Task 1: VM Setup

The Nat Network is 10.211.55.0/24
The private network is 192.168.60.0/24

Host U (Client)

```
seed@SeedUbuntu:~$ ifconfig
enp0s5    Link encap:Ethernet  HWaddr 00:1c:42:b1:12:6d
          inet addr:10.211.55.4  Bcast:10.211.55.255  Mask:255.255.255.0
          inet6 addr: fe80::92bf:1449:a87:363b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2531 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1099 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3539520 (3.5 MB)  TX bytes:70984 (70.9 KB)
```

Gateway (VPN Server)

```
seed@SeedUbuntu:~$ ifconfig
enp0s5    Link encap:Ethernet  HWaddr 00:1c:42:88:b6:0d
          inet addr:10.211.55.5  Bcast:10.211.55.255  Mask:255.255.255.0
          inet6 addr: fe80::3b72:fb76:ff65:3d0d/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:73 errors:0 dropped:0 overruns:0 frame:0
          TX packets:81 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:8342 (8.3 KB)  TX bytes:8363 (8.3 KB)

enp0s6    Link encap:Ethernet  HWaddr 00:1c:42:5f:fc:18
          inet addr:192.168.60.5  Bcast:192.168.60.255  Mask:255.255.255.0
          inet6 addr: fe80::7f5f:42f6:2ad4:4a3a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:96 errors:0 dropped:0 overruns:0 frame:0
          TX packets:170 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:10748 (10.7 KB)  TX bytes:15011 (15.0 KB)
```

Host V (On private network)

```
seed@SeedUbuntu:~$ ifconfig
enp0s5    Link encap:Ethernet  HWaddr 00:1c:42:d6:99:2b
          inet addr:192.168.60.6  Bcast:192.168.60.255  Mask:255.255.255.0
          inet6 addr: fe80::54ee:a7bf:425e:e785/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:67 errors:0 dropped:0 overruns:0 frame:0
          TX packets:175 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7613 (7.6 KB)  TX bytes:15362 (15.3 KB)
```

Task 2: Creating a VPN Tunnel using TUN/TAP

Step 1: Run VPN Server

```
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ sudo ./vpnserver
[sudo] password for seed:
```

seed@SeedUbuntu: /media/psf/VirtualBox Share/vpn/vpn_unencrypted

```
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ sudo sh ./setup_vpnserver.sh
[sudo] password for seed:
Setting IP forwarding preference: net.ipv4.ip_forward = 1
Setting IP Address for tun interface: 192.168.53.1
```

setup_vpnserver.sh

```
1  #!/bin/bash
2
3  echo -n "Setting IP forwarding preference: "
4  sudo sysctl net.ipv4.ip_forward=1
5
6  echo -n "Setting IP Address for tun interface: "
7  ifconfig tun0 192.168.53.1/24 up
8  ifconfig -a tun0 | sed -n "s/.*inet addr:\([0-9\.]*\).*/\1/p"
```

Step 2: Run VPN Client

seed@SeedUbuntu: /media/psf/VirtualBox Share/vpn/vpn_unencrypted

```
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ sudo ./vpnclien
t 10.211.55.5
[sudo] password for seed:
Got a packet from the tunnel
Got a packet from the tunnel
```

seed@SeedUbuntu: /media/psf/VirtualBox Share/vpn/vpn_unencrypted

```
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ sudo sh ./setup
_vpnclient.sh
[sudo] password for seed:
Setting IP Address for tun interface:192.168.53.5
Setting route for tun interface:
    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     0      0        0 tun0
```

setup_vpnclient.sh

```
1  #!/bin/bash
2
3  echo -n "Setting IP Address for tun interface:"
4  ifconfig tun0 192.168.53.5/24 up
5  ifconfig -a tun0 | sed -n "s/.*inet addr:\([0-9\.]*\).*/\1/p"
6
7  echo "Setting route for tun interface: "
8  route add -net 192.168.60.0/24 tun0
9  route -n | grep -i tun0 | sed -n "s/\(.*\)/    \1/p"
```

Step 3: Routing on client and server

I setup routing using a bash script as seen in step 2.

