Neel Khiroya
COMP 3343
100160178
Nov 5, 2023

Virtual Private Network (VPN) Simulator

Introduction

In today's time, the whole world uses the internet. From seeing your bank account to sending messages, almost everything we do includes the internet. However, privacy and security also play a large role on the internet and should not be overlooked. A VPN or Virtual Private Network is an internet tool that ensure privacy and security over the internet by encrypting a connection over the internet. This ensure that the data sent or received is unable to be understood by anyone or anything in between desired connections.

VPN's started off originally being used by businesses to create a secure connection from their home office or network to their work network. This quickly expanded into everyday user's hands allowing them to have a secure tunnel to another network that will then fetch whatever data they need on the internet. While most connections are secure these days, a VPN is still an important factor for online privacy. Using a VPN will let you connect to a website or download data by first listening to what the encrypted data your computer wants. Once it knows where to connect or what to send, the VPN server will then send the request for you to the required destination without any trace of your network attached to the data. This is the secure part since your data is completely encrypted while hopping around the internet until you reach your VPN. Therefore nothing can see the data you sent to the VPN, your IP address. Instead, everything is seeing the VPN's address. Once the VPN has the request you want, it will then encrypt the data and send it back to your computer that originally sent the data.

Problem or Requirement Definition

During this project I will implement a VPN and compare the data sent while connected to the VPN and while not. This report will provide evidence that using a VPN will increase the privacy and security of your network while accessing the internet. By looking at the traffic of my computer while tunneling everything through the VPN, it will prove that all the data sent out is encrypted and is only being sent to my VPN and nowhere else. When looking at the data while not connected with the VPN created, we will be able to see that my computer network is connected to many different servers and addresses, and the data is mostly not encrypted as its being sent.

The biggest problem with not having a VPN is that you may be sharing more data than think. When accessing the internet, you hop from server to server and you are openly giving away your IP. The websites you visit and are allowing your data to be intercepted. This might not be a problem on your home network, but it may be when using public networks such as restaurant Wi-Fi or your school's network as this will let your data be readable by the admin or anyone who may be a spoof admin. The reason your data is readable is because some of your data is not encrypted between you and the final destination. However, a VPN can prevent this.

Literature Review

These days, most connections over the internet are somewhat encrypted. For example, when you log into a social media or email account that login data is encrypted, Google also will encrypt your google searches for user protection, but this doesn't prevent your IP from being hidden or from hiding the websites you visited. My most common internet tool for data encryption is the SSH protocol and SFTP. These protocols allow a computer to connect to another computer securely. If you think of a mesh or ring topology where all the computers are connected and relay data off each other. If you wish to connect to computer C from computer A but computer B is between you, the data you send will not be accessible by computer B due to encryption.

Based on what you require, there are many alternatives to VPNs. For basic data encryption from peer-to-peer networks, you can use SFTP, or a software called Sprend. Sprend is a file transfer application that will allow you to send large files over the internet. Before the files are sent, they are encrypted and then decrypted once received at the designated computer. This application will not let you connect to websites but will securely transfer your data across the internet.

Solution or Design Proposed

My proposed solution would be to implement your own VPN server and client so that you can always make sure your data will be encrypted and your IP will be hidden outside of the tunnel to the VPN server. By always having your data and IP hidden, you can ensure you are more secure on networks and the internet.

The main components in the solution's design include a designated computer that will act as our VPN server. This allows clients to connect to and create a secure network tunnel to transfer encrypted data through. With our data being tunneled to our VPN, we will not hop around the internet with our IP but rather with the VPNs. This means that we will first ping our VPN server, then from the VPN server it will "bounce" around the internet until it reaches its destination. I will test this using 'tracert', a built-in windows tool that will trace a ping and show the addresses that you hop off. You can get free VPNS that are already configured on another server that only require a client such as NordVPN, but I wanted to see the traffic on both the client side and the server side. I also wanted to implement the VPN myself in case I needed to adjust any advance settings while running tests. To implement a software that encrypts data and acts as a VPN, I decided to use 'OpenVPN'. This software will be installed on the server to create the VPN tunnel and allows client-side connections. I decided to go with this software as it seemed to be the most popular and most reliable free server-side VPN software. Another option I was considering was Algo VPN, however, this software is still in development and requires more configuration to operate. To test the VPN and make sure it works I will be using a software called 'WireShark' that will show the traffic on my local network. WireShark is a free software that will show all incoming and outgoing traffic on your computer. By comparing the IP addresses and the data while using and not using the VPN, we will be able to see if the VPN is working properly. I've decided to go with WireShark as I used it before and find it very simple. This application also allows you to add filters such as an IP filter to only see incoming and

