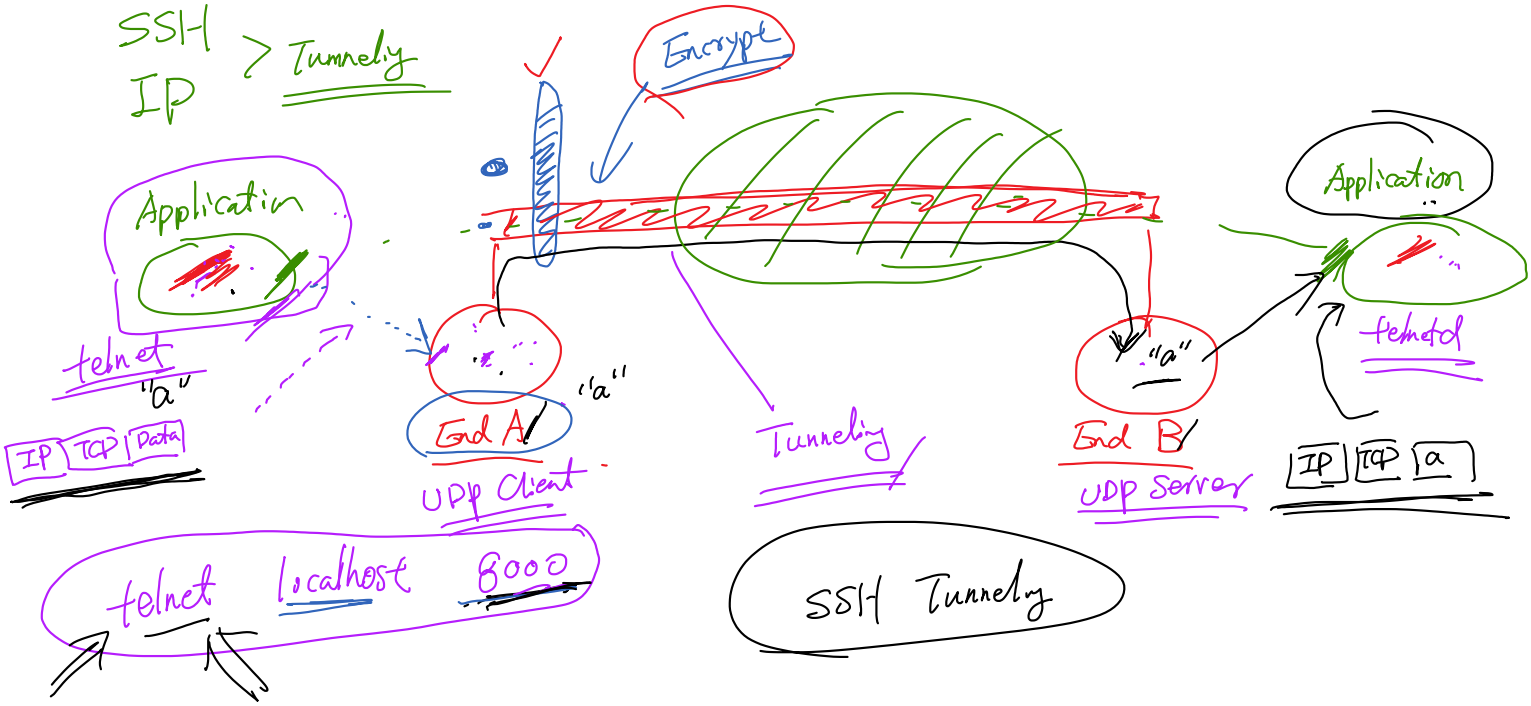
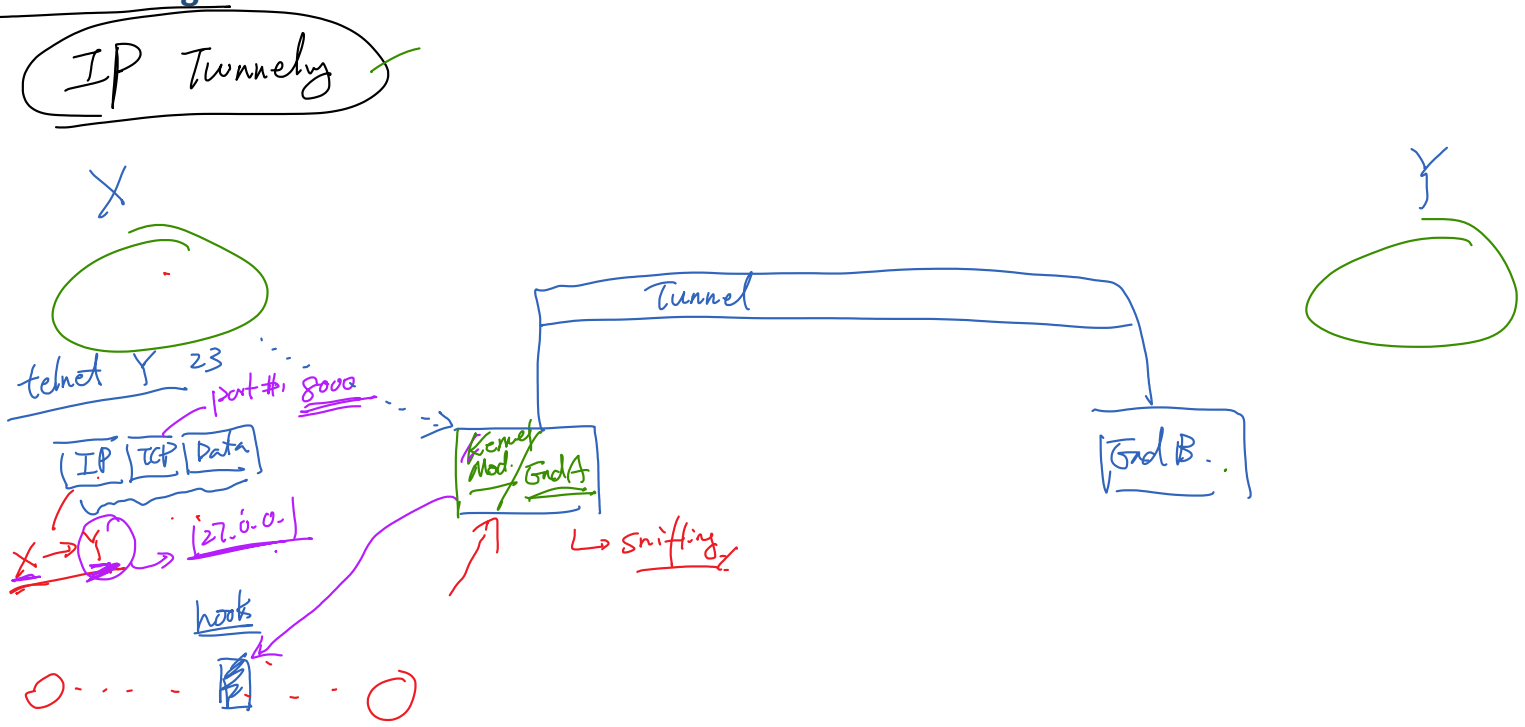
Internet Security

