

Kiel Russell

Lab 7

Task 1:

```
Host U [Running]
tor
/bin/bash
/bin/bash 83x30
inet addr:10.0.2.17 Bcast:10.0.2.255 Mask:255.255.255.0
inet6 addr: fe80::c35b:2b21:482:3022/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:63 errors:0 dropped:0 overruns:0 frame:0
TX packets:61 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:7955 (7.9 KB) TX bytes:6744 (6.7 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:65 errors:0 dropped:0 overruns:0 frame:0
TX packets:65 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:21312 (21.3 KB) TX bytes:21312 (21.3 KB)

[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 10.0.2.1 0.0.0.0 UG 100 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 100 0 0 enp0s3
169.254.0.0 0.0.0.0 255.255.0.0 U 1000 0 0 enp0s3
[03/17/20]seed@VM:~$
```

```
Host U [Running]
/bin/bash
panda.bmp vpn.zip
pic original.bmp words.txt
[03/17/20]seed@VM:~$ sudo ifconfig tun0 192.168.53.1/24 up
SIOCSIFADDR: No such device
tun0: ERROR while getting interface flags: No such device
tun0: ERROR while getting interface flags: No such device
SIOCSIFNETMASK: No such device
[03/17/20]seed@VM:~$ ping 10.0.2.20
PING 10.0.2.20 (10.0.2.20) 56(84) bytes of data.
64 bytes from 10.0.2.20: icmp_seq=1 ttl=64 time=0.363 ms
64 bytes from 10.0.2.20: icmp_seq=2 ttl=64 time=0.768 ms
64 bytes from 10.0.2.20: icmp_seq=3 ttl=64 time=0.781 ms
64 bytes from 10.0.2.20: icmp_seq=4 ttl=64 time=0.527 ms
^C
--- 10.0.2.20 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3055ms
rtt min/avg/max/mdev = 0.363/0.609/0.781/0.177 ms
[03/17/20]seed@VM:~$ ping 192.168.60.6
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.
--- 192.168.60.6 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9206ms

[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 10.0.2.1 0.0.0.0 UG 100 0 0 enp0s3
10.0.2.0 0.0.0.0 255.255.255.0 U 100 0 0 enp0s3
169.254.0.0 0.0.0.0 255.255.0.0 U 1000 0 0 enp0s3
[03/17/20]seed@VM:~$
```

Observation: Used for following configuration for Host U VM with a VPN Client that has a Nat Network adapter with the following IP 10.0.2.17 . HostU connection to the VPN server is accomplished through the same LAB. HostU can use ping to test the server but now reach the HostV machine. The purpose of this lab is to establish a VPN tunnel through a VPN server to allow HostU to talk to HostV through that tunnel.

```
tor
VPN Server [Running]
[03/17/20]seed@VM:~$ ifconfig
enp0s3  Link encap:Ethernet  HWaddr 08:00:27:eb:d3:0a
        inet addr:10.0.2.20  Bcast:10.0.2.255  Mask:255.255.255.0
        inet6 addr: fe80::8332:6af1:4d0f:776f/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:31 errors:0 dropped:0 overruns:0 frame:0
        TX packets:73 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:3526 (3.5 KB)  TX bytes:7574 (7.5 KB)

enp0s8  Link encap:Ethernet  HWaddr 08:00:27:1a:a1:ae
        inet addr:192.168.56.101  Bcast:192.168.56.255  Mask:255.255.255.0
        inet6 addr: fe80::8f13:6869:d68d:38c8/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:62 errors:0 dropped:0 overruns:0 frame:0
        TX packets:55 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:8225 (8.2 KB)  TX bytes:6760 (6.7 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:99 errors:0 dropped:0 overruns:0 frame:0
        TX packets:99 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1
        RX bytes:24713 (24.7 KB)  TX bytes:24713 (24.7 KB)

[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0        10.0.2.1       0.0.0.0         UG    100    0      0 enp0s3
10.0.2.0       0.0.0.0        255.255.255.0   U     100    0      0 enp0s3
169.254.0.0    0.0.0.0        255.255.0.0     U     1000   0      0 enp0s8
192.168.56.0   0.0.0.0        255.255.255.0   U     100    0      0 enp0s8
[03/17/20]seed@VM:~$
```

The image displays two screenshots of a terminal window, likely from a virtual machine, showing network connectivity tests. The terminal is titled 'ator' and shows a bash shell prompt. The background of the terminal window has a blue and white pattern.

The first screenshot shows the following output:

```
/bin/bash
64 bytes from 192.168.60.101: icmp_seq=4 ttl=64 time=142 ms
64 bytes from 192.168.60.101: icmp_seq=5 ttl=64 time=247 ms
64 bytes from 192.168.60.101: icmp_seq=6 ttl=64 time=0.251 ms
^C
--- 192.168.60.101 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5009ms
rtt min/avg/max/mdev = 0.251/124.137/247.853/97.279 ms
[03/17/20]seed@VM:~$ 10.0.2.17
10.0.2.17: command not found
[03/17/20]seed@VM:~$ ping 10.0.2.17
PING 10.0.2.17 (10.0.2.17) 56(84) bytes of data.
64 bytes from 10.0.2.17: icmp_seq=1 ttl=64 time=0.386 ms
64 bytes from 10.0.2.17: icmp_seq=2 ttl=64 time=0.524 ms
64 bytes from 10.0.2.17: icmp_seq=3 ttl=64 time=0.487 ms
^C
--- 10.0.2.17 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2035ms
rtt min/avg/max/mdev = 0.386/0.465/0.524/0.063 ms
[03/17/20]seed@VM:~$
```

The second screenshot shows the same output as the first, indicating that the network connectivity tests were repeated and yielded similar results.

