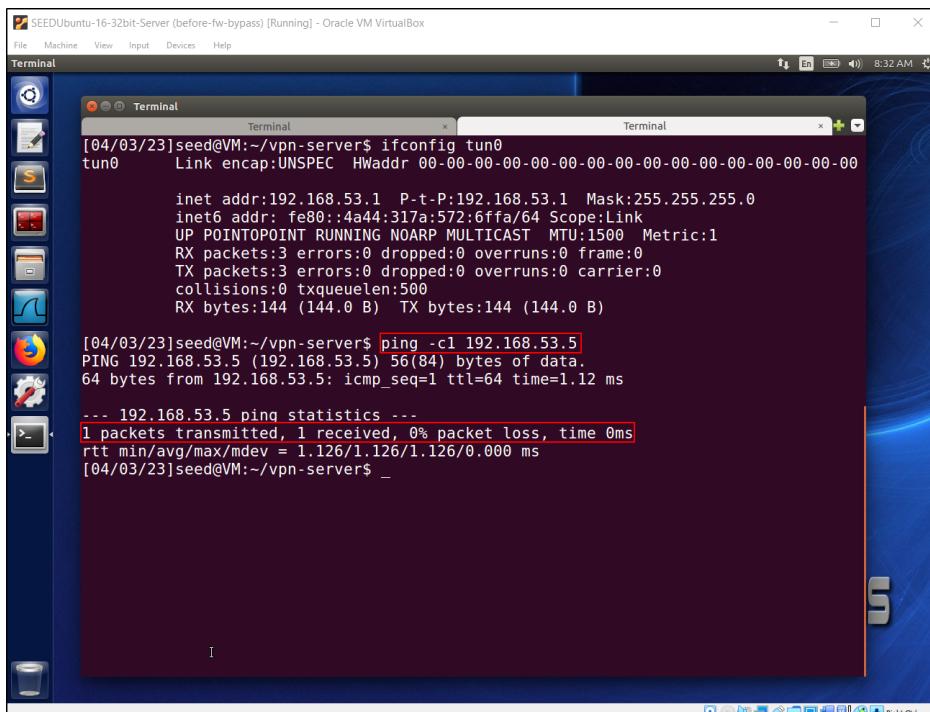
Client and server communicate (ping) each other using tun0 IP address after tun0 interface created as shown in the below figures Figure. 25 and Figure. 26 respectively.



```
[04/03/23]seed@VM:~/vpn-client$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:192.168.53.5 P-t-P:192.168.53.5 Mask:255.255.255.0
          inet6 addr: fe80::4f71:7274:98e5:866d/64 Scope:Link
            UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
            RX packets:0 errors:0 dropped:0 overruns:0 frame:0
            TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:500
            RX bytes:0 (0.0 B) TX bytes:144 (144.0 B)

[04/03/23]seed@VM:~/vpn-client$ ping -c1 192.168.53.1
PING 192.168.53.1 (192.168.53.1) 56(84) bytes of data.
64 bytes from 192.168.53.1: icmp_seq=1 ttl=64 time=0.972 ms
--- 192.168.53.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.972/0.972/0.972/0.000 ms
[04/03/23]seed@VM:~/vpn-client$
```

Figure. 25



```
[04/03/23]seed@VM:~/vpn-server$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:192.168.53.1 P-t-P:192.168.53.1 Mask:255.255.255.0
          inet6 addr: fe80::4a44:317a:572:6ffa/64 Scope:Link
            UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
            RX packets:3 errors:0 dropped:0 overruns:0 frame:0
            TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:500
            RX bytes:144 (144.0 B) TX bytes:144 (144.0 B)

[04/03/23]seed@VM:~/vpn-server$ ping -c1 192.168.53.5
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=1.12 ms
--- 192.168.53.5 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.126/1.126/1.126/0.000 ms
[04/03/23]seed@VM:~/vpn-server$ _
```