Virtual Private Network

Network Tunneling

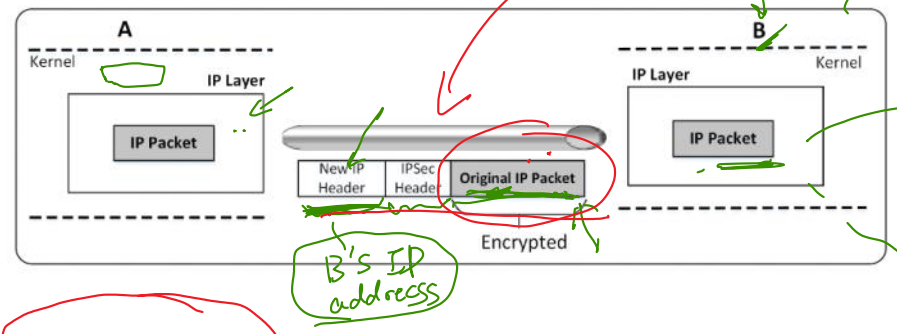


SSH Tunneling

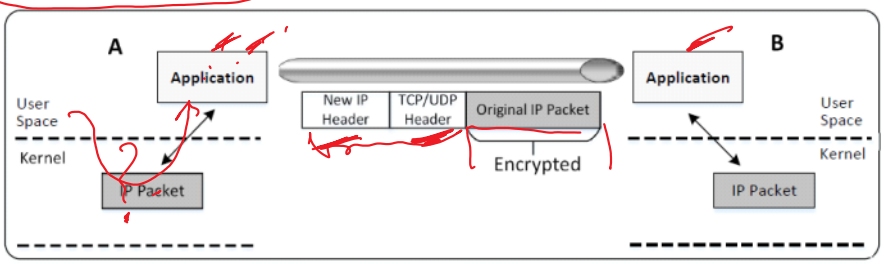


IP Tunneling

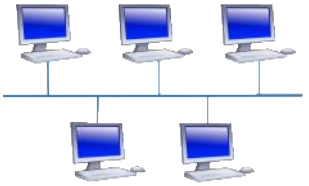
❖ IPSec Approach



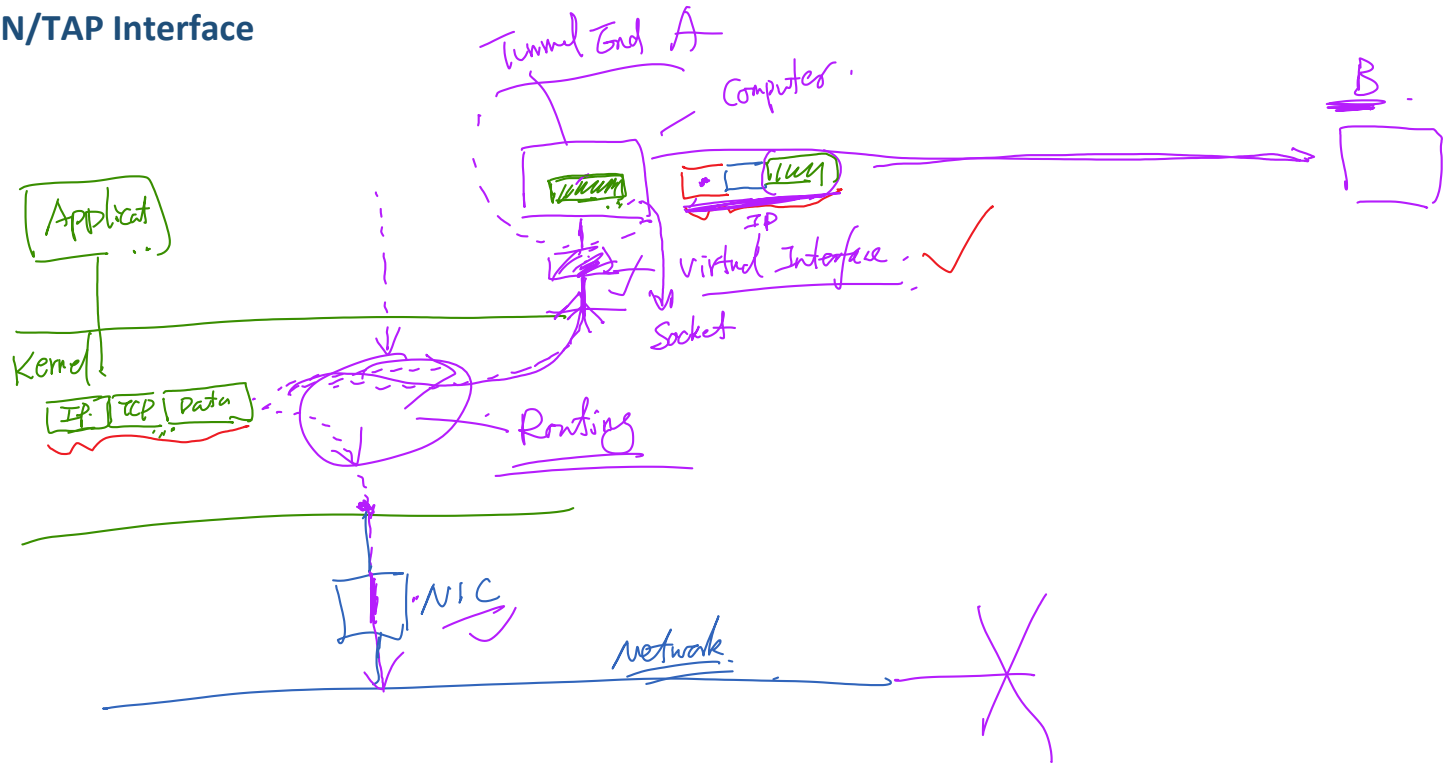
❖ SSL/TLS Approach



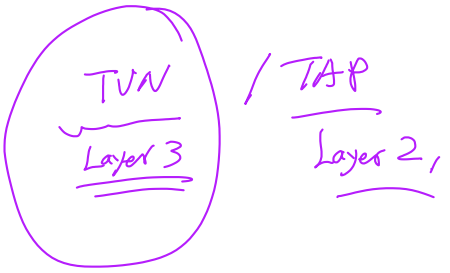
Why Virtual Private Network (VPN)?



The TUN/TAP Interface



Creating TUN/TAP Interface



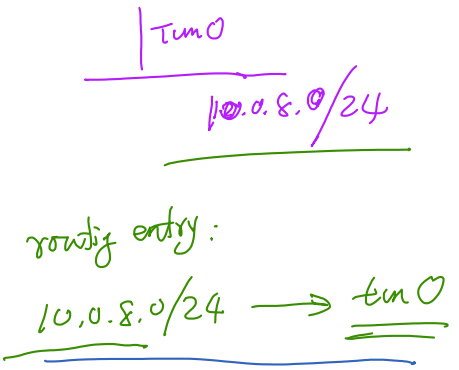
❖ Creating a TUN Interface

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

   ifr.ifr_flags = IFF_TUN | IFF_NO_PI;  ①

   tunfd = open("/dev/net/tun", O_RDWR); ②
   ioctl(tunfd, TUNSETIFF, &ifr);        ③

   printf("TUN file descriptor: %d \n", tunfd);
   // We can interact with the device using this file descriptor.
   // In our experiement, we will do the interaction from a shell.
   // Therefore, we launch the bash shell here.
   execve("/bin/bash", NULL,NULL);       ④

   return 0;
}
```

❖ Configure the TUN Interface

```
% ifconfig -a
tun0 Link encap:UNSPEC HWaddr 00-00-00 ...
     POINTOPOINT NOARP MULTICAST MTU:1500 ...
```

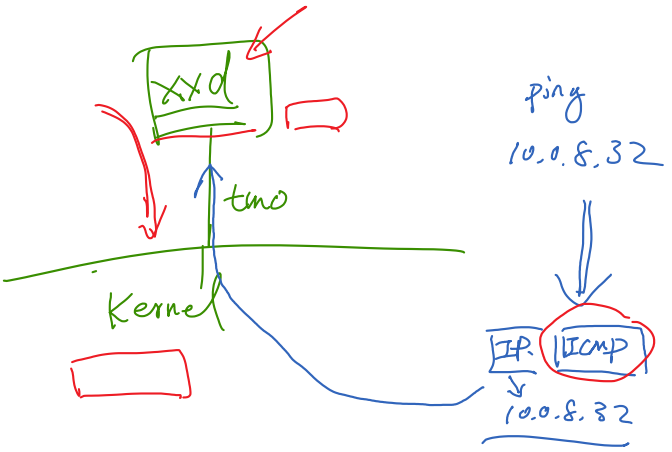
```
% sudo ifconfig tun0 10.0.8.99/24 up