Step 4: Routing on Host V

```
seed@SeedUbuntu: /media/psf/VirtualBox Share/vpn/vpn_unencrypted
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ sudo sh ./setup_vpnpr
ivate.sh
[sudo] password for seed:
Setting private host route
    192.168.53.0    192.168.60.5    255.255.255.0   UG    0      0        0 enp0s5
```

setup_vpnprivate.sh

```
1  #!/bin/bash
2
3  echo "Setting private host route"
4  route add -net 192.168.53.0/24 gw 192.168.60.5
5  route -n | grep -i 192.168.53.0 | sed -n "s/\(.*\)/    \1/p"
```

Step 5: Test the tunnel

Ping

```
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ ping -c 3 192.168.60.6
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.
64 bytes from 192.168.60.6: icmp_seq=1 ttl=63 time=0.479 ms
64 bytes from 192.168.60.6: icmp_seq=2 ttl=63 time=1.19 ms
64 bytes from 192.168.60.6: icmp_seq=3 ttl=63 time=1.16 ms

--- 192.168.60.6 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2012ms
rtt min/avg/max/mdev = 0.479/0.945/1.194/0.330 ms
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.211.55.4	10.211.55.5	UDP	128	50215 → 55555 Len=84
2	0.000073444	192.168.53.5	192.168.60.6	ICMP	100	Echo (ping) request id=0x2ab5, seq=1/256, ttl=64 (no response found!)
3	0.000082359	192.168.53.5	192.168.60.6	ICMP	100	Echo (ping) request id=0x2ab5, seq=1/256, ttl=63 (reply in 4)
4	0.000279184	192.168.60.6	192.168.53.5	ICMP	100	Echo (ping) reply id=0x2ab5, seq=1/256, ttl=64 (request in 3)
5	0.000285041	192.168.60.6	192.168.53.5	ICMP	100	Echo (ping) reply id=0x2ab5, seq=1/256, ttl=63
6	0.000328244	10.211.55.5	10.211.55.4	UDP	128	55555 → 50215 Len=84

Telnet

```
seed@SeedUbuntu:/media/psf/VirtualBox Share/vpn/vpn_unencrypted$ telnet 192.168.60.6
Trying 192.168.60.6...
Connected to 192.168.60.6.
Escape character is '^]'.
Ubuntu 16.04.6 LTS
SeedUbuntu login: seed
Password:
Last login: Fri May  1 14:10:07 EDT 2020 from 192.168.53.5 on pts/2
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-99-generic x86_64)
```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.211.55.4	10.211.55.5	UDP	104	50215 → 55555 Len=60
2	0.000068260	192.168.53.5	192.168.60.6	TCP	76	40788 → 23 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_P
3	0.000077204	192.168.53.5	192.168.60.6	TCP	76	[TCP Out-Of-Order] 40788 → 23 [SYN] Seq=0 Win=64240 Le
4	0.000243976	192.168.60.6	192.168.53.5	TCP	76	23 → 40788 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=
5	0.000248273	192.168.60.6	192.168.53.5	TCP	76	[TCP Out-Of-Order] 23 → 40788 [SYN, ACK] Seq=0 Ack=1 W
6	0.000283344	10.211.55.5	10.211.55.4	UDP	104	55555 → 50215 Len=60
7	0.000467721	10.211.55.4	10.211.55.5	UDP	96	50215 → 55555 Len=52
8	0.000507790	192.168.53.5	192.168.60.6	TCP	68	40788 → 23 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=368
9	0.000512694	192.168.53.5	192.168.60.6	TCP	68	[TCP Dup ACK 8#1] 40788 → 23 [ACK] Seq=1 Ack=1 Win=642
10	0.000584493	10.211.55.4	10.211.55.5	UDP	123	50215 → 55555 Len=79
11	0.000621916	192.168.53.5	192.168.60.6	TELNET	95	Telnet Data ...
12	0.000626023	192.168.53.5	192.168.60.6	TCP	95	[TCP Retransmission] 40788 → 23 [PSH, ACK] Seq=1 Ack=1
13	0.000675245	192.168.60.6	192.168.53.5	TCP	68	23 → 40788 [ACK] Seq=1 Ack=28 Win=65152 Len=0 TSval=23
14	0.000679976	192.168.60.6	192.168.53.5	TCP	68	[TCP Dup ACK 13#1] 23 → 40788 [ACK] Seq=1 Ack=28 Win=6
15	0.000727645	10.211.55.5	10.211.55.4	UDP	96	55555 → 50215 Len=52
16	0.023344936	192.168.60.6	192.168.53.5	TELNET	80	Telnet Data ...
17	0.023356732	192.168.60.6	192.168.53.5	TCP	80	[TCP Retransmission] 23 → 40788 [PSH, ACK] Seq=1 Ack=2
18	0.023408188	10.211.55.5	10.211.55.4	UDP	108	55555 → 50215 Len=64

Step 6 Tunnel-Breaking Test

When breaking the tunnel and attempting to type something new the terminal does not respond. When I reconnect the tunnel, the connection does not resume.

Task 3: Encrypting the Tunnel

At this point I consolidated the VPN server and VPN client programs with the setup steps, so that the IP configuration and routing happens automatically when the server and client are run.

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$
sudo ./vpnserver 192.168.53.1
[sudo] password for seed:
Enter PEM pass phrase:
Setting IP forwarding preference: net.ipv4.ip_forward = 1
Setting IP Address for tun interface: 192.168.53.1
Waiting for connection...
0: New Connection Established
0: New Connection Authenticated
0: Got a packet from TUN
0: Got a packet from TUN
```

