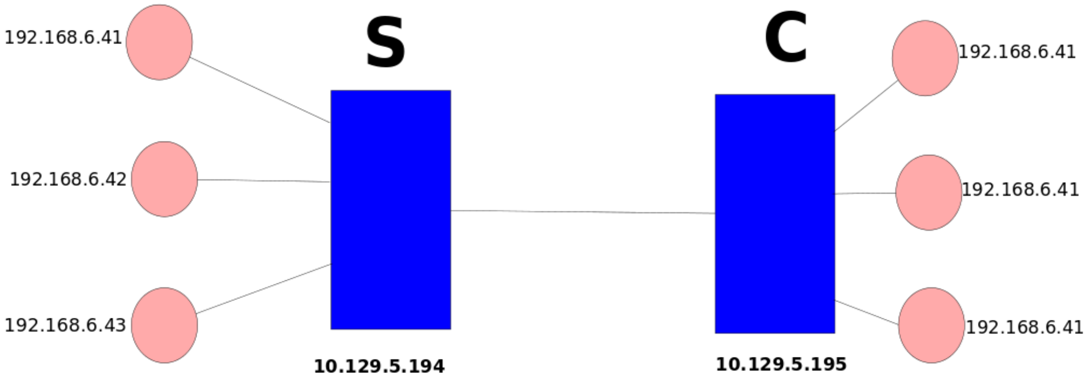
Name: Aniruddh Rao K Roll Num: 133079005
Collaborator: Gaurang Naik

The setup used to implement part 1 and part 2 of PA3

1. A machine C with a physical interface eth0 : IP address 10.129.5.195.
   Machines behind this C machine or C gateway are simulated using virtual interfaces.
   These run clients
   a. eth0:1 with IP 192.168.7.41
   b. eth0:2 with IP 192.168.7.42
   c. eth0:3 with IP 192.168.7.43

2. A machine S with a physical ethernet interface eth0 : IP address 10.129.5.194.
   Machines behind this S machine or S gateway are simulated using virtual interfaces
   Thes run servers.
   a. eth0:1 with IP 192.168.6.41
   b. eth0:2 with IP 192.168.6.42
   c. eth0:3 with IP 192.168.6.43

3. Both machines are connected physically using a ethernet cable.



4. pclient.c file given was used beyond C on virtual interfaces to create and run clients.
   While server1 of PA1 wasmodified to run from particular IP and used to create server on machines beyond S.

Part 1:

IPTABLE rules are written at gateway machines C and S to connect machines beyond them on either sides.
I have written 2 rule on machines C behind which client machines run and have written 1 rule on machine S behind which server is started.
I have written the rules for server running on port 5000.

2 Rules on client side:

One rule to NAT source address of all traffic from client machines IP range 192.168.7.0/24 to IP of client machine C with IP 10.129.5.195 so that it can send the packets to S machine 10.129.5.194 as C and S are connected. MASQUERADE or SNAT can be used for this and I have used SNAT to change the source address here.

Second rule is to NAT the destination address of the packet coming from clients running behind C. Based on which server they are trying to connect to, the destination IP:port map is decided. This is done using DNAT. For connecting to server on 192.168.6.41, the DNAT does change of destination address at C as S IP address and a particular port which is predetermined.

1 Rule at server side:
One rule at destination side or S machine behind which server machines are run is to route the packets that have been NATed at C and come to particular port on S machine which need to be forwarded to respective server to which the client intends to connect to.
Again DNAT is used to filter incoming packets on S on particular ports to be forwarded to machines behind S.
Along with these rules:

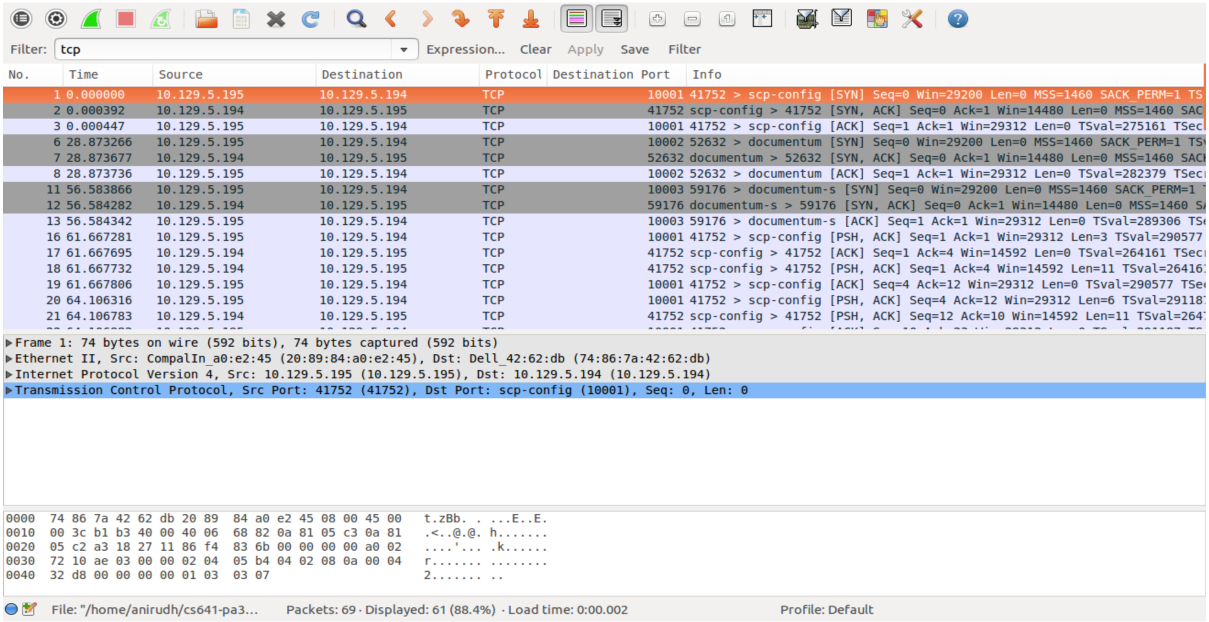
a. A route is defined for all traffic from clients behind C to route through C. This need not be done if a default gateway is defined. Otherwise, define the C machine as gateway to the machines connected behind C. Same is to be done at machines running servers (behind S). For all machines behind S, gateway needs to be defined as S.

b. Configure your C, S machines to be in forwarding/router mode.
This can be done by setting /proc/sys/net/ipv4/ip_forward value as 1

Analysis of PART1 implementation in wireshark:

TCP Connection is established from C i.e 10.129.5.195 to S 10.129.5.194 whenever a client running behind C tries to connect to a server running behind S. Connecting 3 clients behind c to 3 servers behind S shows 3 new connections in the capture.