Figure. 26

Ssh connection between client and server using tun0 interface IP address as shown in the below figures Figure. 27 and Figure. 28 respectively.

```
[04/03/23]seed@VM:~/vpn-server$ ssh 192.168.53.5
seed@192.168.53.5's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

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

1 package can be updated.
0 updates are security updates.

Last login: Mon Apr  3 08:35:52 2023 from 192.168.53.1
[04/03/23]seed@VM:~$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:192.168.53.5 P-t-P:192.168.53.5 Mask:255.255.255.0
          inet6 addr: fe80::4f71:7274:98e5:866d/64 Scope:Link
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
              RX packets:238 errors:0 dropped:0 overruns:0 frame:0
              TX packets:163 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:500
              RX bytes:23952 (23.9 KB) TX bytes:24064 (24.0 KB)

[04/03/23]seed@VM:~$ _
```

Figure. 27

```
[04/03/23]seed@VM:~/vpn-client$ ssh 192.168.53.1
seed@192.168.53.1's password:
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.8.0-36-generic i686)

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

1 package can be updated.
0 updates are security updates.

Last login: Mon Apr  3 08:38:35 2023 from 192.168.53.5
[04/03/23]seed@VM:~$ ifconfig tun0
tun0      Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:192.168.53.1 P-t-P:192.168.53.1 Mask:255.255.255.0
          inet6 addr: fe80::4a44:317a:572:6ffa/64 Scope:Link
              UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 Metric:1
              RX packets:274 errors:0 dropped:0 overruns:0 frame:0
              TX packets:331 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:500
              RX bytes:36738 (36.7 KB) TX bytes:37006 (37.0 KB)

[04/03/23]seed@VM:~$ _
```

Figure. 28

Step 4: Set Up NAT on Server VM

Enable the NAT Network on server as shown below.

```
[04/03/23]seed@VM:~/vpn-server$ sudo iptables -F
[04/03/23]seed@VM:~/vpn-server$ sudo iptables -t nat -F
[04/03/23]seed@VM:~/vpn-server$ sudo iptables -t nat -A POSTROUTING -j MASQUERADE -o enp0s3
[04/03/23]seed@VM:~/vpn-server$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
[04/03/23]seed@VM:~/vpn-server$ _
```

Figure. 29

Client can connect with target telehack.com via ping tool with Wireshark packet capture as shown below.

```
[04/03/23]seed@VM:~/vpn-client$ ping -c1 telehack.com
PING telehack.com (64.13.139.230) 56(84) bytes of data.
64 bytes from telehack.com (64.13.139.230): icmp_seq=1 ttl=44 time=333 ms

--- telehack.com ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 333.592/333.592/333.592/0.000 ms
[04/03/23]seed@VM:~/vpn-client$ _
```

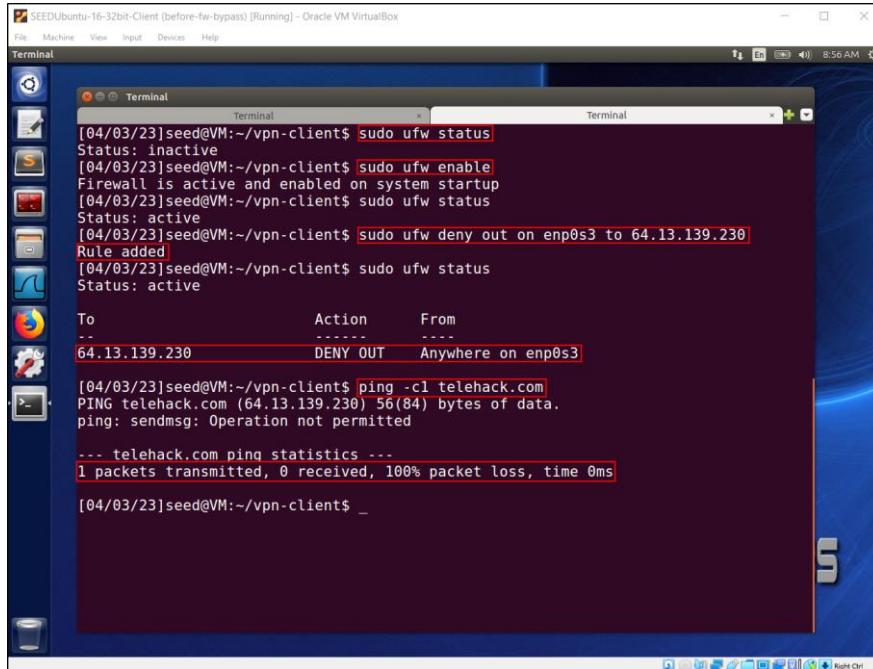
No.	Time	Source	Src Port	Destination	Dst Port	Protocol	Length	Info
1	2023-04-03 08:58:14	10.0.2.4	60089	192.168.1.1	53	DNS	72	Standard query 0xb099 A telehack...
2	2023-04-03 08:58:14	192.168.1.1	53	10.0.2.4	60089	DNS	88	Standard query response 0xb099 A...
3	2023-04-03 08:58:14	10.0.2.4		64.13.139.230		ICMP	98	Echo (ping) request id=0x26f8, s...
4	2023-04-03 08:58:15	64.13.139.230		10.0.2.4		ICMP	98	Echo (ping) reply id=0x26f8, s...
5	2023-04-03 08:58:15	10.0.2.4	27956	192.168.1.1	53	DNS	86	Standard query 0x8a0e PTR 230.139...
6	2023-04-03 08:58:15	192.168.1.1	53	10.0.2.4	27956	DNS	112	Standard query response 0x8a0e PT...