% ifconfig
tun0 Link encap:UNSPEC HWaddr 00-00-00 ...
     inet addr:10.0.8.99 P-t-P:10.0.8.99 Mask:255.255.255.0
     UP POINTOPOINT RUNNING NOARP MULTICAST MTU:1500 ...
```



Read From and Write to the TUN Interface

❖ **Read from the TUN interface (`ping 10.0.8.32`).**

```
$ sudo ./tundemo
TUN file descriptor: 3

# xxd <& 3
0000000: 4500 0054 0000 4000 4001 1627 0a00 0863  E..T..@.@..'...c
0000010: 0a00 0820 0800 3b19 10cf 0001 da1d 9f57  ... ..;........W
0000020: 439e 0400 0809 0a0b 0c0d 0e0f 1011 1213  C...............
0000030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223  ............ !"#
0000040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233  $%&'()*+,-./0123
0000050: 3435 3637
```



In Programs

```
len = read(tunfd, buff, BUFF_SIZE);
```

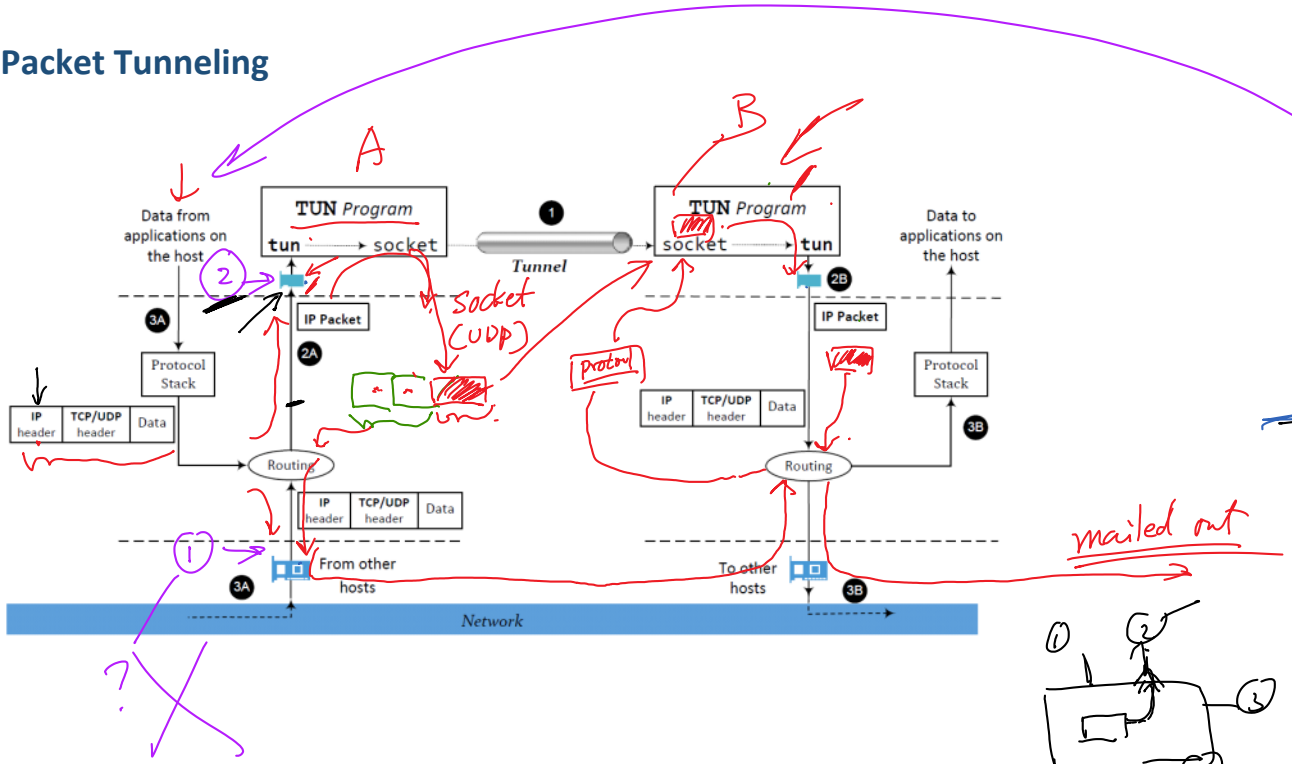
❖ **Write to the TUN interface.**

```
# cat packetfile >& 3
```

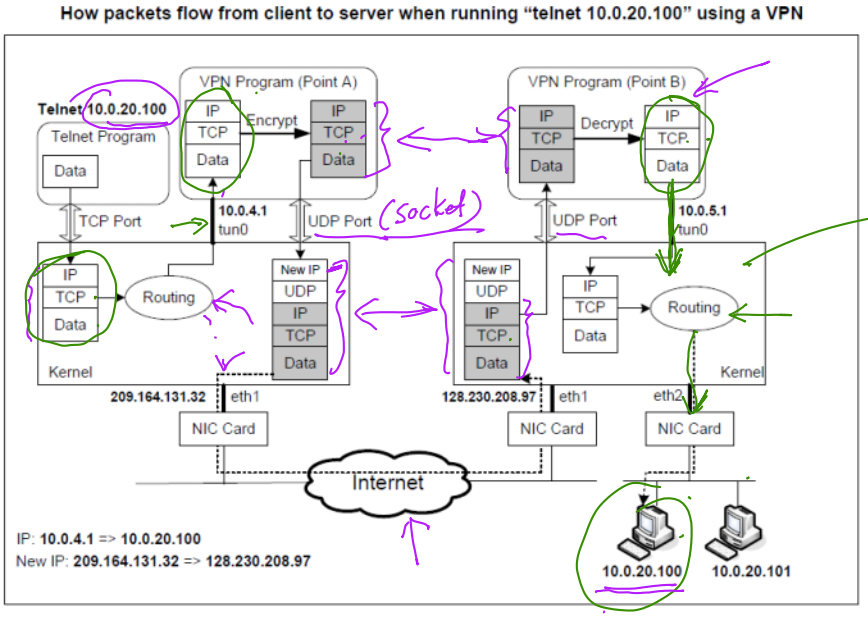
In Programs