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$
 sudo ./vpnclient vpn.server.my 4433 192.168.53.4
[sudo] password for seed:
Setting IP Address for tun interface: 192.168.53.4
Setting route for tun interface:
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     0      0        0 tun0
Attempting TCP Connection...Success
Attempting SSL connection...Success!
SSL connection using AES256-GCM-SHA384
Username: seed
Password:
Authenticating User...Success!
Got a packet from TUN
Got a packet from TUN
Got a packet from the tunnel
```

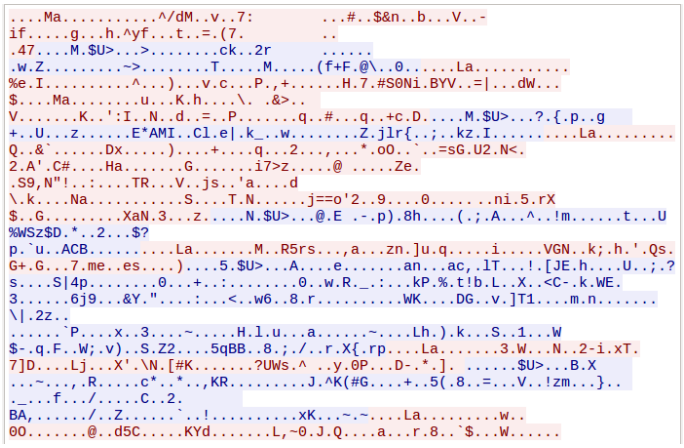
In order to demonstrate the encryption, I ran telnet from host U to connect to host V.

```
seed@SeedUbuntu:~$ telnet 192.168.60.6
Trying 192.168.60.6...
Connected to 192.168.60.6.
Escape character is '^]'.
Ubuntu 16.04.6 LTS
SeedUbuntu login: seed
Password:
Last login: Wed Apr 29 17:37:40 EDT 2020 from localhost on pts/2
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-99-generic x86_64)

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


0 packages can be updated.
0 updates are security updates.