At client



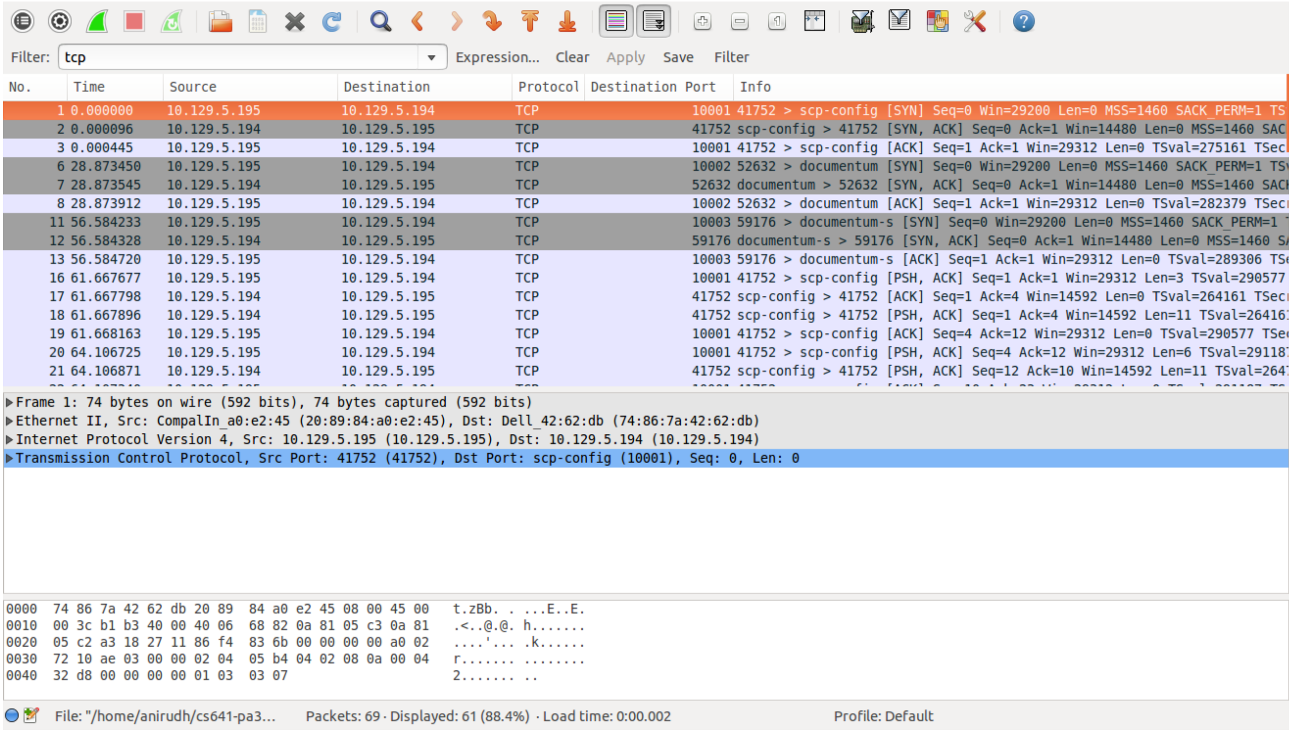
```
No.     Time         Source          Destination     Protocol Destination Port Info
      1 0.000000     10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TS
      2 0.000392     10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SAC
      3 0.000447     10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=275161 TSec
      6 28.873266    10.129.5.195    10.129.5.194    TCP                10002 52632 > documentum [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TS
      7 28.873677    10.129.5.194    10.129.5.195    TCP                52632 documentum > 52632 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SAC
      8 28.873736    10.129.5.195    10.129.5.194    TCP                10002 52632 > documentum [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=282379 TSec
     11 56.583866    10.129.5.195    10.129.5.194    TCP                10003 59176 > documentum-s [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
     12 56.584282    10.129.5.194    10.129.5.195    TCP                59176 documentum-s > 59176 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 S
     13 56.584342    10.129.5.195    10.129.5.194    TCP                10003 59176 > documentum-s [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=289306 TS
     16 61.667281    10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=3 TSval=290577
     17 61.667695    10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [ACK] Seq=1 Ack=4 Win=14592 Len=0 TSval=264161 TSec
     18 61.667732    10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [PSH, ACK] Seq=1 Ack=4 Win=14592 Len=11 TSval=26416
     19 61.667806    10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [ACK] Seq=4 Ack=12 Win=29312 Len=0 TSval=290577 TSe
     20 64.106316    10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [PSH, ACK] Seq=4 Ack=12 Win=29312 Len=6 TSval=29118
     21 64.106783    10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [PSH, ACK] Seq=12 Ack=10 Win=14592 Len=11 TSval=264
```

```
▶Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▶Ethernet II, Src: CompalIn_a0:e2:45 (20:89:84:a0:e2:45), Dst: Dell_42:62:db (74:86:7a:42:62:db)
▶Internet Protocol Version 4, Src: 10.129.5.195 (10.129.5.195), Dst: 10.129.5.194 (10.129.5.194)
▶Transmission Control Protocol, Src Port: 41752 (41752), Dst Port: scp-config (10001), Seq: 0, Len: 0
```

```
0000  74 86 7a 42 62 db 20 89  84 a0 e2 45 08 00 45 00   t.zBb. . ...E..E.
0010  00 3c b1 b3 40 00 40 06  68 82 0a 81 05 c3 0a 81   .<..@.@. h.......
0020  05 c2 a3 18 27 11 86 f4  83 6b 00 00 00 00 a0 02   ....'... .k......
0030  72 10 ae 03 00 00 02 04  05 b4 04 02 08 0a 00 04   r....... ........
0040  32 d8 00 00 00 00 01 03  03 07                     2....... ..
```

File: "/home/anirudh/cs641-pa3...   Packets: 69 · Displayed: 61 (88.4%) · Load time: 0:00.002   Profile: Default