```
write(tunfd, buff, len);
```

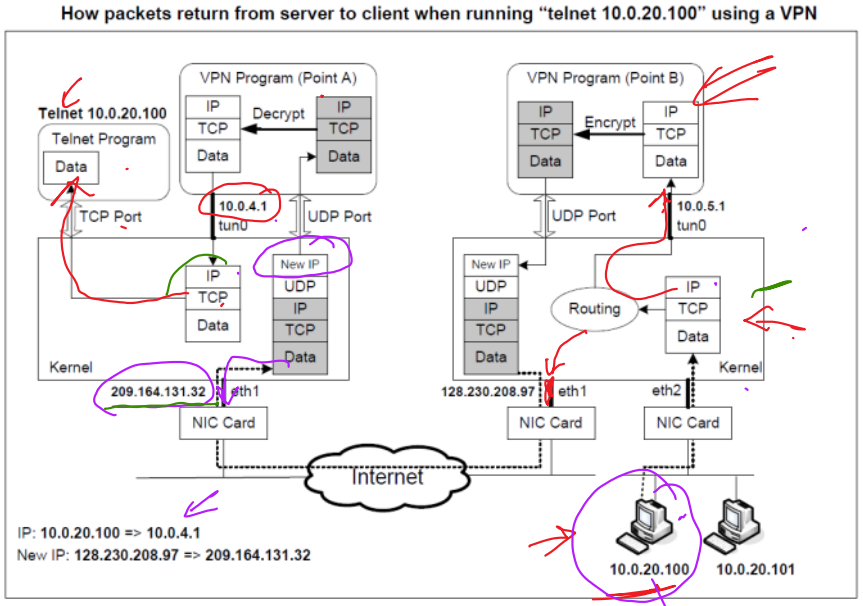
Packet Tunneling



How VPN Works: Outgoing Traffic



How VPN Works: Return Traffic



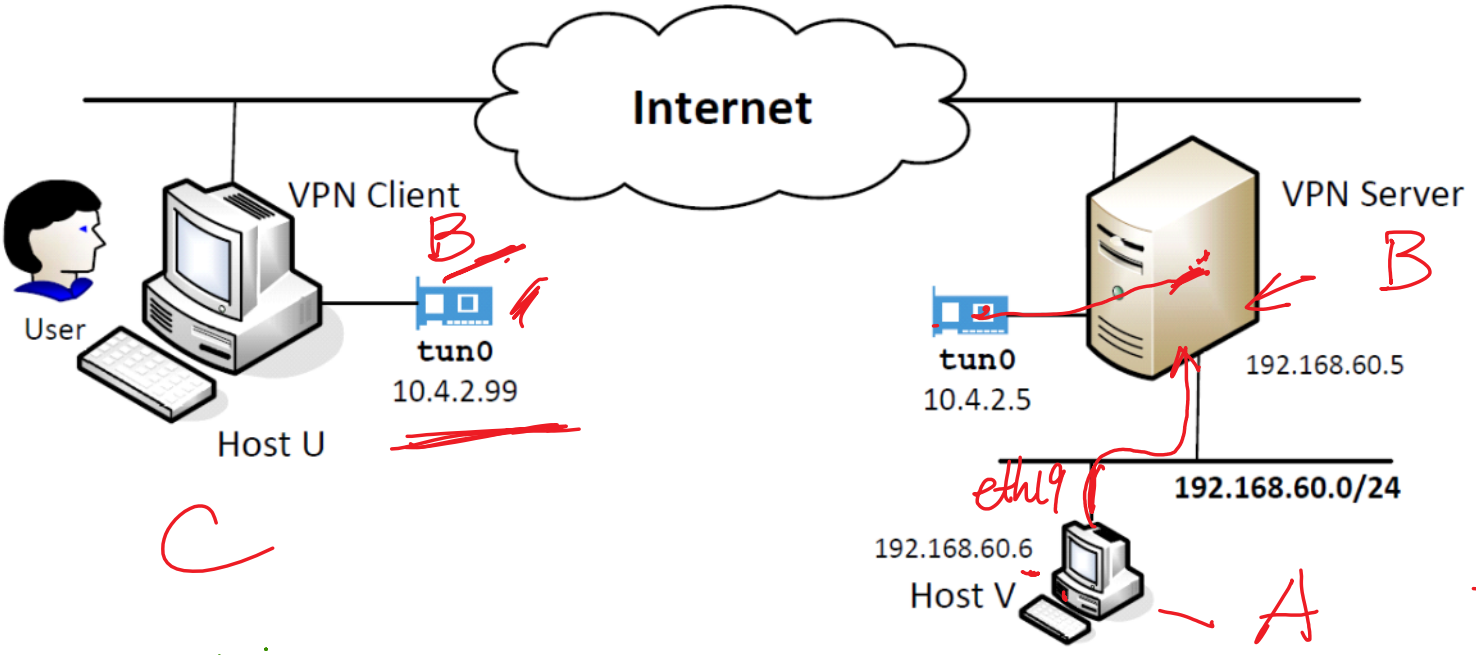
Reply SRC IP + Routing

F: Routing

SRC IP = 10.0.4.1

SRC port : 9000

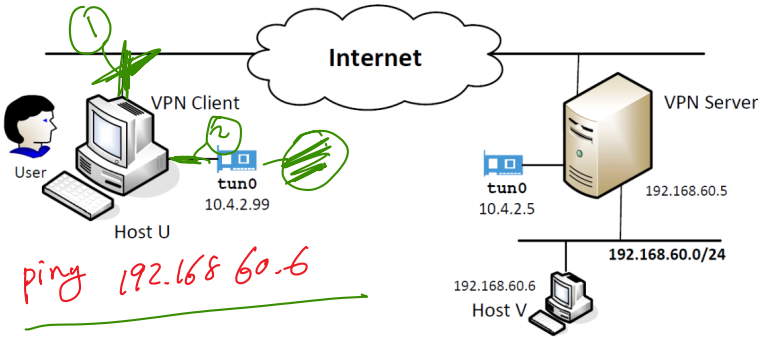
Question: Network Setup



Question: Where should we run the following commands?

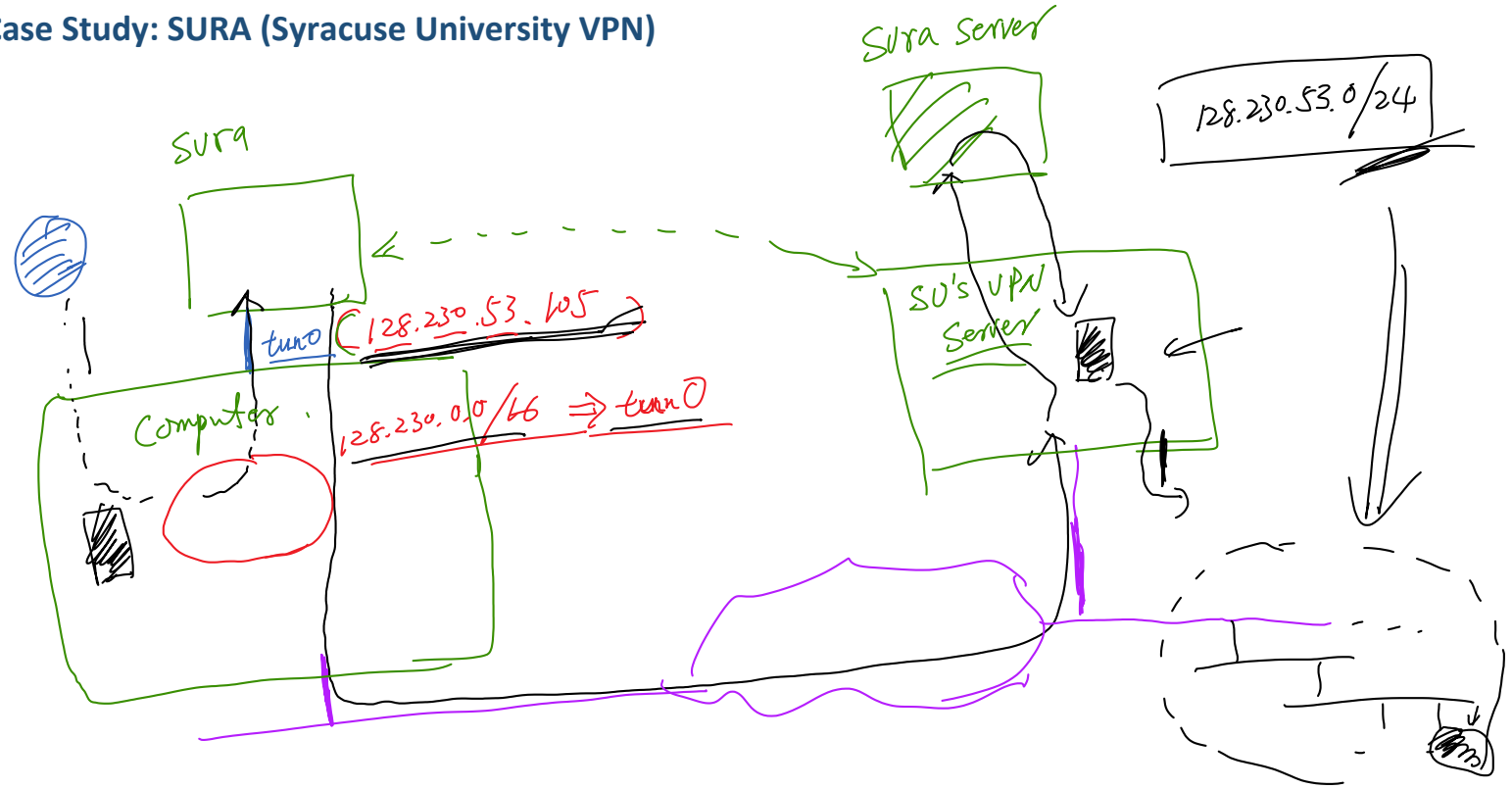
```
A: $ sudo route add -net 10.4.2.0/24 gw 192.168.60.5 eth19
B: $ sudo route add -net 10.4.2.0/24 tun0
C: $ sudo route add -net 192.168.60.0/24 tun0
```

Testing VPN



No.	Source	Destination	Protocol	Length	Info
1	10.4.2.99	192.168.60.6	ICMP	100	Echo (ping) request id=0x0e85, seq=1/256, ttl=64
2	10.0.2.6	10.0.2.5	UDP	128	Source port: 59793 Destination port: 55555
3	10.0.2.5	10.0.2.6	UDP	128	Source port: 55555 Destination port: 59793
4	192.168.60.6	10.4.2.99	ICMP	100	Echo (ping) reply id=0x0e85, seq=1/256, ttl=63
5	10.4.2.99	192.168.60.6	ICMP	100	Echo (ping) request id=0x0e85, seq=2/512, ttl=64
6	10.0.2.6	10.0.2.5	UDP	128	Source port: 59793 Destination port: 55555
7	10.0.2.5	10.0.2.6	UDP	128	Source port: 55555 Destination port: 59793
8	192.168.60.6	10.4.2.99	ICMP	100	Echo (ping) reply id=0x0e85, seq=2/512, ttl=63

Case Study: SURA (Syracuse University VPN)



SURA: Before Running VPN

❖ Interfaces

