**VPN Project**

**Part 1: Setting up the SEED virtual machines and the network:**

Step 1: Go to link <https://seedsecuritylabs.org/lab_env.html> , then go to the “SEED Ubuntu16.04 VM (32-bit)” section and click on the zip file next to DigitalOcean to download the file (this will take some time to download). Once you are done downloading the zip file, extract it to a location you can remember.

Step 2: Open VirtualBox Manager, and click New:

Graphical user interface, text, application

Description automatically generated

Step 3: Choose a name such as VPN Server, but choose for Type “Linux” and Version “Ubuntu (32-bit)”. Then click Next:

Graphical user interface, application

Description automatically generated

Step 4: Click Next:

Graphical user interface, text, application

Description automatically generated

Step 5: Choose for your virtual hard disk file the second option shown below from your extracted folder, when you click the folder icon in the “Use an existing virtual hard disk file” option in Step 6:

Graphical user interface, application

Description automatically generated

Step 6: Click Create:

Graphical user interface, text, application, Word

Description automatically generated

What should appear on your VirtualBox Manager:

Graphical user interface, application

Description automatically generated

Step 7: Next go to File, then Preferences. Then click on the Network tab on the left. Then the green plus icon in the right (if you don’t have a NAT Network):

Graphical user interface, text, application, email

Description automatically generated

Step 8: Click on the new Nat Network shown, then choose a Network Name, then a Network CIDR. Then click OK, then OK again:

Graphical user interface, text, application

Description automatically generated

Step 9: Go to your Settings for the VPN Server:

Graphical user interface, text, application

Description automatically generated

Step 10: Choose the Network tab, then for “Attached to:” choose “NAT Network”, then the name of your new NAT Network (mine is VPNNatNetwork). Then for Promiscuous Mode choose “Allow VMS”. Then Click OK:

Graphical user interface, application

Description automatically generated

Step 11: Right click on your Server VM in VirtualBox Manager and choose Clone. Then choose new name for your cloned VM (below is name VPN Client), then choose “Generate new MAC addresses for all network adapters” for your MAC Address Policy. Then click Next:

Graphical user interface, text, application, email

Description automatically generated

Step 12: Choose “Full clone”, then click Clone:

Graphical user interface, text, application, email

Description automatically generated

What should appear (you should have the same Network settings for your VPN Client as you do with the VPN Server):  
Graphical user interface, application

Description automatically generated

Step 13: Start both your VPN Server and VPN Client virtual machines, and open Terminal for both. Then execute ifconfig -a to find the IP addresses for both machines:

Text

Description automatically generated

VPN Client’s IP address is 11.0.2.4

A screenshot of a cell phone

Description automatically generated with medium confidence

VPN Server’s IP address is 11.0.2.5

Step 14: Ping from one VM to another, and vice versa **(take screenshots for your report)**:

A screenshot of a computer

Description automatically generated with medium confidence

Pinged VPN Server from VPN Client

A screenshot of a computer

Description automatically generated with medium confidence

Pinged VPN Client from VPN Server

Step 15: Make sure that VPN Client VM is able to ping itself, and do the same for the VPN Server VM:

Text

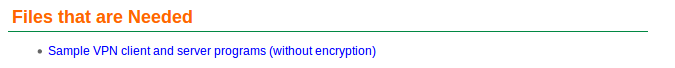
Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

**Part 2: Downloading and compiling vpnserver and vpnclient programs**

Step 16: Visit <https://seedsecuritylabs.org/Labs_16.04/Networking/Firewall_VPN/> on both of your virtual machines using Firefox. Then scroll down the page to click on “Sample VPN client and server programs (without encryption)” on each virtual machine:



Step 17: Next click “Save File”, the OK:

Graphical user interface, text, application

Description automatically generated

You will see the following in Downloads

Graphical user interface, application

Description automatically generated

Step 18: Right click on the file and click “Extract Here”, then you will see the following:

Graphical user interface, chart, funnel chart

Description automatically generated

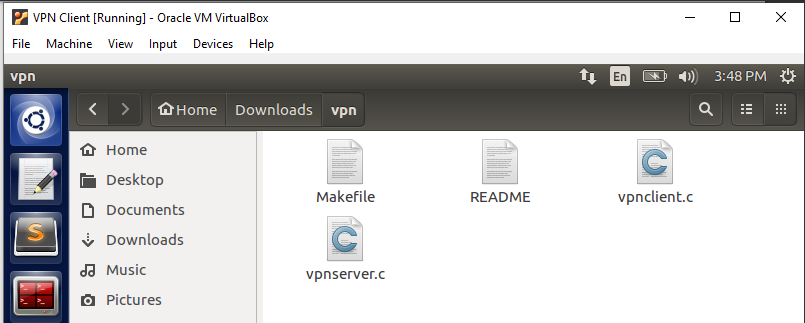
Note: Please make sure that you do Steps 16 – 18 for both virtual machines