outgoing data for that IP address, which will help me filter the testing data. The other packet sniffer I was considering using is called “TCPDump” but this program doesn’t have a GUI. I wanted to use a GUI to see the data better for my screenshots and for others to easily read the data as well. A GUI also tends to be more visually simple.

Implementation

The biggest problem I faced when Implementing a VPN is allowing it to be connected to public networks. This involves having a server or computer that’s capable of receiving data over the internet. I also need data that we can send to another address over the internet so that we can compare the data sent with and without a VPN and to view the data that’s being sent. Once the data is tested, I want to clearly see the unencrypted data versus the encrypted data. This became another problem as most of the services we use to communicate and send data are already encrypted. I tried uploading data to my personal SHH SFTP server, and performing a google search but there was not much that I could compare as most of the information was encrypted. Although this is a good thing in practical use, it made it difficult for testing. Also websites are still not encrypted so this doesn’t stop networks from seeing your IP since you are not tunneling data through the VPN but rather sending it to the website’s server.

To solve these problems, a shared CPU server with the capability to be connected to the internet was purchased. I also installed and created a small FTP server so that we can send data over the internet to the sever without it being naturally encrypted. The software ‘OpenVPN’ was installed on the shared CPU server to implement the VPN, and this also came with a client-side application to help connect to the VPN.

Once ‘OpenVPN’ was install on my server and I downloaded the client on my local computer, I ran a few tests on my network to see how data was being transferred. Using ‘WireShark’ to see the traffic, there was a handful of different IP addresses that my computer was talking to (see Figure 1). Some of these might be for the weather or time etc. but it vwas receiving data from my IP address. To test that we are hopping around the internet and creating a source to compare to when we check that our tunnel is working, I pinged Google.com and my website (Khiroya.ca) with ‘tracert’. The results showed that we pinged multiple addresses before reaching our destination. These pings include information that you might want to keep private and give proof that we are not already tunneled to a specific address as we always ping my local network first and then different addresses (Figure 2). Using Wireshark, I was able to look at the data it was sending. However, the data was encrypted so it didn’t make much sense. To test that WireShark could view data that wasn’t encrypted, I decided to send the Bee Movie script as a text file to my personal FTP server that doesn’t use an encrypted protocol. After filtering the IP addresses, I was able to easily view the data and see the data that was sent (see Figure 3). This is a positive result as I now know that WireShark is reading the data correctly on my network.

No.	Time	Source	Destination	Protocol	Length	Info
387	26.139131	35.212.120.122	192.168.2.15	RTCP	94	Receiver Report
388	26.146666	192.168.2.15	52.111.230.3	TLSv1.2	82	Application Data
389	26.229993	192.168.2.15	52.207.108.103	TLSv1.2	86	Application Data
390	26.231972	52.111.230.3	192.168.2.15	TCP	60	443 → 54100 [ACK] Seq=1 A
391	26.261343	52.207.108.103	192.168.2.15	TLSv1.2	82	Application Data
392	26.316139	192.168.2.15	52.207.108.103	TCP	54	53790 → 8885 [ACK] Seq=33
393	26.976668	172.65.244.155	192.168.2.15	TCP	60	[TCP Keep-Alive] 5223 → 6
394	26.976683	192.168.2.15	172.65.244.155	TCP	54	[TCP Keep-Alive ACK] 6227
395	27.130468	35.212.120.122	192.168.2.15	RTCP	94	Receiver Report
396	27.157282	162.247.241.2	192.168.2.15	TCP	60	[TCP Keep-Alive] 443 → 62
397	27.157295	192.168.2.15	162.247.241.2	TCP	54	[TCP Keep-Alive ACK] 6219
398	27.505019	162.247.241.2	192.168.2.15	TCP	60	[TCP Keep-Alive] 443 → 62
399	27.505035	192.168.2.15	162.247.241.2	TCP	54	[TCP Keep-Alive ACK] 6220
400	27.539410	162.247.241.2	192.168.2.15	TCP	60	[TCP Keep-Alive] 443 → 62
401	27.539435	192.168.2.15	162.247.241.2	TCP	54	[TCP Keep-Alive ACK] 6220
402	27.596121	Sagemcom_83:4f:64	Spanning-tree-(for-…	STP	60	Conf. Root = 32768/0/0c:a
403	28.111875	Sagemcom_83:4f:64	Spanning-tree-(for-…	STP	60	RST. Root = 32768/0/0c:ac
404	28.138153	35.212.120.122	192.168.2.15	RTCP	94	Receiver Report