```
PS C:\Users\kevin> ipconfig

Windows IP Configuration


Wireless LAN adapter Wireless Network Connection 2:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Wireless Network Connection:

   Connection-specific DNS Suffix  . : syr.edu
   Link-local IPv6 Address . . . . . : fe80::30c5:d02c:ed1d:2d2e%13
   IPv4 Address. . . . . . . . . . . : 10.1.56.64
   Subnet Mask . . . . . . . . . . . : 255.255.192.0
   Default Gateway . . . . . . . . . : 10.1.0.1
```

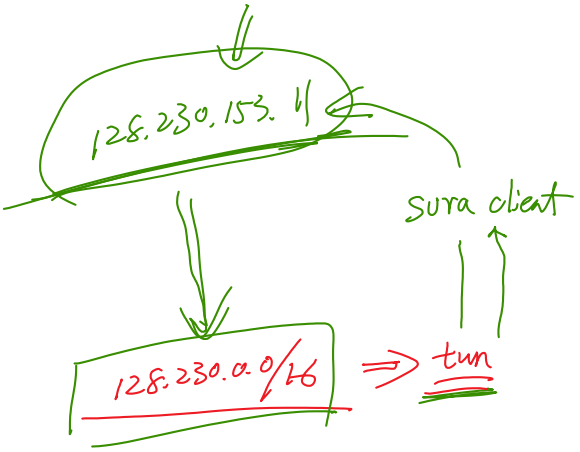
❖ Routing table (Windows: Route PRINT)

```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0       10.1.0.1        10.1.56.64     25
         10.1.0.0    255.255.192.0         On-link        10.1.56.64    281
       10.1.56.64  255.255.255.255         On-link        10.1.56.64    281
      10.1.63.255  255.255.255.255         On-link        10.1.56.64    281
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    306
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    306
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    306
    192.168.147.0    255.255.255.0         On-link     192.168.147.1    276
    192.168.147.1  255.255.255.255         On-link     192.168.147.1    276
  192.168.147.255  255.255.255.255         On-link     192.168.147.1    276
        224.0.0.0        240.0.0.0         On-link         127.0.0.1    306
        224.0.0.0        240.0.0.0         On-link     192.168.147.1    276
        224.0.0.0        240.0.0.0         On-link        10.1.56.64    281
  255.255.255.255  255.255.255.255         On-link         127.0.0.1    306
  255.255.255.255  255.255.255.255         On-link     192.168.147.1    276
  255.255.255.255  255.255.255.255         On-link        10.1.56.64    281
===========================================================================
```

SURA: After Running VPN

❖ Interfaces

```
PS C:\Users\kevin> ipconfig

Windows IP Configuration

PPP adapter Syracuse University Remote Access VPN:

   Connection-specific DNS Suffix  . :
   IPv4 Address. . . . . . . . . . . : 128.230.153.98
   Subnet Mask . . . . . . . . . . . : 255.255.255.255
   Default Gateway . . . . . . . . . :

Wireless LAN adapter Wireless Network Connection 2:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Wireless LAN adapter Wireless Network Connection:

   Connection-specific DNS Suffix  . : syr.edu
   Link-local IPv6 Address . . . . . : fe80::30c5:d02c:ed1d:2d2e%13
   IPv4 Address. . . . . . . . . . . : 10.1.56.64
   Subnet Mask . . . . . . . . . . . : 255.255.192.0
   Default Gateway . . . . . . . . . : 10.1.0.1
```



❖ Routing table

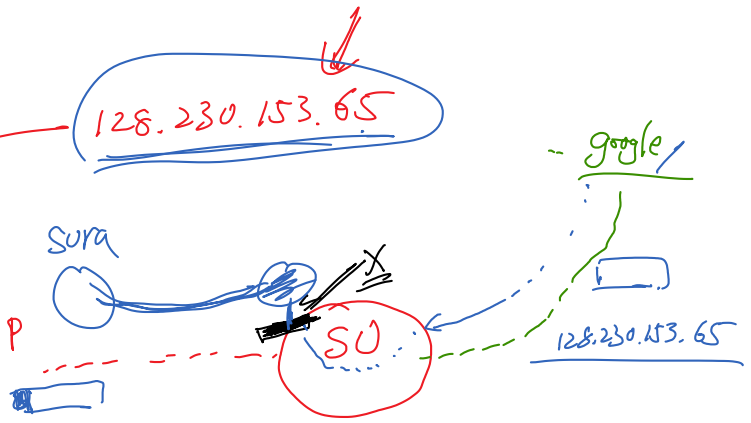
```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0         10.1.0.1       10.1.56.64     25
         10.1.0.0    255.255.192.0         On-link       10.1.56.64    281
       10.1.56.64  255.255.255.255         On-link       10.1.56.64    281
       10.1.63.255  255.255.255.255         On-link       10.1.56.64    281
        127.0.0.0        255.0.0.0         On-link        127.0.0.1    306
        127.0.0.1  255.255.255.255         On-link        127.0.0.1    306
  127.255.255.255  255.255.255.255         On-link        127.0.0.1    306
      128.230.0.0      255.255.0.0   128.230.153.30   128.230.153.98     21
    128.230.153.11  255.255.255.255         10.1.0.1       10.1.56.64     26
   128.230.153.98  255.255.255.255         On-link    128.230.153.98    276
     192.168.147.0    255.255.255.0         On-link     192.168.147.1    276
     192.168.147.1  255.255.255.255         On-link     192.168.147.1    276
   192.168.147.255  255.255.255.255         On-link     192.168.147.1    276
         224.0.0.0        240.0.0.0         On-link        127.0.0.1    306
         224.0.0.0        240.0.0.0         On-link     192.168.147.1    276
         224.0.0.0        240.0.0.0         On-link        10.1.56.64    281
         224.0.0.0        240.0.0.0         On-link    128.230.153.98    276
   255.255.255.255  255.255.255.255         On-link        127.0.0.1    306
   255.255.255.255  255.255.255.255         On-link     192.168.147.1    276
   255.255.255.255  255.255.255.255         On-link        10.1.56.64    281
   255.255.255.255  255.255.255.255         On-link    128.230.153.98    276
===========================================================================
```

Question: Find the IP Addresses

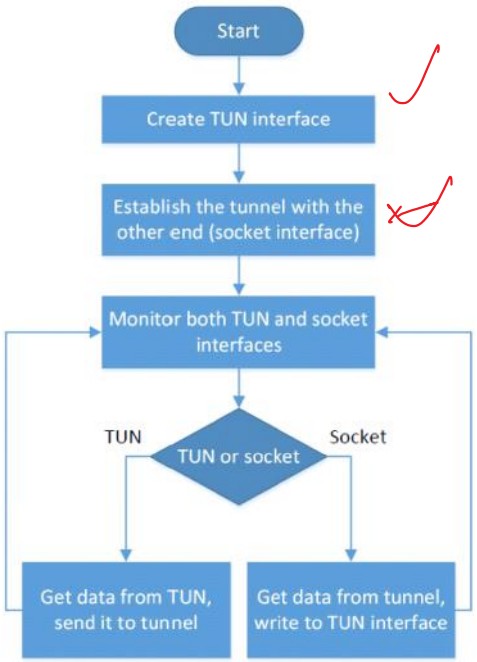
SU's VPN is called SURA. If you run SURA on your computer, once you have logged in, a VPN tunnel will be established between your host machine and SU's network (128.230.0.0/16). After I run SURA, the routing table on my computer appears as in the picture below. Please answer the following questions.

1. What is my computer's real IP address (i.e., the IP address of my WiFi card)?
2. What is the IP address of the VPN server?
3. What is the IP address of my TUN interface?

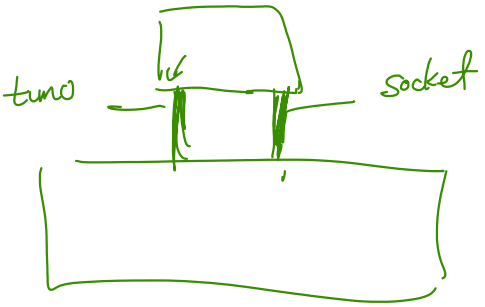
```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      192.168.0.1    192.168.0.13     25
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    306
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    306
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    306
      128.230.0.0      255.255.0.0  128.230.153.30   128.230.153.65     21
   128.230.153.11  255.255.255.255      192.168.0.1     192.168.0.13     26
   128.230.153.65  255.255.255.255         On-link   128.230.153.65    276
      192.168.0.0    255.255.255.0         On-link     192.168.0.13    281
     192.168.0.13  255.255.255.255         On-link     192.168.0.13    281
    192.168.0.255  255.255.255.255         On-link     192.168.0.13    281
     192.168.56.0    255.255.255.0         On-link     192.168.56.1    266
     192.168.56.1  255.255.255.255         On-link     192.168.56.1    266
   192.168.56.255  255.255.255.255         On-link     192.168.56.1    266
     192.168.60.0    255.255.255.0         On-link     192.168.60.1    266
     192.168.60.1  255.255.255.255         On-link     192.168.60.1    266
   192.168.60.255  255.255.255.255         On-link     192.168.60.1    266
    192.168.200.0    255.255.255.0         On-link    192.168.200.1    266
    192.168.200.1  255.255.255.255         On-link    192.168.200.1    266
  192.168.200.255  255.255.255.255         On-link    192.168.200.1    266
```



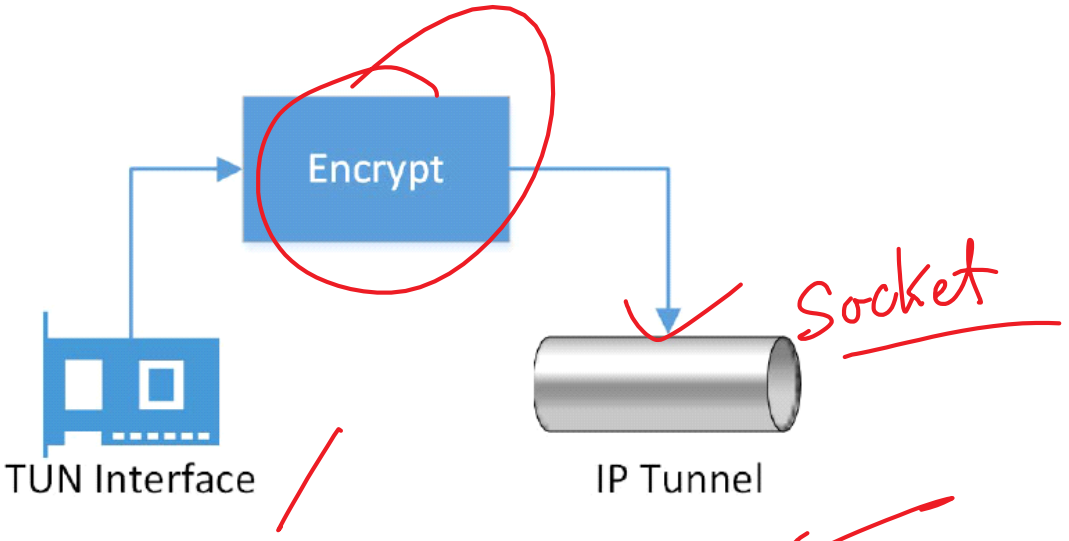
Programming VPN




```
fd_set readFDSet;
int ret, sockfd, tunfd;