Observation: The above screenshot shows how the VPN server is configured with a Nat Network adapter with IP address 10.0.2.20 and VPN Server – Host Only adapter with IP address 192.168.56.101. Through the use of dual adapters there is now a tunnel for HostU and HostV to communicate.

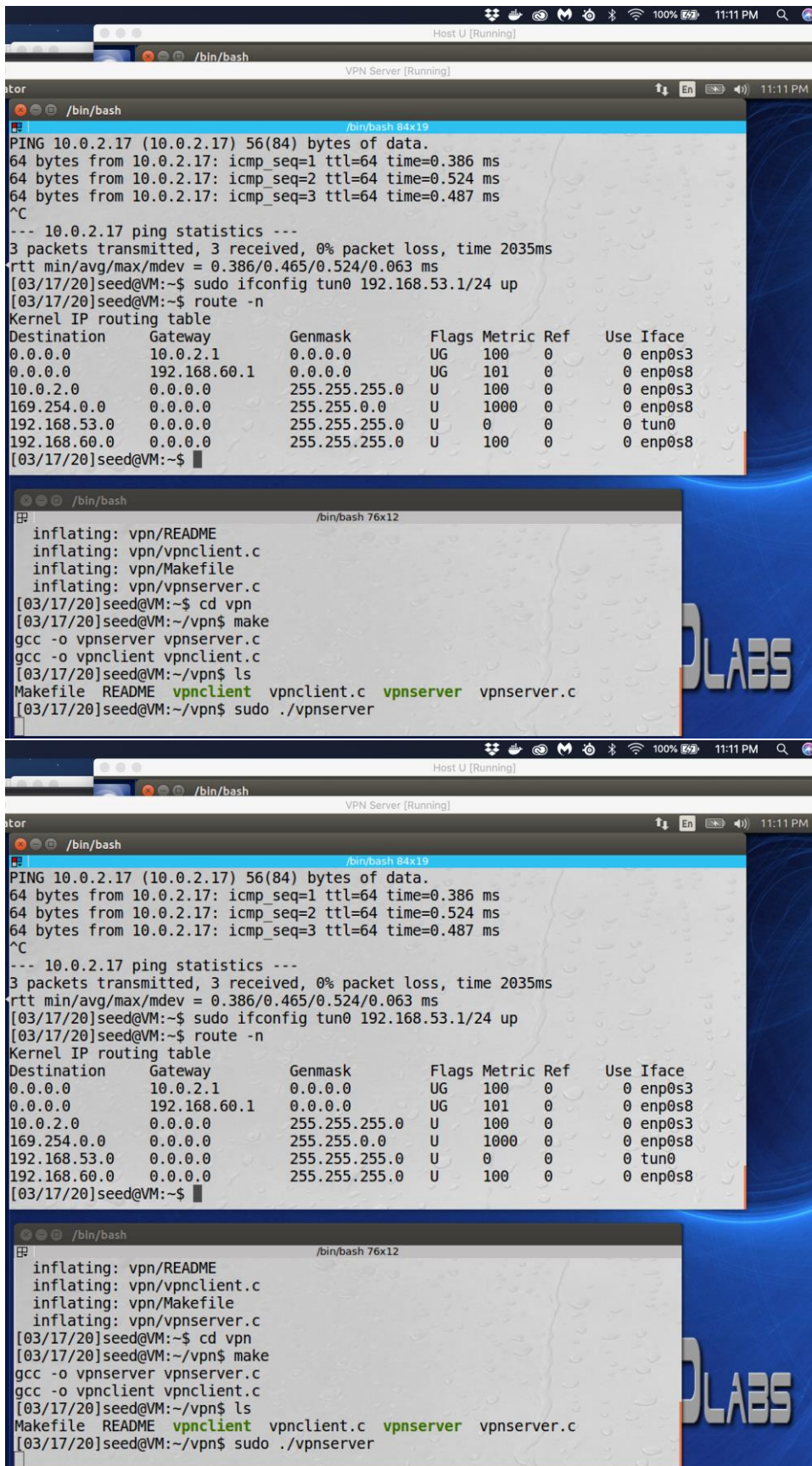

```
Host V [Running]
/bin/bash
/bin/bash 81x30
inet addr:192.168.60.6 Bcast:192.168.60.255 Mask:255.255.255.0
inet6 addr: fe80::d34d:ad2:977d:df1a/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:29 errors:0 dropped:0 overruns:0 frame:0
TX packets:70 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:3564 (3.5 KB) TX bytes:7006 (7.0 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:53 errors:0 dropped:0 overruns:0 frame:0
TX packets:53 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:12437 (12.4 KB) TX bytes:12437 (12.4 KB)

[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
0.0.0.0 192.168.60.6 0.0.0.0 UG 100 0 0 enp0s3
169.254.0.0 0.0.0.0 255.255.0.0 U 1000 0 0 enp0s3
192.168.60.0 0.0.0.0 255.255.255.0 U 100 0 0 enp0s3
[03/17/20]seed@VM:~$
```

```
Host U [Running]
Host V [Running]
/bin/bash
/bin/bash 81x33
64 bytes from 192.168.60.1: icmp_seq=41 ttl=64 time=92.6 ms
64 bytes from 192.168.60.1: icmp_seq=42 ttl=64 time=0.325 ms
64 bytes from 192.168.60.1: icmp_seq=43 ttl=64 time=0.338 ms
64 bytes from 192.168.60.1: icmp_seq=44 ttl=64 time=0.436 ms
64 bytes from 192.168.60.1: icmp_seq=45 ttl=64 time=0.439 ms
64 bytes from 192.168.60.1: icmp_seq=46 ttl=64 time=0.363 ms
64 bytes from 192.168.60.1: icmp_seq=47 ttl=64 time=0.297 ms
64 bytes from 192.168.60.1: icmp_seq=48 ttl=64 time=65.1 ms
64 bytes from 192.168.60.1: icmp_seq=49 ttl=64 time=0.304 ms
64 bytes from 192.168.60.1: icmp_seq=50 ttl=64 time=0.360 ms
64 bytes from 192.168.60.1: icmp_seq=51 ttl=64 time=0.264 ms
64 bytes from 192.168.60.1: icmp_seq=52 ttl=64 time=0.288 ms
64 bytes from 192.168.60.1: icmp_seq=53 ttl=64 time=40.7 ms
64 bytes from 192.168.60.1: icmp_seq=54 ttl=64 time=0.267 ms
64 bytes from 192.168.60.1: icmp_seq=55 ttl=64 time=75.8 ms
64 bytes from 192.168.60.1: icmp_seq=56 ttl=64 time=0.306 ms
64 bytes from 192.168.60.1: icmp_seq=57 ttl=64 time=0.347 ms
64 bytes from 192.168.60.1: icmp_seq=58 ttl=64 time=0.285 ms
^C
--- 192.168.60.1 ping statistics ---
58 packets transmitted, 58 received, 0% packet loss, time 58145ms
rtt min/avg/max/mdev = 0.264/51.844/273.926/80.303 ms
[03/17/20]seed@VM:~$ ping 10.0.2.20
PING 10.0.2.20 (10.0.2.20) 56(84) bytes of data.
64 bytes from 10.0.2.20: icmp_seq=1 ttl=64 time=0.369 ms
64 bytes from 10.0.2.20: icmp_seq=2 ttl=64 time=0.568 ms
64 bytes from 10.0.2.20: icmp_seq=3 ttl=64 time=0.685 ms
64 bytes from 10.0.2.20: icmp_seq=4 ttl=64 time=0.698 ms
^C
--- 10.0.2.20 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3055ms
rtt min/avg/max/mdev = 0.369/0.580/0.698/0.131 ms
[03/17/20]seed@VM:~$
```