Also I had used 10001, 10002, 10003 as ports at S to connect to 3 servers in and defined the same in iptable rules. Screen shot below shows the same



```
No.     Time         Source          Destination     Protocol Destination Port Info
      1 0.000000     10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TS
      2 0.000096     10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SAC
      3 0.000445     10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=275161 TSec
      6 28.873450    10.129.5.195    10.129.5.194    TCP                10002 52632 > documentum [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TS
      7 28.873545    10.129.5.194    10.129.5.195    TCP                52632 documentum > 52632 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SAC
      8 28.873912    10.129.5.195    10.129.5.194    TCP                10002 52632 > documentum [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=282379 TSec
     11 56.584233    10.129.5.195    10.129.5.194    TCP                10003 59176 > documentum-s [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
     12 56.584328    10.129.5.194    10.129.5.195    TCP                59176 documentum-s > 59176 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 S
     13 56.584720    10.129.5.195    10.129.5.194    TCP                10003 59176 > documentum-s [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=289306 TS
     16 61.667677    10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=3 TSval=290577
     17 61.667798    10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [ACK] Seq=1 Ack=4 Win=14592 Len=0 TSval=264161 TSec
     18 61.667896    10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [PSH, ACK] Seq=1 Ack=4 Win=14592 Len=11 TSval=26416
     19 61.668163    10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [ACK] Seq=4 Ack=12 Win=29312 Len=0 TSval=290577 TSe
     20 64.106725    10.129.5.195    10.129.5.194    TCP                10001 41752 > scp-config [PSH, ACK] Seq=4 Ack=12 Win=29312 Len=6 TSval=29118
     21 64.106871    10.129.5.194    10.129.5.195    TCP                41752 scp-config > 41752 [PSH, ACK] Seq=12 Ack=10 Win=14592 Len=11 TSval=264
```

```
▶Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▶Ethernet II, Src: CompalIn_a0:e2:45 (20:89:84:a0:e2:45), Dst: Dell_42:62:db (74:86:7a:42:62:db)
▶Internet Protocol Version 4, Src: 10.129.5.195 (10.129.5.195), Dst: 10.129.5.194 (10.129.5.194)
▶Transmission Control Protocol, Src Port: 41752 (41752), Dst Port: scp-config (10001), Seq: 0, Len: 0
```

```
0000  74 86 7a 42 62 db 20 89  84 a0 e2 45 08 00 45 00   t.zBb. . ...E..E.
0010  00 3c b1 b3 40 00 40 06  68 82 0a 81 05 c3 0a 81   .<..@.@. h.......
0020  05 c2 a3 18 27 11 86 f4  83 6b 00 00 00 00 a0 02   ....'... .k......
0030  72 10 ae 03 00 00 02 04  05 b4 04 02 08 0a 00 04   r....... ........
0040  32 d8 00 00 00 00 01 03  03 07                     2....... ..
```

File: "/home/anirudh/cs641-pa3...   Packets: 69 · Displayed: 61 (88.4%) · Load time: 0:00.002   Profile: Default