Step 19: On you VPN server virtual machine, open the Terminal. Then go to your extracted “vpn” folder (see above), then execute “make”:

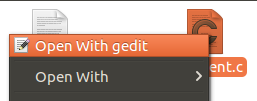
Text

Description automatically generated

Step 20: On your VPN Client virtual machine, open your extracted vpn folder, and right click on vpnclient.c and click on Gedit:



Click “Open with gedit”:



Step 21: Toward the top of the file shown be a line of code that states, #define SERVER\_IP “127.0.0.1”. Replace this IP address with the VPN Server’s IP address

Before:



After:



Step 22: Then click Save in the top right:

Shape, rectangle

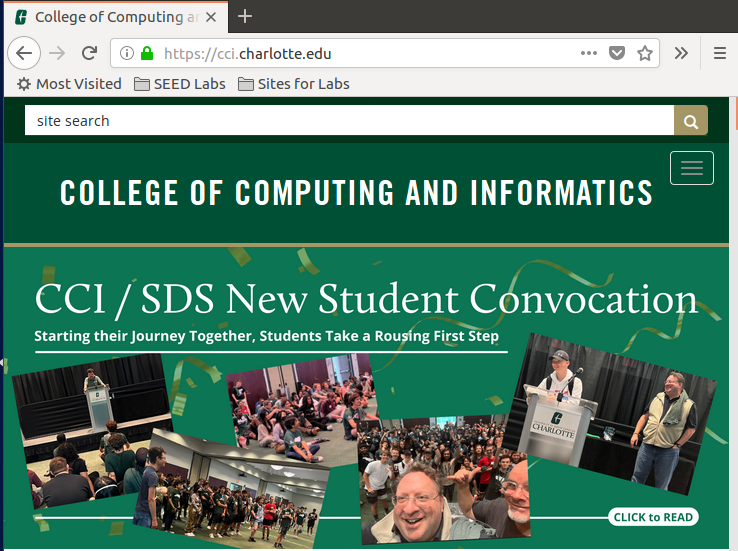
Description automatically generated

Step 23: Open Terminal on your VPN Client virtual machine, then go to the location of your extracted vpn folder and execute “make”:  
Text

Description automatically generated

**Part 3: Setting up the firewall**

Step 24: On your VPN Client virtual machine, visit any webpage using your Firefox browser (this will be the webpage you will be blocking your client from accessing with a firewall). For example, I am accessing UNCC’s CCI page:



Step 25: Find the IP address of that link. For example, I am using whois.domaintools.com where I am able to find the IP address for cci.charlotte.edu (see yellow highlighted below):

Graphical user interface, text, application, email

Description automatically generated

Step 26: Then execute on your VPN Client “sudo ufw enable”:  
Text

Description automatically generated

Step 27: Use your enp0s3 interface (recall Step 13) and the IP address of the webpage you visited (in my case, its 152.15.38.60) to execute the following command:  
Text

Description automatically generated

Step 28: Then execute “sudo ufw status”:

A picture containing text

Description automatically generated

Step 29: When I try to access cci.charlotte.edu again on my VPN Client, I get the following:

Graphical user interface, text, application

Description automatically generated

**Part 4: Setting up VPN tunnel**

Step 30: On your VPN server virtual machine, go to your extracted vpn folder on Terminal, and execute the following:

Text

Description automatically generated

Step 31: In another terminal, execute “ifconfig -a”. You should then see a tun0 interface:

Text

Description automatically generated

Step 32: Next execute “sudo ifconfig tun0 192.168.53.1/24 up”:

Text

Description automatically generated

Step 33: When you execute “ifconfig -a”, you will then see the following for tun0 where tun0 now has IP address of 192.168.53.1:

Text

Description automatically generated

Step 34: Next execute “sudo sysctl net.ipv4.ip\_forward=1” in order for your VPN server to act as a gateway:

Text

Description automatically generated

Step 35: On your VPN Client virtual machine, navigate to your extracted vpn folder and execute “sudo ./vpnclient” using the VPN Server’s IP address (recall Step 13).