FD_ZERO(&readFDSet);
FD_SET(sockfd, &readFDSet);
FD_SET(tunfd, &readFDSet);
ret = select(FD_SETSIZE, &readFDSet, NULL, NULL, NULL);

if (FD_ISSET(sockfd, &readFDSet)){
   // Read from sockfd and write to tunfd
}

if (FD_ISSET(tunfd, &readFDSet)){
   // Read from tunfd and write to sockfd
}
```

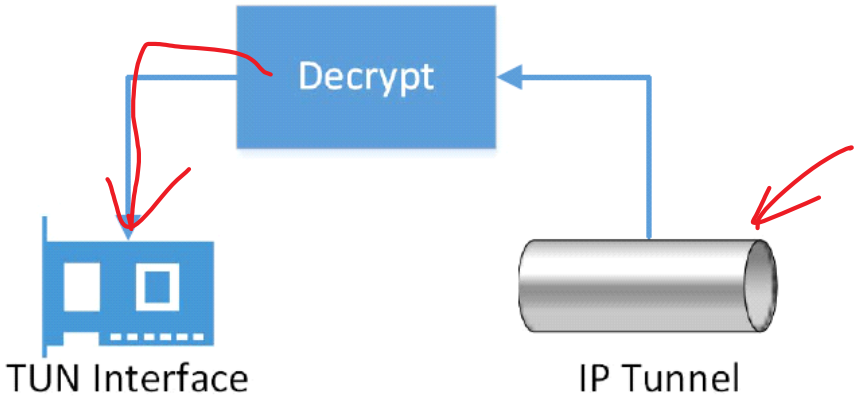


From TUN Interface to Socket (Tunnel)



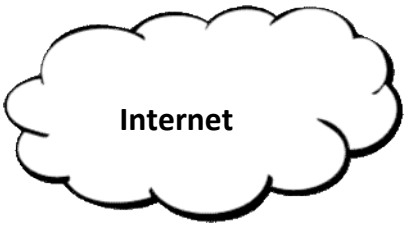

```
void tunSelected(int tunfd, int sockfd){
    int  len;
    char buff[BUFF_SIZE];

    printf("Got a packet from TUN\n");

    bzero(buff, BUFF_SIZE);
    len = read(tunfd, buff, BUFF_SIZE);
    sendto(sockfd, buff, len, 0, (struct sockaddr *) &peerAddr,
                    sizeof(peerAddr));
}
```

From Socket (Tunnel) to TUN Interface

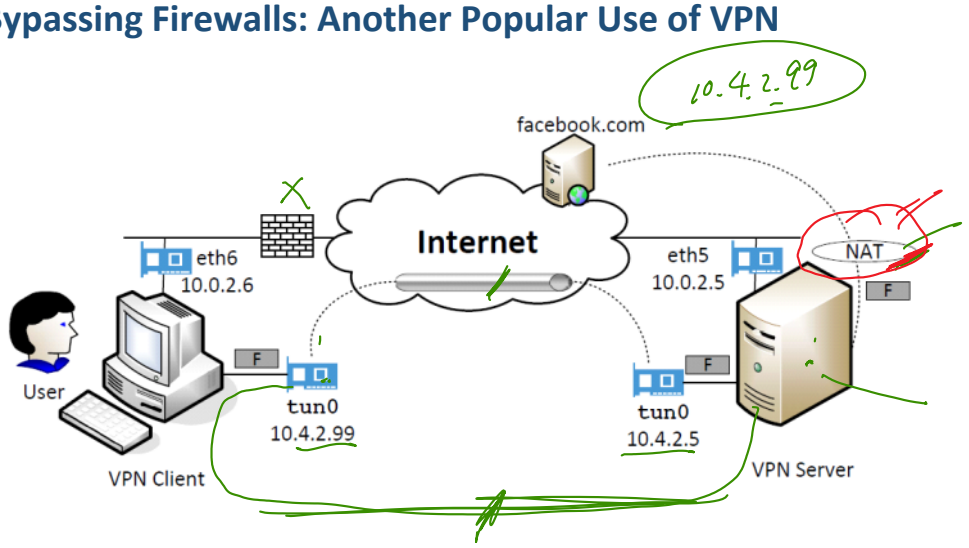



```
void socketSelected (int tunfd, int sockfd){
    int  len;
    char buff[BUFF_SIZE];

    printf("Got a packet from the tunnel\n");