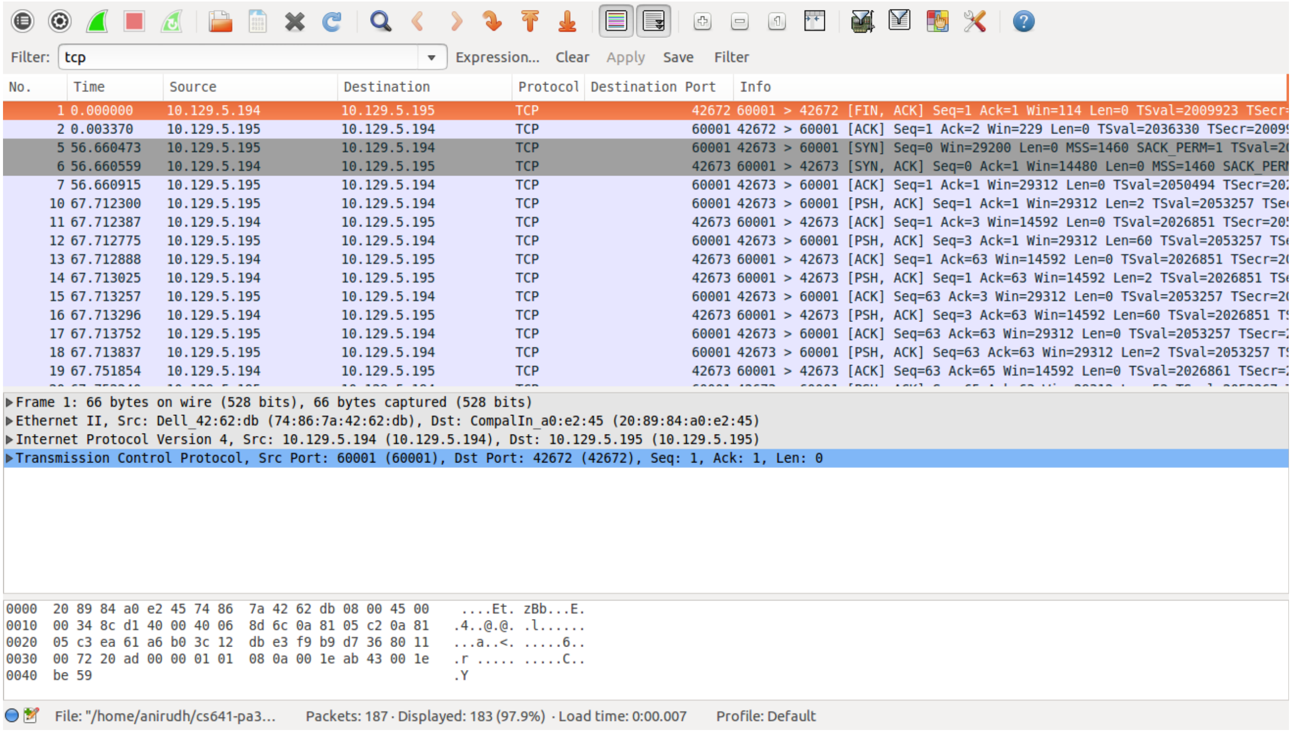
Part 2

In this part, 2 tun devices are created on either sides i.e S and on C.

Any traffic from machines running clients and destined to IP range192.168.6.0/24 (where servers run) is routed through tun device at C.
Similarly any traffic from server running machines destined to IP range of clients behind C i.e. 192.168.7.0/24 range is routed through tundevice at S

Then, A client c program tunc.c is run at C and tuns.c is run at S. The C programs take care of writing from tun device to eth0 and reading from eth0 to tun device at either ends.
The tunc and tuns codes are kept running and then we can communicate from clients behind C to server S by running servers and clients.
The screen shots explain that only one TCP connection is established between S and C.
All other connections are tunneled through this. I have captured tcp dump at both S, C as well as tun devices ceated at the end.
At tun devices tcp connections are established from a client behind C to server behind S for every client-server pair trying to communicate. Whereas there is only one TCP connection from S to C, which is persistant till the tunnel runs.

ScreenShot of TCP dump captured at S:



Filter: tcp

```
No.   Time        Source         Destination    Protocol Destination Port Info
    1 0.000000    10.129.5.194   10.129.5.195   TCP                42672 60001 > 42672 [FIN, ACK] Seq=1 Ack=1 Win=114 Len=0 TSval=2009923 TSecr=
    2 0.003370    10.129.5.195   10.129.5.194   TCP                60001 42672 > 60001 [ACK] Seq=1 Ack=2 Win=229 Len=0 TSval=2036330 TSecr=2009
    5 56.660473   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=2
    6 56.660559   10.129.5.194   10.129.5.195   TCP                42673 60001 > 42673 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460 SACK_PERM
    7 56.660915   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2050494 TSecr=202
   10 67.712300   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=2 TSval=2053257 TSe
   11 67.712387   10.129.5.194   10.129.5.195   TCP                42673 60001 > 42673 [ACK] Seq=1 Ack=3 Win=14592 Len=0 TSval=2026851 TSecr=205
   12 67.712775   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [PSH, ACK] Seq=3 Ack=1 Win=29312 Len=60 TSval=2053257 TS
   13 67.712888   10.129.5.194   10.129.5.195   TCP                42673 60001 > 42673 [ACK] Seq=1 Ack=63 Win=14592 Len=0 TSval=2026851 TSecr=2
   14 67.713025   10.129.5.194   10.129.5.195   TCP                42673 60001 > 42673 [PSH, ACK] Seq=1 Ack=63 Win=14592 Len=2 TSval=2026851 TS
   15 67.713257   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [ACK] Seq=63 Ack=3 Win=29312 Len=0 TSval=2053257 TSecr=2
   16 67.713296   10.129.5.194   10.129.5.195   TCP                42673 60001 > 42673 [PSH, ACK] Seq=3 Ack=63 Win=14592 Len=60 TSval=2026851 T
   17 67.713752   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [ACK] Seq=63 Ack=63 Win=29312 Len=0 TSval=2053257 TSecr=
   18 67.713837   10.129.5.195   10.129.5.194   TCP                60001 42673 > 60001 [PSH, ACK] Seq=63 Ack=63 Win=29312 Len=2 TSval=2053257 T
   19 67.751854   10.129.5.194   10.129.5.195   TCP                42673 60001 > 42673 [ACK] Seq=63 Ack=65 Win=14592 Len=0 TSval=2026861 TSecr=
```