Figure 1

```
Tracing route to google.com [142.251.40.142]
over a maximum of 30 hops:

  1     1 ms    <1 ms    <1 ms  mynetwork.home [192.168.2.1]
  2     7 ms     7 ms     6 ms  10.178.218.5
  3     8 ms     6 ms     7 ms  ae15-182.cr02.hlfx.ns.aliant.net [142.166.181.141]
  4     7 ms     7 ms     6 ms  hg-0-2-0-0-50.cr01.hlfx.ns.aliant.net [142.166.149.93]
  5    25 ms    25 ms    25 ms  be19.bx02.nycm.ny.aliant.net [207.231.227.62]
  6    25 ms    24 ms    24 ms  74.125.119.154
  7    25 ms    25 ms    25 ms  142.251.78.63
  8    25 ms    25 ms    25 ms  216.239.49.65
  9    26 ms    26 ms    26 ms  lga25s80-in-f14.1e100.net [142.251.40.142]

Trace complete.
```

Figure 2

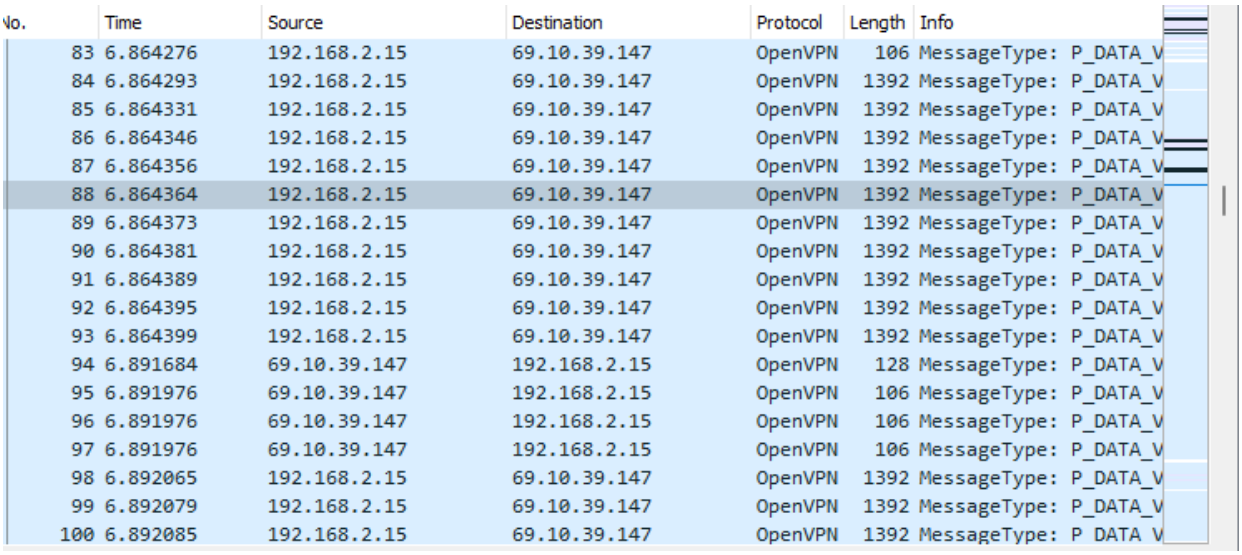
```
I got a feeling we'll be\r\n
working late tonight!\r\n
(The bee honey factories are back up and running)\r\n
(Meanwhile at Vanessa's shop)\r\n
VANESSA:\r\n
(To customer)\r\n
Here's your change. Have a great\r\n
afternoon! Can I help who's next?\r\n
 :\r\n
Would you like some honey with that?\r\n
It is bee-approved. Don't forget these.\r\n
(There is a room in the shop where Barry does legal work for other animals.\r\n
He is currently talking with a Cow)\r\n
COW:\r\n
Milk, cream, cheese, it's all me.\r\n
And I don't see a nickel!\r\n
 :\r\n
Sometimes I just feel\r\n
like a piece of meat!\r\n
BARRY:\r\n
I had no idea.\r\n
VANESSA:\r\n
```

```
1210  20 62 65 65 20 69 73 20  6c 69 76 69 6e 67 20 6d   bee is  living m
1220  79 20 6c 69 66 65 21 21  0d 0a 41 4e 44 59 3a 0d   y life!! ··ANDY:·
1230  0a 4c 65 74 20 69 74 20  67 6f 2c 20 4b 65 6e 6e   ·Let it  go, Kenn
1240  79 2e 0d 0a 4b 45 4e 3a  0d 0a 2d 20 57 68 65 6e   y.··KEN: ··- When
1250  20 77 69 6c 6c 20 74 68  69 73 20 6e 69 67 68 74    will th is night
1260  6d 61 72 65 20 65 6e 64  3f 21 0d 0a 41 4e 44 59   mare end ?!··ANDY
1270  3a 0d 0a 2d 20 4c 65 74  20 69 74 20 61 6c 6c 20   :··- Let  it all
1280  67 6f 2e 0d 0a 42 41 52  52 59 3a 0d 0a 2d 20 42   go.··BAR RY:··- B
```