Frame 1: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface 0
 ▶ Ethernet II, Src: PcsCompu_0b:53:4f (00:00:27:b0:53:4f), Dst: RealtekU_12:35:00 (52:54:00:12:35:00)
 ▶ Internet Protocol Version 4, Src: 10.0.2.4, Dst: 192.168.1.1
 ▶ User Datagram Protocol, Src Port: 60089, Dst Port: 53

Figure. 30

Step 5: Bypass firewall with VPN tunnel tun0

Ufw tool used to set firewall rules as deny from client to access target domain telehack.com and try to connect via ping to target but 100% packet loss as shown below.



```
[04/03/23]seed@VM:~/vpn-client$ sudo ufw status
Status: inactive
[04/03/23]seed@VM:~/vpn-client$ sudo ufw enable
Firewall is active and enabled on system startup
[04/03/23]seed@VM:~/vpn-client$ sudo ufw status
Status: active
[04/03/23]seed@VM:~/vpn-client$ sudo ufw deny out on enp0s3 to 64.13.139.230
Rule added
[04/03/23]seed@VM:~/vpn-client$ sudo ufw status
Status: active

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

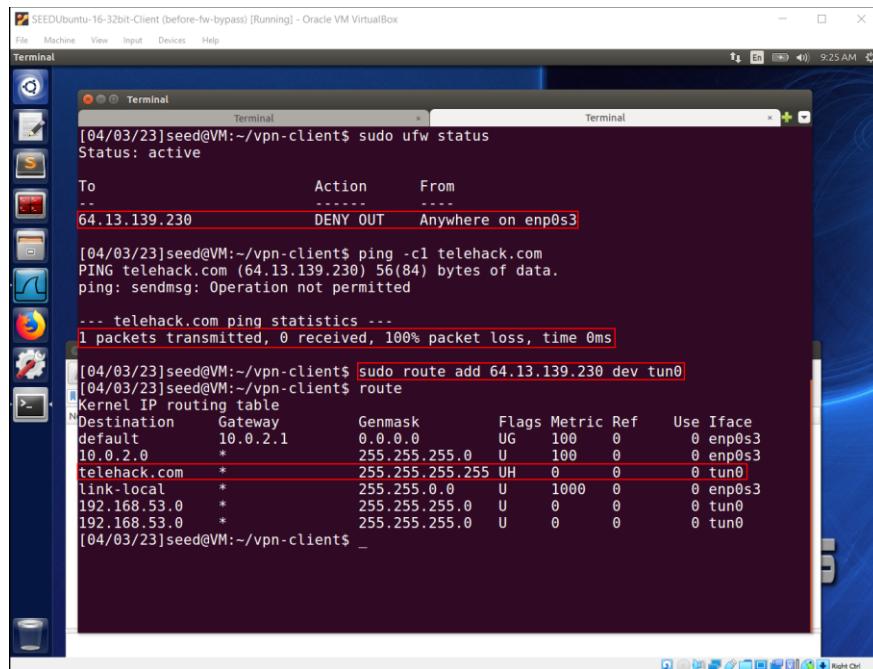
[04/03/23]seed@VM:~/vpn-client$ ping -c1 telehack.com
PING telehack.com (64.13.139.230) 56(84) bytes of data.
ping: sendmsg: Operation not permitted

--- telehack.com ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

[04/03/23]seed@VM:~/vpn-client$ _
```