    bzero(buff, BUFF_SIZE);
    len = recvfrom(sockfd, buff, BUFF_SIZE, 0, NULL, NULL);
    write(tunfd, buff, len);
}
```

Bypassing Firewalls: Another Popular Use of VPN



Bypassing Firewalls: Another Popular Use of VPN

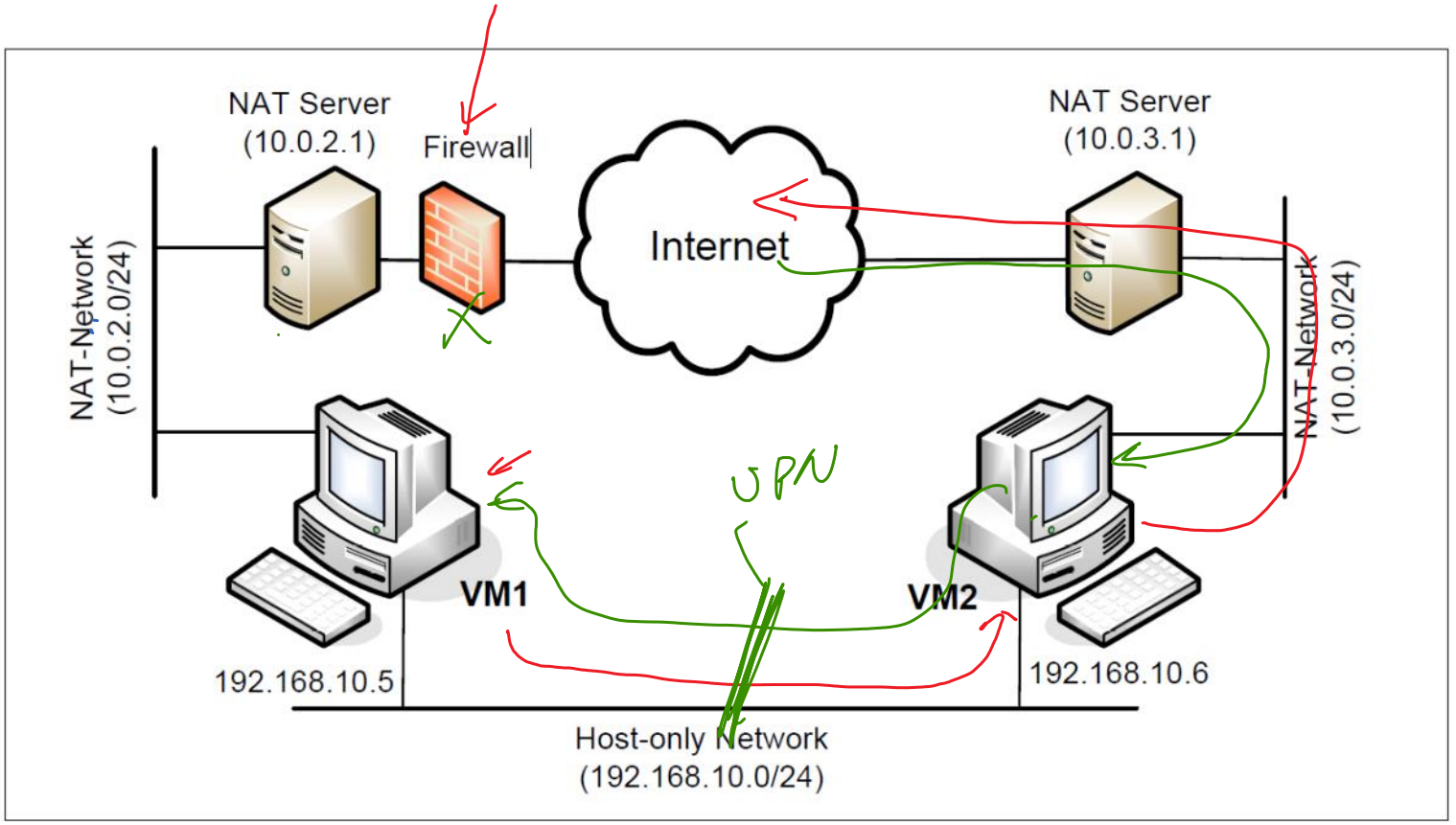


Question: Bypassing Firewall

Assume that your company's firewall blocks access to Facebook from inside the company network. But you are a SU alumni, and you still have access to SU's sura VPN. Please describe how you can use sura to bypass your company's firewall, so you can still get access to Facebook.

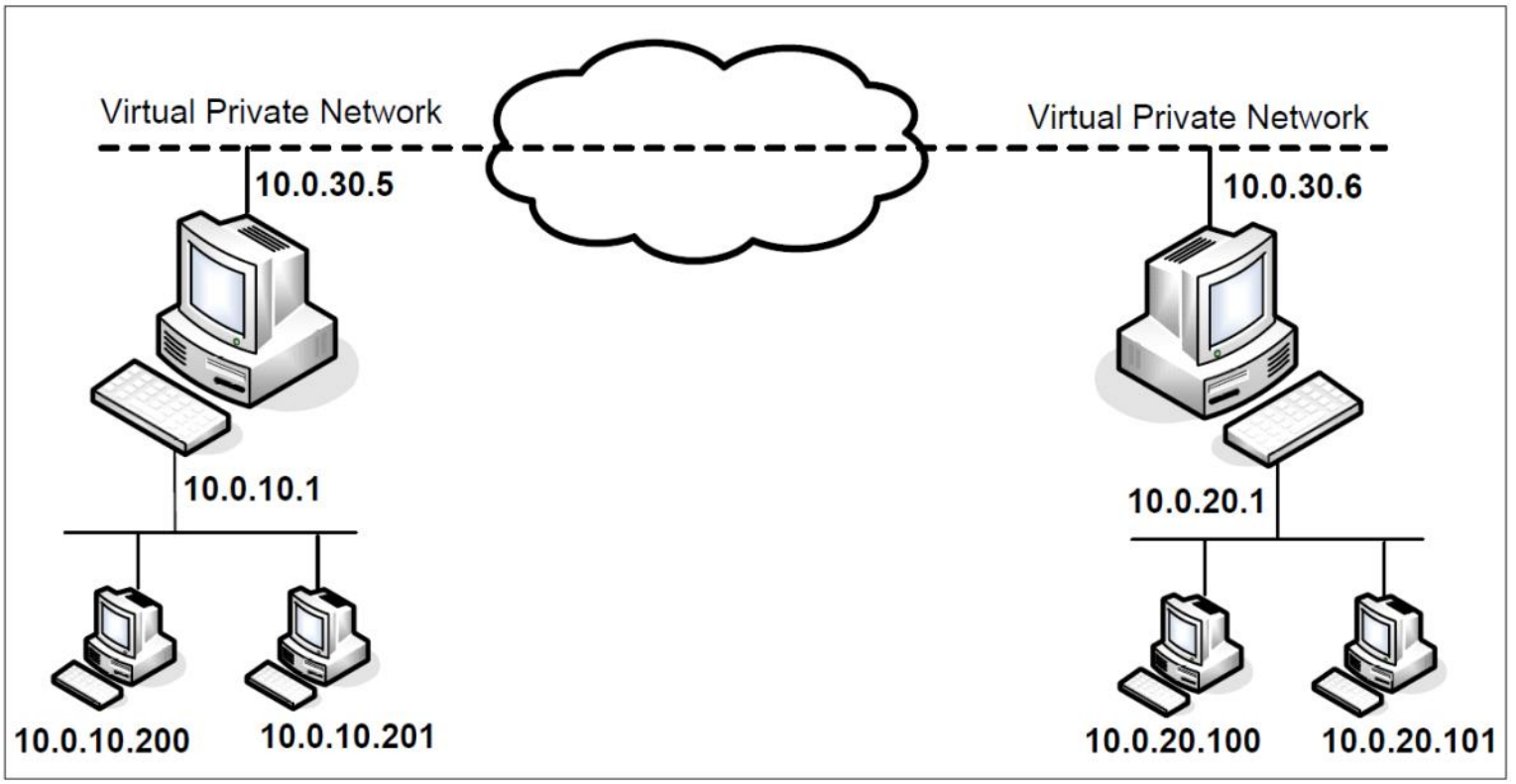
```
IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      192.168.0.1     192.168.0.13     25
        127.0.0.0        255.0.0.0         On-link         127.0.0.1    306
        127.0.0.1  255.255.255.255         On-link         127.0.0.1    306
  127.255.255.255  255.255.255.255         On-link         127.0.0.1    306
      128.230.0.0      255.255.0.0   128.230.153.30   128.230.153.65     21
   128.230.153.11  255.255.255.255      192.168.0.1     192.168.0.13     26
   128.230.153.65  255.255.255.255         On-link    128.230.153.65    276
      192.168.0.0    255.255.255.0         On-link      192.168.0.13    281
     192.168.0.13  255.255.255.255         On-link      192.168.0.13    281
    192.168.0.255  255.255.255.255         On-link      192.168.0.13    281
     192.168.56.0    255.255.255.0         On-link      192.168.56.1    266
     192.168.56.1  255.255.255.255         On-link      192.168.56.1    266
   192.168.56.255  255.255.255.255         On-link      192.168.56.1    266
     192.168.60.0    255.255.255.0         On-link      192.168.60.1    266
     192.168.60.1  255.255.255.255         On-link      192.168.60.1    266
   192.168.60.255  255.255.255.255         On-link      192.168.60.1    266
    192.168.200.0    255.255.255.0         On-link     192.168.200.1    266
    192.168.200.1  255.255.255.255         On-link     192.168.200.1    266
  192.168.200.255  255.255.255.255         On-link     192.168.200.1    266
```

Lab Setup




```
$ sudo iptables -t mangle -A POSTROUTING -d 128.230.210.0/24 -o eth12 -j DROP
```

VPN Applicatoin: Private Network



Create a TUN Interface (Virtual Network Interface)

❖ Code.

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

ifr.ifr_flags = IFF_TUN | IFF_NO_PI;

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

❖ Compile and run the code.

```
seed@ubuntu(10.0.2.18):~/vpn/TunDemo$ gcc -o tundemo tundemo.c
seed@ubuntu(10.0.2.18):~/vpn/TunDemo$ sudo ./tundemo
TUN file descriptor: 3
[07/01/16 15:57] root@ubuntu:.../TunDemo#
```

❖ Check the interface.

```
seed@ubuntu(10.0.2.18):~/vpn$ ifconfig tun0
tun0      Link encap:UNSPEC  HWaddr 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)
```

❖ Assign an IP address to the tun0 interface.

```
seed@ubuntu(10.0.2.18):~/vpn$ sudo ifconfig tun0 10.0.4.99/24 up
seed@ubuntu(10.0.2.18):~/vpn$ 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:10.0.4.99  P-t-P:10.0.4.99  Mask:255.255.255.0
          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)
```

❖ Check the route for the 10.0.4.0/24 network (the route is automatically added).

```
seed@ubuntu(10.0.2.18):~/vpn$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         10.0.2.1        0.0.0.0         UG    0      0        0 eth18
10.0.2.0        *               255.255.255.0   U     1      0        0 eth18
10.0.4.0        *               255.255.255.0   U     0      0        0 tun0
link-local      *               255.255.0.0     U     1000   0        0 eth18
192.168.56.0    *               255.255.255.0   U     1      0        0 eth16
```

If the route is not there, use the following command to add it:

```
$ sudo route add -net 10.0.4.0/24 tun0
```