Figure 3

Knowing that WIreShark is working correctly, and my data was readable, I then turned on the 'OpenVPN' client to check if I had a secure tunnel to my server. Once connected to the VPN, WireShark began to almost communicate to the IP address of my server only (see Figure 3). This is exactly what I was hoping for as it means all my traffic is now being tunneled to my VPN. This data will now be encrypted and my IP address will be no longer be used because the VPN will connect to other servers for me and then tunnel the encrypted data back to my computer.

With the data being tunneled to my VPN I can use WireShark to test that the data is also being encrypted. Using WireShark and the same technique used to view the data without the VPN, I was not able to view the data as it is now encrypted and doesn't make sense anymore. This meaning that the VPN is working correctly and can be used for results.



No.	Time	Source	Destination	Protocol	Length	Info
83	6.864276	192.168.2.15	69.10.39.147	OpenVPN	106	MessageType: P_DATA_V
84	6.864293	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
85	6.864331	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
86	6.864346	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
87	6.864356	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
88	6.864364	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
89	6.864373	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
90	6.864381	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
91	6.864389	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
92	6.864395	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
93	6.864399	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
94	6.891684	69.10.39.147	192.168.2.15	OpenVPN	128	MessageType: P_DATA_V
95	6.891976	69.10.39.147	192.168.2.15	OpenVPN	106	MessageType: P_DATA_V
96	6.891976	69.10.39.147	192.168.2.15	OpenVPN	106	MessageType: P_DATA_V
97	6.891976	69.10.39.147	192.168.2.15	OpenVPN	106	MessageType: P_DATA_V
98	6.892065	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
99	6.892079	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V
100	6.892085	192.168.2.15	69.10.39.147	OpenVPN	1392	MessageType: P_DATA_V

Figure 4

Results and Discussion

The data is now encrypted and secure as all data is being tunnelled to our VPN before hopping around the internet with our information. Using 'tracert', this can be confirmed since we will always ping our VPNs gateway and then our VPN itself. From there, our VPN will send a ping to receive the data we want (Figure 5).