Observation: HostV is a computer inside the private networks that is configured to have a Host Only adapter with ip address 192.168.60.6



Observation: The VPN Server is first spun up through the command `sudo ./vpnservice`. Then I need to configure the tun() interface. The command is `sudo ifconfig tun0 (ip address)/24 up`. This command accomplishes assigning the IP address to the tun(), spun up the VPN server, and configured the route through the route table.


```
Host U [Running]
/bin/bash
VPN Server [Running]
11:12 PM

Search your computer
/bin/bash 84x19
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:1149 errors:0 dropped:0 overruns:0 frame:0
TX packets:1149 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:109904 (109.9 KB) TX bytes:109904 (109.9 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::9f02:a812:43b0:6d5c/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)

[03/17/20]seed@VM:~$

/bin/bash
/bin/bash 76x12
inflating: vpn/README
inflating: vpn/vpnclient.c
inflating: vpn/Makefile
inflating: vpn/vpnserver.c
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ make
gcc -o vpnserver vpnserver.c
gcc -o vpnclient vpnclient.c
[03/17/20]seed@VM:~/vpn$ ls
Makefile README vpnclient.c vpnserver.c
[03/17/20]seed@VM:~/vpn$ sudo ./vpnserver

Host U [Running]
/bin/bash
VPN Server [Running]
11:12 PM

Search your computer
/bin/bash 84x19
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:1149 errors:0 dropped:0 overruns:0 frame:0
TX packets:1149 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:109904 (109.9 KB) TX bytes:109904 (109.9 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::9f02:a812:43b0:6d5c/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)

[03/17/20]seed@VM:~$

/bin/bash
/bin/bash 76x12
inflating: vpn/README
inflating: vpn/vpnclient.c
inflating: vpn/Makefile
inflating: vpn/vpnserver.c
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ make
gcc -o vpnserver vpnserver.c
gcc -o vpnclient vpnclient.c
[03/17/20]seed@VM:~/vpn$ ls
Makefile README vpnclient.c vpnserver.c
[03/17/20]seed@VM:~/vpn$ sudo ./vpnserver
```


Observation: The above screenshot displays the IP address configured to the tun() interface (ipaddress)

```
Host U [Running]
tor
/bin/bash
/bin/bash 83x30
#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 "10.0.2.20"
struct sockaddr_in peerAddr;

int createTunDevice() {
    int tunfd;
    struct ifreq ifr;
    memset(&ifr, 0, sizeof(ifr));

    ifr.ifr_flags = IFF_TUN | IFF_NO_PI;

    tunfd = open("/dev/net/tun", 0_RDWR);
    ioctl(tunfd, TUNSETIFF, &ifr);

    return tunfd;
}

int connectToUDPServer(){
    int sockfd;
-- INSERT --
12,29 Top
```