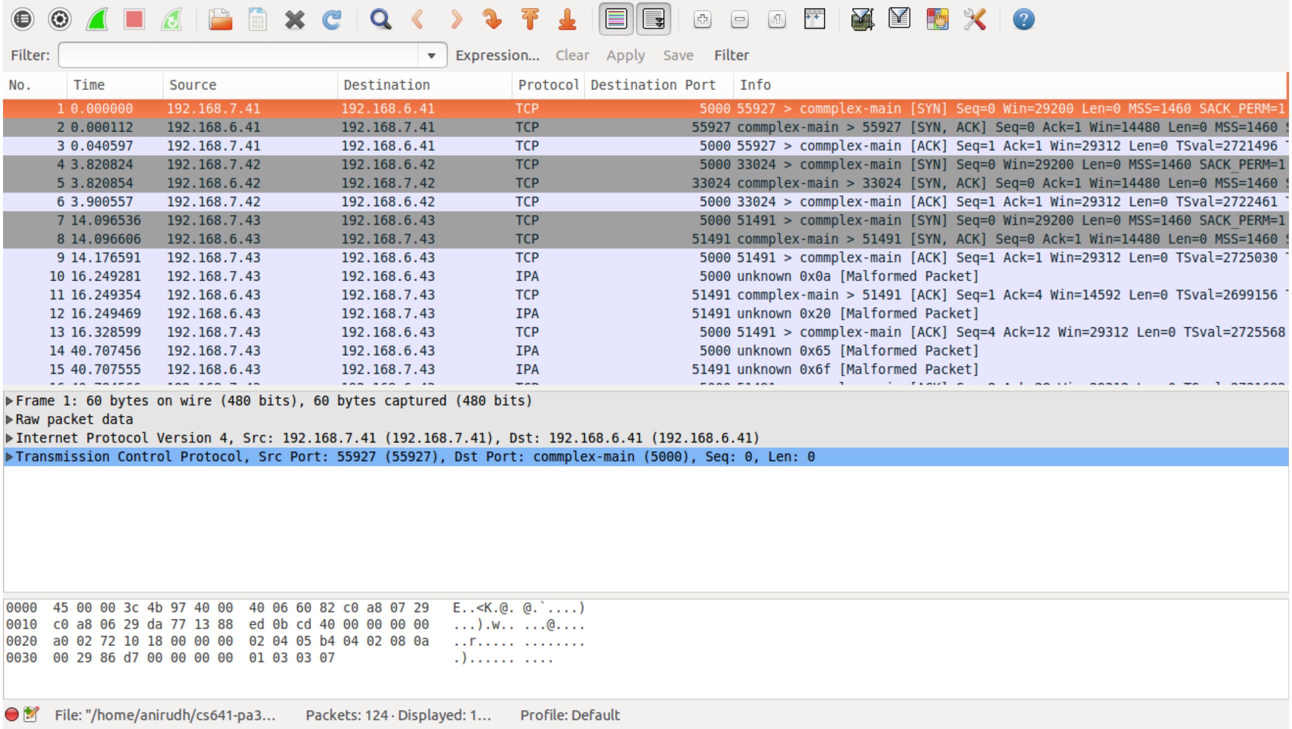
```
▶Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)
▶Ethernet II, Src: Dell_42:62:db (74:86:7a:42:62:db), Dst: CompalIn_a0:e2:45 (20:89:84:a0:e2:45)
▶Internet Protocol Version 4, Src: 10.129.5.194 (10.129.5.194), Dst: 10.129.5.195 (10.129.5.195)
▶Transmission Control Protocol, Src Port: 60001 (60001), Dst Port: 42672 (42672), Seq: 1, Ack: 1, Len: 0
```

```
0000  20 89 84 a0 e2 45 74 86  7a 42 62 db 08 00 45 00    ....Et. zBb...E.
0010  00 34 8c d1 40 00 40 06  8d 6c 0a 81 05 c2 0a 81   .4..@.@. .l......
0020  05 c3 ea 61 a6 b0 3c 12  db e3 f9 b9 d7 36 80 11   ...a..<. .....6..
0030  00 72 20 ad 00 00 01 01  08 0a 00 1e ab 43 00 1e   .r ..... .....C..
0040  be 59                                              .Y
```