Figure. 31

Routing entry added accordingly to bypass firewall on the client as shown below.



```
[04/03/23]seed@VM:~/vpn-client$ sudo ufw status
Status: active

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

[04/03/23]seed@VM:~/vpn-client$ ping -c1 telehack.com
PING telehack.com (64.13.139.230) 56(84) bytes of data.
ping: sendmsg: Operation not permitted

--- telehack.com ping statistics ---
1 packets transmitted, 0 received, 100% packet loss, time 0ms

[04/03/23]seed@VM:~/vpn-client$ sudo route add 64.13.139.230 dev tun0
[04/03/23]seed@VM:~/vpn-client$ route
Kernel IP routing table
Destination     Gateway         Genmask        Flags Metric Ref    Use Iface
default         10.0.2.1       0.0.0.0       UG    100    0        0 enp0s3
10.0.2.0        *              255.255.255.0  U     100    0        0 enp0s3
telehack.com    *              255.255.255.255 UH    0      0        0 tun0
link-local      *              255.255.0.0    U     1000   0        0 enp0s3
192.168.53.0   *              255.255.255.0    U     0      0        0 tun0
192.168.53.0   *              255.255.255.0    U     0      0        0 tun0

[04/03/23]seed@VM:~/vpn-client$ _
```

Figure. 32

Able to bypass the firewall and ping to telehack.com with the help of VPN tunnel tun0 with Wireshark packet capture as shown below.

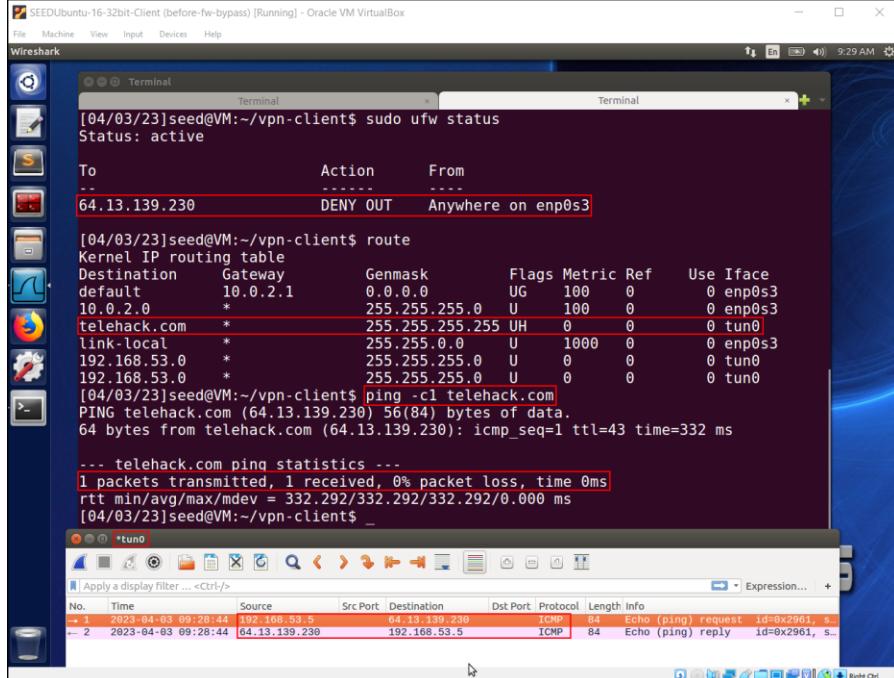


Figure. 33

Now, client can connect to target telehack.com via VPN tunnel tun0 interface IP address. As, evidence curl (port 80) and telnet (port 23) tools with Wireshark packet capture shown in the below figures Figure. 34 and Figure. 35.