Observation: The ipaddress is configured in the above screenshot to the IP address that is standing in for the VPN server.

```
Host U [Running]
tor
/bin/bash
/bin/bash 83x19
[03/17/20]seed@VM:~$ ping 192.168.60.6
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.

--- 192.168.60.6 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9206ms

[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination      Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0          10.0.2.1       0.0.0.0         UG    100    0      0 enp0s3
10.0.2.0         0.0.0.0        255.255.255.0   U     100    0      0 enp0s3
169.254.0.0      0.0.0.0        255.255.0.0     U     1000   0      0 enp0s3
[03/17/20]seed@VM:~$ vim vpnclient
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ vim vpnclient
[03/17/20]seed@VM:~/vpn$ ls
Makefile  README  vpnclient  vpnclient.c  vpnserver  vpnserver.c
[03/17/20]seed@VM:~/vpn$ vim vpnclient.c
[03/17/20]seed@VM:~/vpn$ sudo ifconfig tun0 192.168.53.5/24 up
/bin/bash
/bin/bash 80x24
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ sudo ./vpnclient
```

Observation: Using the created VPN client we run the vpnclient program through the configured tun() interface using the ipaddress "" for the tun interface to connect through the vpn client. The following command "sudo ifconfig tun0 ""/24 up will route traffic automatically through the vpn tunnel.

```

tor
VPN Server [Running]
/bin/bash
/bin/bash 87x17
0.0.0.0      192.168.60.1    0.0.0.0     UG    101    0      0 enp0s8
10.0.2.0     0.0.0.0         255.255.255.0 U     100    0      0 enp0s3
169.254.0.0  0.0.0.0         255.255.0.0  U    1000    0      0 enp0s8
192.168.53.0 0.0.0.0         255.255.255.0 U     0      0      0 tun0
192.168.60.0 0.0.0.0         255.255.255.0 U    100    0      0 enp0s8
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ make
gcc -o vpnserver vpnserver.c
gcc -o vpnclient vpnclient.c
[03/17/20]seed@VM:~/vpn$ sudo ./vpnserver
^C
[03/17/20]seed@VM:~/vpn$ sudo ./vpnserver
Connected with the client: Hello
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel

```

Observation: The server program initializes a TUN interface(tun0) and is in waiting mode for a tunnel connection request from the vpn client. The client program creates the TUN interface(tun0) and forwards a "hello" message to establish the VPN tunnel as shown in the screenshot above. Through this tun0 interface packets will be enable communication through the VPN tunnel.

Step 3: Configuring Routing on the VPN Client and Vpn Server

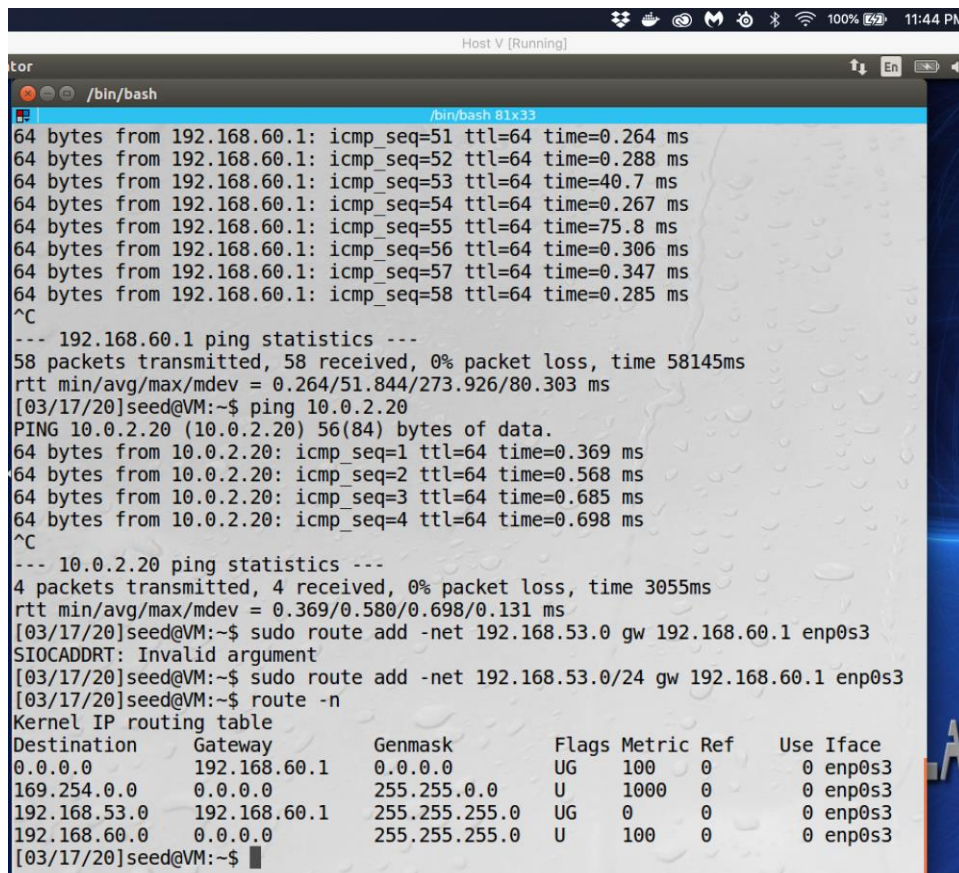
```
VPN Server [Running]
minator
Host U [Running]
/bin/bash
/bin/bash 80x18
gcc10.0.2.0      0.0.0.0      255.255.255.0  U    100    0      0 enp0s3
[03/169.254.0.0  0.0.0.0      255.255.0.0    U    1000   0      0 enp0s3
Ker[03/17/20]seed@VM:~$ sudo ifconfig tun0 192.168.53.5/24 up
DesSIOCSIFADDR: No such device
0.0tun0: ERROR while getting interface flags: No such device
10.0tun0: ERROR while getting interface flags: No such device
169SIOCSIFNETMASK: No such device
[03/03/17/20]seed@VM:~$ sudo ifconfig tun0 192.168.53.5/24 up
[03/03/17/20]seed@VM:~$ sudo route add -net 192.168.60.0/24 tun0
Ker[03/17/20]seed@VM:~$ route -n
DesKernel IP routing table
0.0Destination      Gateway      Genmask      Flags Metric Ref    Use Iface
10.00.0.0.0          10.0.2.1     0.0.0.0      UG    100    0      0 enp0s3
16910.0.2.0          0.0.0.0      255.255.255.0 U    100    0      0 enp0s3
192169.254.0.0       0.0.0.0      255.255.0.0   U    1000   0      0 enp0s3
[0303/192.168.53.0    0.0.0.0      255.255.255.0 U    0      0      0 tun0
[0303/192.168.60.0    0.0.0.0      255.255.255.0 U    0      0      0 tun0
SIO[03/17/20]seed@VM:~$
[03/17/20]seed@VM:~$ sudo ./vpnclient
sudo: ./vpnclient: command not found
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ sudo ./vpnclient
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
```