seed@SeedUbuntu:~$ ls
Desktop    Downloads         fontconfig  Pictures  Templates
Documents  examples.desktop  Music       Public    Videos
seed@SeedUbuntu:~$
```

If you follow the TCP stream in Wireshark from the Host U (VPN client) to the VPN server you can see that the data is encrypted and unreadable.

tcp.stream eq 0

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.211.55.4	10.211.55.5	TLSv1.2	150	Application Data
2	0.000022816	10.211.55.5	10.211.55.4	TCP	68	4433 → 45870 [ACK]
7	0.000556631	10.211.55.5	10.211.55.4	TLSv1.2	150	Application Data
8	0.000667164	10.211.55.4	10.211.55.5	TCP	68	45870 → 4433 [ACK]
9	0.000753043	10.211.55.4	10.211.55.5	TLSv1.2	149	Application Data
12	0.044985531	10.211.55.5	10.211.55.4	TCP	68	4433 → 45870 [ACK]
13	0.568941017	10.211.55.4	10.211.55.5	TLSv1.2	150	Application Data
14	0.568963495	10.211.55.5	10.211.55.4	TCP	68	4433 → 45870 [ACK]
19	0.569501815	10.211.55.5	10.211.55.4	TLSv1.2	150	Application Data
20	0.569702514	10.211.55.4	10.211.55.5	TLSv1.2	149	Application Data
23	0.612481431	10.211.55.5	10.211.55.4	TCP	68	4433 → 45870 [ACK]
24	4.751786419	10.211.55.4	10.211.55.5	TLSv1.2	145	Application Data
25	4.751863234	10.211.55.5	10.211.55.4	TCP	68	4433 → 45870 [ACK]
29	11.134948815	10.211.55.5	10.211.55.4	TLSv1.2	151	Application Data
30	11.134966992	10.211.55.5	10.211.55.4	TCP	68	4433 → 45870 [ACK]
35	11.135522001	10.211.55.5	10.211.55.4	TLSv1.2	151	Application Data
36	11.135696011	10.211.55.4	10.211.55.5	TLSv1.2	149	Application Data



If you do the same and follow the TCP stream from the VPN server to host V (the private host) then you can see that the data has been decrypted and is now readable.

tcp.stream eq 1

No.	Time	Source	Destination	Protocol	Length	Info
3	0.000104471	192.168.53.5	192.168.60.6	TELNET	69	Telnet Data ...
4	0.000113085	192.168.53.5	192.168.60.6	TCP	69	[TCP Keep-Alive] 3425
5	0.000505233	192.168.60.6	192.168.53.5	TELNET	69	Telnet Data ...
6	0.000510908	192.168.60.6	192.168.53.5	TCP	69	[TCP Keep-Alive] 23 →
10	0.000806447	192.168.53.5	192.168.60.6	TCP	68	34256 → 23 [ACK] Seq=2
11	0.000813498	192.168.53.5	192.168.60.6	TCP	68	[TCP Keep-Alive ACK] 3
15	0.569070509	192.168.53.5	192.168.60.6	TELNET	69	Telnet Data ...
16	0.569079155	192.168.53.5	192.168.60.6	TCP	69	[TCP Keep-Alive] 3425
17	0.569440086	192.168.60.6	192.168.53.5	TELNET	69	Telnet Data ...
18	0.569448471	192.168.60.6	192.168.53.5	TCP	69	[TCP Keep-Alive] 23 →
21	0.569772925	192.168.53.5	192.168.60.6	TCP	68	34256 → 23 [ACK] Seq=3
22	0.569779308	192.168.53.5	192.168.60.6	TCP	68	[TCP Keep-Alive ACK] 3
31	11.135068565	192.168.53.5	192.168.60.6	TELNET	70	Telnet Data ...
32	11.135076590	192.168.53.5	192.168.60.6	TCP	70	[TCP Retransmission] 3
33	11.135431954	192.168.60.6	192.168.53.5	TELNET	70	Telnet Data ...
34	11.135440131	192.168.60.6	192.168.53.5	TCP	70	[TCP Retransmission] 2
37	11.135753708	192.168.53.5	192.168.60.6	TCP	68	34256 → 23 [ACK] Seq=5

Wireshark · Follow TCP Stream (tcp.stream eq 1) · any

```
llss
.
.[0m.[01;34mDesktop.[0m    .[01;34mDownloads.[0m         .[01;34mfontconfig.
[0m  .[01;34mPictures.[0m  .[01;34mTemplates.[0m
.[01;34mDocuments.[0m  examples.desktop  .[01;34mMusic.[0m       .
[01;34mPublic.[0m    .[01;34mVideos.[0m
.]0;seed@SeedUbuntu: ~..[01;32mseed@SeedUbuntu.[00m:.[01;34m~.[00m$
```

Task 4: Authenticating the VPN Server

In the code below, the server certificate verification is carried out by the first two lines of code. *SSL_CTX_set_verify* tells the application to verify the server's certificate when connecting. *SSL_CTX_load_verify_locations* informs the program where it can find the CA certificates to perform the verification. Finally, the last two lines verify the hostname against the certificate's subject parameter.

```
SSL_CTX_set_verify(ctx, SSL_VERIFY_PEER, NULL);
if (SSL_CTX_load_verify_locations(ctx, NULL, CA_DIR) < 1)
{
   printf("Error setting the verify locations. \n");
   exit(0);
}
ssl = SSL_new(ctx);

X509_VERIFY_PARAM *vpm = SSL_get0_param(ssl);
X509_VERIFY_PARAM_set1_host(vpm, hostname, 0);
```

Intended server demo:

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$
sudo ./vpnclient vpn.server.my 4433 192.168.53.5
[sudo] password for seed:
Setting IP Address for tun interface: 192.168.53.5
Setting routes for tun interface:
192.168.53.0    0.0.0.0         255.255.255.0   U     0      0        0 tun0
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     0      0        0 tun0
Attempting TCP Connection...Success
Attempting SSL connection...Success!
SSL connection using AES256-GCM-SHA384
Username: seed
Password:
Authenticating User...Success!
Got a packet from TUN
Got a packet from the tunnel
```

Not intended server demo:

In order to simulate a VPN attempting to impersonate vpn.server.my, I changed the name in /etc/hosts to vpn.server.yours and pointed it to the same IP address. This way I can have another host name use my server's certificate. When running the client pointed to vpn.server.yours the hostname verification fails.