Text

Description automatically generated

What you will now see on the VPN server:

Text

Description automatically generated

Step 36: Open another Terminal window in your VPN Client, and execute “sudo ifconfig tun0 192.168.53.5/24 up”:

Text

Description automatically generated

What now appears on VPN Server:

Text

Description automatically generated

**Part 5: Add routing to VPN tunnel**

Step 37: On VPN server virtual machine, execute “sudo route add -net 192.168.53.0/24 tun0” (192.168.53.0 is my VPN network address, but you may have something different):

Text

Description automatically generated

Step 38: On VPN client virtual machine, execute the same command of “sudo route add -net 192.168.53.0/24 tun0”:

Text

Description automatically generated

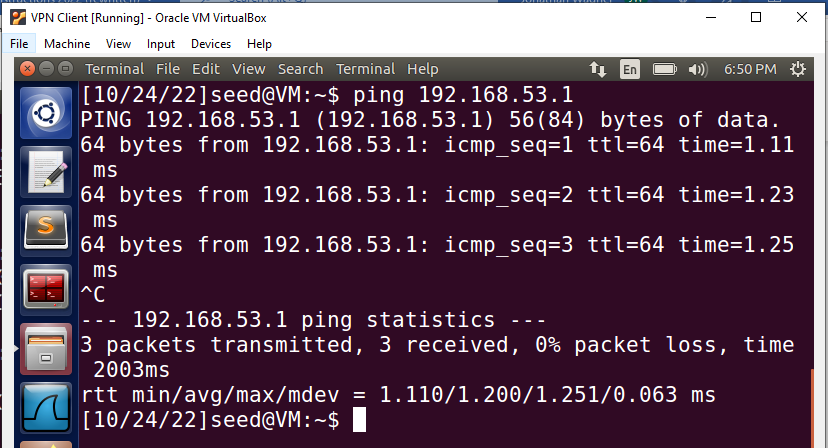
Step 39: Next execute on your VPN client the same command, but this time you are using the IP address of the site you blocked (I have mine listed for range below, so instead of listing the IP address I blocked of 152.15.38.60, I list 152.15.38.0/24. Otherwise, the below command won’t work):

Text

Description automatically generated

Step 40: You should now be able to have the VPN Client and VPN Server ping each other’s tun0 interfaces **(take screenshots for your report)**:

Note: If you are having trouble pinging the VPN Client, try disabling the VPN Client’s firewall with “sudo ufw disable” and try pinging the VPN Client from the VPN Server. Then reenable VPN Client’s firewall with “sudo ufw enable”, then try pinging the VPN Client from the VPN Server.



Pinging VPN Server’s tun0 interface from VPN Client

A screenshot of a computer

Description automatically generated with medium confidence

Pinging VPN Client’s tun0 interface from VPN Server

**Part 6: Enabling NAT on your VPN Server:**

Step 41: Go to your VPN server virtual machine. For cleaning iptables rules, execute the following two commands:

Text

Description automatically generated

Step 42: Then execute “sudo iptables -t nat -A POSTROUTING -j MASQUERADE -o enp0s3” (for this command, use the interface that is linked to your VPN Server’s IP address from Step 13, in my case its “enp0s3”):

Text

Description automatically generated

Step 43: Now try accessing the blocked link on your VPN Client (I was able to reach my blocked link below):

Graphical user interface, website

Description automatically generated

**What you need to turn in:**

1. (40 points) Take screenshots of the following:
   1. The VPN Server pinging the VPN Client’s enp0s3 interface (recall Step 14)
   2. The VPN Client pinging the VPN Server’s enp0s3 interface (recall Step 14)
   3. The VPN Server pinging the VPN Client’s tun0 interface (recall Step 40)
   4. The VPN Client pinging the VPN Server’s tun0 interface (recall Step 40)
2. (40 points) Follow the below steps, then answer the question that follows:
   1. Open Wireshark on both the VPN Server and VPN Client.
   2. On the VPN Server’s Wireshark, listen to the tun0 interface
   3. On the VPN Client’s Wireshark, listen to the enps0s3 interface
   4. Have the VPN Server ping the VPN Client’s tun0 interface’s IP address.
   5. Take screenshots of what you see on Wireshark on both the VPN Server and VPN Client

Based on what you see on Wireshark on both virtual machines, how does VPN tunneling hide an IP packet within another IP packet? Please explain using the screenshots you took.

1. (40 points) Follow the below steps, then answer the question that follows:
   1. Open Wireshark on your VPN Client
   2. Listen to the enps0s3 interface
   3. Visit the blocked webpage
   4. Take screenshot(s) of what you see on Wireshark
   5. Next listen to the tun0 interface
   6. Visit the blocked webpage again
   7. Take screenshot(s) of what you see on Wireshark

How is the VPN Client able to access the webpage that’s blocked by its firewall? Please explain using the screenshots you took.

References:

<https://seedsecuritylabs.org/Labs_16.04/Documents/SEEDVM_VirtualBoxManual.pdf>

<https://seedsecuritylabs.org/Labs_16.04/PDF/Firewall_VPN.pdf>