Observation: The route through the tun0 interface is setup with the following command `sudo route add -net "ipaddress"/24 tun0`. This instructs the VPN Client to forward all traffic from "ipaddress" through the VPN tunnel we just created.

```
[03/18/20]seed@VM:~/vpn$ sudo ifconfig tun0 192.168.59.1/24 up
[03/18/20]seed@VM:~/vpn$ 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
default          192.168.60.1 0.0.0.0      UG    101    0      0 enp0s8
10.0.2.0         *            255.255.255.0 U    100    0      0 enp0s3
link-local       *            255.255.0.0   U    1000   0      0 enp0s8
192.168.59.0     *            255.255.255.0 U    0      0      0 tun0
192.168.60.0     *            255.255.255.0 U    100    0      0 enp0s8
[03/18/20]seed@VM:~/vpn$ sudo ifconfig tun0 192.168.59.1/24 up
[03/18/20]seed@VM:~/vpn$ 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
default          192.168.60.1 0.0.0.0      UG    101    0      0 enp0s8
10.0.2.0         *            255.255.255.0 U    100    0      0 enp0s3
link-local       *            255.255.0.0   U    1000   0      0 enp0s8
192.168.59.0     *            255.255.255.0 U    0      0      0 tun0
192.168.60.0     *            255.255.255.0 U    100    0      0 enp0s8
```


Observation: On the client and server VM's, the traffic flow needs to be done through a routing table entry that all traffic from the "ipaddress"/24 network are to go through the tun0 interface. This entry is put into the routing table automatically when we assigned the ipaddress to the tun0 interface.

Step 4: Configure Routing on Host V



```
Host V [Running]
tor
/bin/bash
/bin/bash 81x33
64 bytes from 192.168.60.1: icmp_seq=51 ttl=64 time=0.264 ms
64 bytes from 192.168.60.1: icmp_seq=52 ttl=64 time=0.288 ms
64 bytes from 192.168.60.1: icmp_seq=53 ttl=64 time=40.7 ms
64 bytes from 192.168.60.1: icmp_seq=54 ttl=64 time=0.267 ms
64 bytes from 192.168.60.1: icmp_seq=55 ttl=64 time=75.8 ms
64 bytes from 192.168.60.1: icmp_seq=56 ttl=64 time=0.306 ms
64 bytes from 192.168.60.1: icmp_seq=57 ttl=64 time=0.347 ms
64 bytes from 192.168.60.1: icmp_seq=58 ttl=64 time=0.285 ms
^C
--- 192.168.60.1 ping statistics ---
58 packets transmitted, 58 received, 0% packet loss, time 58145ms
rtt min/avg/max/mdev = 0.264/51.844/273.926/80.303 ms
[03/17/20]seed@VM:~$ ping 10.0.2.20
PING 10.0.2.20 (10.0.2.20) 56(84) bytes of data:
64 bytes from 10.0.2.20: icmp_seq=1 ttl=64 time=0.369 ms
64 bytes from 10.0.2.20: icmp_seq=2 ttl=64 time=0.568 ms
64 bytes from 10.0.2.20: icmp_seq=3 ttl=64 time=0.685 ms
64 bytes from 10.0.2.20: icmp_seq=4 ttl=64 time=0.698 ms
^C
--- 10.0.2.20 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3055ms
rtt min/avg/max/mdev = 0.369/0.580/0.698/0.131 ms
[03/17/20]seed@VM:~$ sudo route add -net 192.168.53.0 gw 192.168.60.1 enp0s3
SIOCADDRT: Invalid argument
[03/17/20]seed@VM:~$ sudo route add -net 192.168.53.0/24 gw 192.168.60.1 enp0s3
[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0        192.168.60.1   0.0.0.0         UG    100    0      0 enp0s3
169.254.0.0    0.0.0.0        255.255.0.0     U     1000   0      0 enp0s3
192.168.53.0    192.168.60.1   255.255.255.0   UG     0      0      0 enp0s3
192.168.60.0    0.0.0.0        255.255.255.0   U     100    0      0 enp0s3
[03/17/20]seed@VM:~$
```

Observation: Any reply packets should now utilize the same VPN tunnel so they are encrypted. This is done by routing all packets for the "ipaddress"/24 network through the VPN tunnel. For HostV, the following command `sudo route add -net "ipaddress"/24 gw "ipaddress of gateway" enp0s3`. HostV will then connect to the "ipaddress"/24 network via the enp0s3 network interface.