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$
sudo ./vpnclient vpn.server.yours 4433 192.168.53.5
Setting IP Address for tun interface: 192.168.53.5
Setting routes for tun interface:
192.168.53.0    0.0.0.0         255.255.255.0   U     0      0        0 tun0
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     0      0        0 tun0
Attempting TCP Connection...Success
Attempting SSL connection...140701997155992:error:14090086:SSL routines:ssl3_
get_server_certificate:certificate verify failed:s3_clnt.c:1264:
```

Task 5: Authenticating the VPN Client

When attempting to connect to the VPN server, the client is asked for a user name and password. The data is transmitted through the ssl tunnel and compared to the /etc/shadow file on the server machine. If the user is a valid user on the server, they are permitted to connect. Otherwise the connection is terminated.

Successful Authentication:

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$
sudo ./vpnclient vpn.server.my 4433 192.168.53.5
Setting IP Address for tun interface: 192.168.53.5
Setting routes for tun interface:
192.168.53.0    0.0.0.0         255.255.255.0   U     0      0        0 tun0
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     0      0        0 tun0
192.168.60.0    0.0.0.0         255.255.255.0   U     1      0        0 tun0
Attempting TCP Connection...Success
Attempting SSL connection...Success!
SSL connection using AES256-GCM-SHA384
Username: seed
Password:
Authenticating User...Success!
Got a packet from TUN
Got a packet from TUN
Got a packet from the tunnel
Got a packet from the tunnel
```

Failed Authentication:

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$
sudo ./vpnclient vpn.server.my 4433 192.168.53.5
Setting IP Address for tun interface: 192.168.53.5
Setting routes for tun interface:
192.168.53.0    0.0.0.0         255.255.255.0   U     0      0        0 tun0
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     0      0        0 tun0
192.168.60.0    0.0.0.0         255.255.255.0   U     2      0        0 tun0
Attempting TCP Connection...Success
Attempting SSL connection...Success!
SSL connection using AES256-GCM-SHA384
Username: notseed
Password:
Authenticating User...Failure!
```

Task 6: Supporting Multiple Clients

In order to support multiple clients I utilized the fort() method in a similar way to how a standard TCP server works. The numbers at the beginning of each line represent which client is communicating.

```
seed@SeedUbuntu:~/Desktop/Parallels Shared Folders/VirtualBox Share/vpn/vpn$ sudo ./vpnserver 192.168.53.1
[sudo] password for seed:
Enter PEM pass phrase:
Setting IP forwarding preference: net.ipv4.ip_forward = 1
Setting IP Address for tun interface: 192.168.53.1
Setting route for tun interface:
192.168.53.0    0.0.0.0         255.255.255.0   U     0      0        0 tun0
192.168.53.0    0.0.0.0         255.255.255.0   U     0      0        0 tun0
Waiting for connection...
0: New Connection Established
0: New Connection Authenticated
0: Got a packet from TUN
0: Got a packet from TUN
0: Got a packet from TUN
0: Got a packet from TUN
0: Got a packet from TUN
0: Got a packet from TUN
0: Got a packet from the tunnel
0: Got a packet from the tunnel
0: Got a packet from the tunnel
1: New Connection Established
1: New Connection Authenticated
1: Got a packet from the tunnel
1: Got a packet from the tunnel
0: Got a packet from the tunnel
2: New Connection Established
1: Got a packet from the tunnel
2: New Connection Authenticated
2: Got a packet from the tunnel
2: Got a packet from the tunnel
2: Got a packet from the tunnel
1: Got a packet from the tunnel
```

After connecting multiple clients, pinging the private host V still works.

```
seed@SeedUbuntu:~$ ping -c 1 192.168.60.6
PING 192.168.60.6 (192.168.60.6) 56(84) bytes of data.
64 bytes from 192.168.60.6: icmp_seq=1 ttl=63 time=0.938 ms

--- 192.168.60.6 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.938/0.938/0.938/0.000 ms
```