File: "/home/anirudh/cs641-pa3...   Packets: 187 · Displayed: 183 (97.9%) · Load time: 0:00.007   Profile: Default

Shows only one TCP connection running between 10.129.5.194 and 10.129.5.195
 happens when a tunnel is created.

Screenshot of TCP dump at tun device at C:



```
No.   Time       Source         Destination    Protocol Destination Port  Info
    1 0.000000   192.168.7.41   192.168.6.41   TCP                 5000 55927 > commplex-main [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
    2 0.000112   192.168.6.41   192.168.7.41   TCP                55927 commplex-main > 55927 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460
    3 0.040597   192.168.7.41   192.168.6.41   TCP                 5000 55927 > commplex-main [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2721496
    4 3.820824   192.168.7.42   192.168.6.42   TCP                 5000 33024 > commplex-main [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
    5 3.820854   192.168.6.42   192.168.7.42   TCP                33024 commplex-main > 33024 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460
    6 3.900557   192.168.7.42   192.168.6.42   TCP                 5000 33024 > commplex-main [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2722461
    7 14.096536  192.168.7.43   192.168.6.43   TCP                 5000 51491 > commplex-main [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1
    8 14.096606  192.168.6.43   192.168.7.43   TCP                51491 commplex-main > 51491 [SYN, ACK] Seq=0 Ack=1 Win=14480 Len=0 MSS=1460
    9 14.176591  192.168.7.43   192.168.6.43   TCP                 5000 51491 > commplex-main [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=2725030
   10 16.249281  192.168.7.43   192.168.6.43   IPA                 5000 unknown 0x0a [Malformed Packet]
   11 16.249354  192.168.6.43   192.168.7.43   TCP                51491 commplex-main > 51491 [ACK] Seq=1 Ack=4 Win=14592 Len=0 TSval=2699156
   12 16.249469  192.168.6.43   192.168.7.43   IPA                51491 unknown 0x20 [Malformed Packet]
   13 16.328599  192.168.7.43   192.168.6.43   TCP                 5000 51491 > commplex-main [ACK] Seq=4 Ack=12 Win=29312 Len=0 TSval=2725568
   14 40.707456  192.168.7.43   192.168.6.43   IPA                 5000 unknown 0x65 [Malformed Packet]
   15 40.707555  192.168.6.43   192.168.7.43   IPA                51491 unknown 0x6f [Malformed Packet]
```

```
▶Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
▶Raw packet data
▶Internet Protocol Version 4, Src: 192.168.7.41 (192.168.7.41), Dst: 192.168.6.41 (192.168.6.41)
▶Transmission Control Protocol, Src Port: 55927 (55927), Dst Port: commplex-main (5000), Seq: 0, Len: 0
```

```
0000  45 00 00 3c 4b 97 40 00  40 06 60 82 c0 a8 07 29   E..<K.@. @.`....)
0010  c0 a8 06 29 da 77 13 88  ed 0b cd 40 00 00 00 00   ...).w.. ...@....
0020  a0 02 72 10 18 00 00 00  02 04 05 b4 04 02 08 0a   ..r..... ........
0030  00 29 86 d7 00 00 00 00  01 03 03 07               .)...... ....
```

File: "/home/anirudh/cs641-pa3... Packets: 124 · Displayed: 1... Profile: Default

Show TCP connection established for evervy client server connection.

192.168.7.41 --> 192.168.6.41:5000
192.168.7.42 --> 192.168.6.42:5000
192.168.7.43 --> 192.168.6.43:5000

Comparision of Solutions in part1 and part2:

The solution in part 2 is more elegant and flexible compared to the solution in part1. The solution in part 1 is more rigid as iptables rules are bound to particular IP and port.

Also each connection of client-server eats up free ports of C and S in part1 where as it would use just one tcp connection in case of tunneling using solution in part2.

Since solution in part2 uses just one TCP connection , it may also affect the preformance if more client-servers tunnel through gatway C-S.