Step 5: Test the VPN tunnel

Prior to the configuring of the VPN server there would be no reply from HostV because the tunnel connecting the two VM's is not established yet. Following the VPN being setup now HostU and HostV can communicate to one another through the VPN tunnel.

```
Host U [Running]
tor
/bin/bash
/bin/bash 83x19
[03/17/20]seed@VM:~$ ping 192.168.60.6
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.

--- 192.168.60.6 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9206ms

[03/17/20]seed@VM:~$ route -n
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
0.0.0.0          10.0.2.1        0.0.0.0          UG    100    0      0 enp0s3
10.0.2.0         0.0.0.0         255.255.255.0    U     100    0      0 enp0s3
169.254.0.0      0.0.0.0         255.255.0.0      U     1000   0      0 enp0s3
[03/17/20]seed@VM:~$ vim vpnclient
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ vim vpnclient
[03/17/20]seed@VM:~/vpn$ ls
Makefile  README  vpnclient  vpnclient.c  vpnserver  vpnserver.c
[03/17/20]seed@VM:~/vpn$ vim vpnclient.c
[03/17/20]seed@VM:~/vpn$ sudo ifconfig tun0 192.168.53.5/24 up
/bin/bash
/bin/bash 80x24
[03/17/20]seed@VM:~$ cd vpn
[03/17/20]seed@VM:~/vpn$ sudo ./vpnclient
```

Observation: Pinging HostV from HostU will result in packets being sent over the gateway- host only adapter.

Observation: The screenshot above shows the packets generated when using ping between to test packet receipt on HostV from HostU. The way that our routing table is set up the ICMP packet is routed to the TUN interface, the VPN application then gets the ICMP packet, forwards the packet to the VPN tunnel, stores that inside a UDP packet then sends it towards the VPN server. The reply UDP packet from the VPN server contains the ICMP echo reply packet. The tunnel on the VPN client then receives this UDP packet, extracts the ICMP packet, and sends that to the kernel through the tun0 interface.

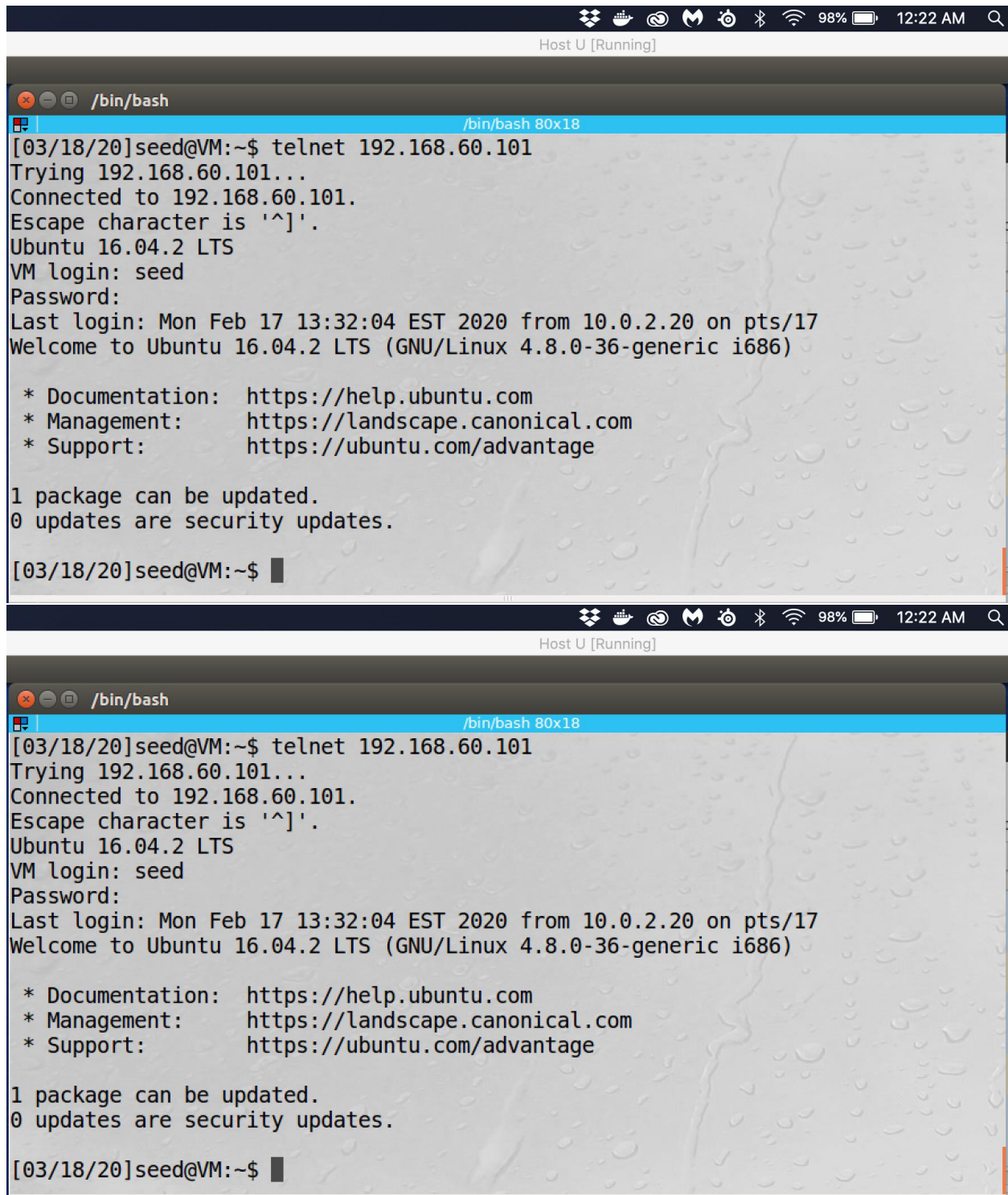
The image contains two identical screenshots of a terminal window. The window title is "/bin/bash" and the terminal title bar is "/bin/bash 80x18". The terminal output shows a user at a VM prompt ([03/18/20]seed@VM:~\$) running the command "telnet 192.168.60.101". The output indicates a successful connection to 192.168.60.101, showing the escape character as '^]', the Ubuntu version as 16.04.2 LTS, and the login details for user 'seed'. It also displays system information, including the last login time and a list of updates (1 package can be updated, 0 security updates). The terminal ends with the prompt [03/18/20]seed@VM:~\$.