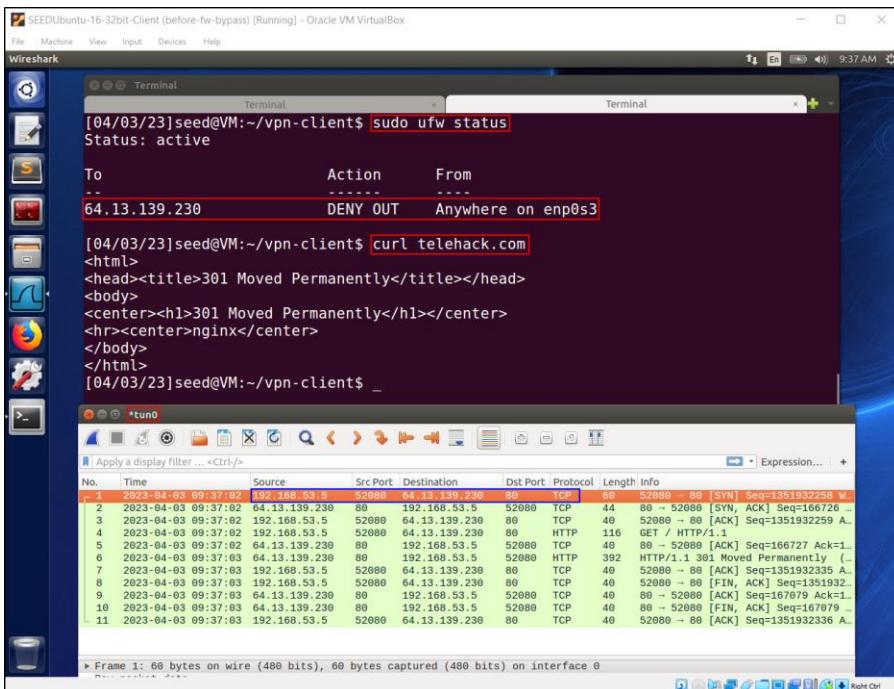


Figure. 34

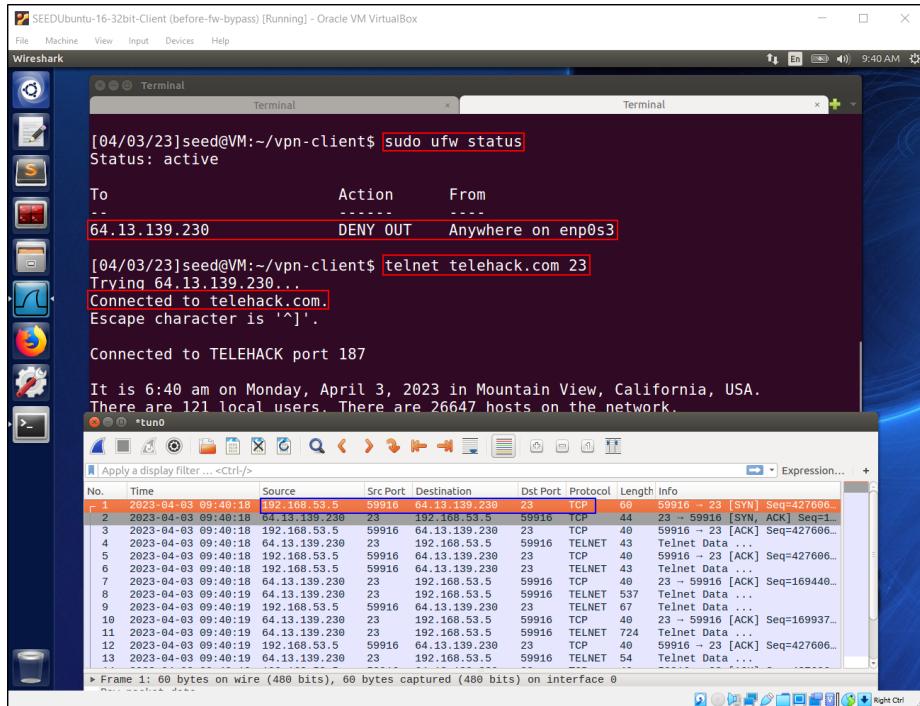


Figure. 35

3. References

1. "Firewall Evasion Techniques and Countermeasures" by SANS Institute: <https://www.sans.org/reading-room/whitepapers/firewalls/firewall-evasion-techniques-countermeasures-33098>
2. "VPN Evasion Techniques: A Deep Dive" by Infosec Institute: <https://resources.infosecinstitute.com/vpn-evasion-techniques-a-deep-dive/>
3. "Bypassing firewalls using VPNs" by Help Net Security: <https://www.helpnetsecurity.com/2018/09/13/bypassing-firewalls-vpns/>
4. "The Dangers of Using a VPN" by Norton: <https://us.norton.com/internetsecurity-privacy-the-dangers-of-using-a-vpn.html>