Observation: The screenshot above displays the successful connection to the telnet server on Host V within the private network from HostU that is communicating through the VPN tunnel.

Observation: The screenshot above is the wireshark packet capture file from the successful connection to the telnet server that is initiated on HostV that is contained within the private network from HostU using the VPN tunnel.

Step 6: Testing of breaking the VPN Tunnel

Observation: Connection through telnet from HostV to HostU for this demonstration of breaking the tunnel.



```
[03/18/20]seed@VM:~$ telnet 192.168.60.101
Trying 192.168.60.101...
Connected to 192.168.60.101.
Escape character is '^]'.
Ubuntu 16.04.2 LTS
VM login: seed
Password:
Last login: Mon Feb 17 13:32:04 EST 2020 from 10.0.2.20 on pts/17
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.

[03/18/20]seed@VM:~$
```

Observation: The above screenshot shows that the VPN tunnel that we have make is broken. Telnet is of course still live but the tunnel with which packets were forwarded is no longer active. This will cause

the packets to keep attempting to send the packets but the result will be that none of the packets will successfully be transmitted through the VPN tunnel.

```
Host U [Running] Host V [Running] 100% 12:15 AM

/bin/bash
/bin/bash 80x18
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:1547 errors:0 dropped:0 overruns:0 frame:0
TX packets:1547 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:138271 (138.2 KB) TX bytes:138271 (138.2 KB)

tun0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
inet addr:192.168.59.2 P-t-P:192.168.59.2 Mask:255.255.255.0
inet6 addr: fe80::b53e:c888:431d:ba49/64 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)

[03/18/20]seed@VM:~$

/bin/bash 80x18
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
^C
[03/18/20]seed@VM:~/vpn$ sudo ./vpnclient
^C
[03/18/20]seed@VM:~/vpn$ sudo ./vpnclient
^C
[03/18/20]seed@VM:~/vpn$ sudo ./vpnclient
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
```



```
Host U [Running] Host V [Running]
/bin/bash
/bin/bash 80x18
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:65536 Metric:1
RX packets:1547 errors:0 dropped:0 overruns:0 frame:0
TX packets:1547 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:138271 (138.2 KB) TX bytes:138271 (138.2 KB)
tun0 Link encap:UNSPEC HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
-00
inet addr:192.168.59.2 P-t-P:192.168.59.2 Mask:255.255.255.0
inet6 addr: fe80::b53e:c888:431d:ba49/64 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)
[03/18/20]seed@VM:~$
/bin/bash 80x18
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
^C
[03/18/20]seed@VM:~/vpn$ sudo ./vpncclient
^C
[03/18/20]seed@VM:~/vpn$ sudo ./vpncclient
^C
[03/18/20]seed@VM:~/vpn$ sudo ./vpncclient
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from the tunnel
Got a packet from TUN
Got a packet from TUN
Got a packet from TUN
```

Observation: The above screenshot shows that the telnet connection is still active, TCP will keep trying to send packets, but none will be successful because the VPN tunnel is broken. Although all of the information that is typed into the telnet session will be saved in a buffer until the tunnel is reconnecting causing all of the typed information to be sent through from the buffer.

```
Host U [Running]

/bin/bash
4 packets transmitted, 0 received, 100% packet loss, time 3147ms

[03/18/20]seed@VM:~$ sudo route add -net 192.168.60.0/24 tun0
[03/18/20]seed@VM:~$ 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=63 time=531 ms
64 bytes from 192.168.60.101: icmp_seq=2 ttl=63 time=19.1 ms
64 bytes from 192.168.60.101: icmp_seq=3 ttl=63 time=494 ms
64 bytes from 192.168.60.101: icmp_seq=4 ttl=63 time=10.9 ms
64 bytes from 192.168.60.101: icmp_seq=5 ttl=63 time=15.0 ms
64 bytes from 192.168.60.101: icmp_seq=6 ttl=63 time=14.3 ms
64 bytes from 192.168.60.101: icmp_seq=7 ttl=63 time=4.66 ms
^C
--- 192.168.60.101 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6276ms
rtt min/avg/max/mdev = 4.660/155.781/531.702/226.264 ms
[03/18/20]seed@VM:~$

/bin/bash 80x18
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
Got a packet from TUN
Got a packet from the tunnel
```

Observation: This shows what happens when the tunnels are reconnected, the VPN server/client are restarted, and then on a second terminal the “ipaddress” is assigned to their tun0 interfaces.

510	2020-03-18	00:18:54.8940516	192.168.200.1	10.0.2.17	DNS	139 Standard query response 0x4985 No such name
533	2020-03-18	00:19:06.0374330	10.0.2.20	10.0.2.3	DHCP	342 DHCP Request - Transaction ID 0x338d524b
534	2020-03-18	00:19:06.0412081	10.0.2.3	255.255.255.255	DHCP	590 DHCP ACK - Transaction ID 0x338d524b
542	2020-03-18	00:20:16.3731842	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
543	2020-03-18	00:20:16.9042756	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
544	2020-03-18	00:20:17.3943374	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
545	2020-03-18	00:20:17.4132667	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
546	2020-03-18	00:20:18.4447321	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
547	2020-03-18	00:20:18.9377087	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
548	2020-03-18	00:20:19.4459354	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
549	2020-03-18	00:20:19.4566981	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
550	2020-03-18	00:20:20.5793513	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
551	2020-03-18	00:20:20.5936663	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
554	2020-03-18	00:20:21.6135249	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
557	2020-03-18	00:20:21.6272643	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
558	2020-03-18	00:20:22.6498432	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
559	2020-03-18	00:20:22.6538858	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
510	2020-03-18	00:18:54.8940516	192.168.200.1	10.0.2.17	DNS	139 Standard query response 0x4985 No such name
533	2020-03-18	00:19:06.0374330	10.0.2.20	10.0.2.3	DHCP	342 DHCP Request - Transaction ID 0x338d524b
534	2020-03-18	00:19:06.0412081	10.0.2.3	255.255.255.255	DHCP	590 DHCP ACK - Transaction ID 0x338d524b
542	2020-03-18	00:20:16.3731842	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
543	2020-03-18	00:20:16.9042756	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
544	2020-03-18	00:20:17.3943374	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
545	2020-03-18	00:20:17.4132667	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
546	2020-03-18	00:20:18.4447321	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
547	2020-03-18	00:20:18.9377087	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
548	2020-03-18	00:20:19.4459354	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
549	2020-03-18	00:20:19.4566981	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
550	2020-03-18	00:20:20.5793513	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
551	2020-03-18	00:20:20.5936663	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
554	2020-03-18	00:20:21.6135249	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
557	2020-03-18	00:20:21.6272643	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84
558	2020-03-18	00:20:22.6498432	10.0.2.17	10.0.2.20	UDP	126 43872 - 55555 Len=84
559	2020-03-18	00:20:22.6538858	10.0.2.20	10.0.2.17	UDP	126 55555 - 43872 Len=84

Observation: The wireshark capture shows that all of the packets that were in the buffer were finally sent through.

The image contains two screenshots of a terminal window, likely from a virtual machine. The terminal shows a series of commands and their outputs. The first screenshot shows the user running 'ls' and seeing a directory listing. The second screenshot shows the user running 'time' and seeing timing information. The terminal window has a title bar that says 'Host U [Running]' and a status bar at the bottom showing system icons and the time '12:28 A'.

```
p /bin/bash
/bin/bash 80x18
The program 'hello' can be found in the following packages:
* hello
* hello-traditional
Try: sudo apt install <selected package>
[03/18/20]seed@VM:~$ ls
android      Desktop      examples.desktop  Music      source      Videos
bin          Documents    get-pip.py        Pictures    Templates
Customization Downloads    lib              Public      testfile
[03/18/20]seed@VM:~$ s
s: command not found
[03/18/20]seed@VM:~$ time

real    0m0.000s
user    0m0.000s
sys      0m0.000s
[03/18/20]seed@VM:~$ date
Wed Mar 18 00:28:06 EDT 2020
[03/18/20]seed@VM:~$
```

The second screenshot shows the same terminal window with the same commands and outputs, indicating that the commands were buffered and then executed when the VPN tunnel was reconnected.

```
p /bin/bash
/bin/bash 80x18
The program 'hello' can be found in the following packages:
* hello
* hello-traditional
Try: sudo apt install <selected package>
[03/18/20]seed@VM:~$ ls
android      Desktop      examples.desktop  Music      source      Videos
bin          Documents    get-pip.py        Pictures    Templates
Customization Downloads    lib              Public      testfile
[03/18/20]seed@VM:~$ s
s: command not found
[03/18/20]seed@VM:~$ time

real    0m0.000s
user    0m0.000s
sys      0m0.000s
[03/18/20]seed@VM:~$ date
Wed Mar 18 00:28:06 EDT 2020
[03/18/20]seed@VM:~$
```

Observation: The VPN tunnel has been successfully reconnected, all of the commands were placed into a buffer, waiting on them to be sent over the telnet server connection. When telnet reconnects the server will receive the commands in the buffer, echo the commands back to the telnet client, then display them to the terminal window. Because this is a TCP transmission as soon as the VPN tunnel was reestablished the buffered commands show up on the client side terminal