Our IP is hidden from the internet. This is great if you want to visit a site you're unsure is safe or perhaps download something you might not know the download source of. Looking at the undecrypted data (Figure 3), the unencrypted data can clearly be read resulting in this data not being save from network sniffers. However, now that our VPN is in place, our data cannot be read by someone whose intercepting it. It is not readable without the encryption key (Figure 6).

```
C:\Users\neel>tracert google.com

Tracing route to google.com [142.251.40.238]
over a maximum of 30 hops:

  1    25 ms    25 ms    25 ms  172.27.230.1
  2    26 ms    27 ms    27 ms  69.10.39.145
  3    39 ms    43 ms    43 ms  173.225.97.9
  4    26 ms    26 ms    26 ms  vl562.cr1.lga1.us.as19318.net [64.20.32.173]
  5    26 ms    26 ms    26 ms  64.20.32.183
  6    26 ms    25 ms    25 ms  core8-teb3.trouble-free.net [64.20.32.212]
  7    27 ms    26 ms    27 ms  198.32.161.20
  8    26 ms    26 ms    26 ms  108.170.248.97
  9    27 ms    27 ms    26 ms  108.170.236.91
 10    26 ms    26 ms    26 ms  lga34s39-in-f14.1e100.net [142.251.40.238]

Trace complete.

C:\Users\neel>tracert khiroya.ca

Tracing route to khiroya.ca [172.67.174.77]
over a maximum of 30 hops:

  1    25 ms    24 ms    24 ms  172.27.230.1
  2    31 ms    38 ms    31 ms  69.10.39.145
  3    44 ms    40 ms    36 ms  173.225.97.10
  4    82 ms    25 ms    25 ms  switch34.as19318.net [64.20.32.165]
  5    25 ms    25 ms    24 ms  core8-teb3.trouble-free.net [64.20.32.212]
  6    26 ms    27 ms    27 ms  de-cix-new-york.as13335.net [206.82.104.31]
  7    33 ms    26 ms    26 ms  162.158.152.3
  8    27 ms    26 ms    26 ms  172.67.174.77

Trace complete.
```

Figure 5- VPN IP is 69.10.39.145

Figure 6

Its important to keep in mind that you are not directly connecting to the location or website that you clicked on but rather you are hopping around the internet from server to server until you reach your location or website that you are requesting. When connecting to google.com, I visited three different Bell hubs that redirected my request as well. Multiple other servers also redirected my request(Figure 2). While pinging these destinations you leave traces of your information. However, if you are connected to a VPN, due to you tunneling all your data through it instead; your first ping will be the VPN itself, from there the VPN will bounce around (figure 5) hiding your trace on the internet.

Seeing that my data is encrypted and tunneled, I can conclude that my VPN is working properly. I have received the results I expected. The results show that without a VPN it's possible to intercept and read data that is being uploaded as well as seeing the source and IP destination. This is a high security risk in that if not encrypted, your data is being sent over the internet along with your IP, username, and password. Although there are many solutions for this problem as most of the data transfer protocols, we use today are encrypted naturally without the use of a VPN. Although this hides much of the data being sent, you still leave your IP unhidden as well as the website you visit. However, our tests show that this will not be the case with a VPN. There are many VPN clients who do not require you to create your own VPN server and will still give the same results that I have concluded. For example, Proton VPN is a free VPN that will also create a tunnel for your data, encrypt it and hide your information from the internet. However this doesn't hide your information from Proton VPN themselves. There are many other free VPNs available for easy use that will also give the same results as my private server VPN achieved.

Conclusion

With our test results being presented and with the evidence that my VPN is tunning data and encrypting it, I can conclude that my VPN is working correctly and I have received the test results that I was expecting. The evidence shows that my connection is tunnelled directly to my VPN IP address. From there, it will ping the server needed to reach the designation and then will tunnel the data back to my local computer IP address without exposing my information on the internet. The VPN is proven to encrypt data since sending unencrypted data through my VPN will still be encrypted. However, if I'm not connected to the VPN, it's possible to intercept and read the data being transferred since it is unencrypted (Figure 2). The use of a VPN will increase the security of the topology that requires the connection to other user computers to access the internet. Since you are hoping from these computers, it's possible for someone to read any data that is unencrypted. It has also been noted that hackers may spoof internet networks. If you were to connect to that network thinking it's a regular connection, but actually it is tunnelling your data though the hacker's computer, to the router. This means that someone will see the data you are requesting from the server. Although your request and personal data will most likely be encrypted, hackers will still be able to view the websites and redirect websites you wish to visit. This can also be fixed with a VPN as the data will be encrypted itself again such that it's not readable by someone who's intercepting it.

For future research, it would be interesting to look at a connection that is both connecting to the same VPN, meaning that in theory; the data should be completely encrypted the whole time other the internet.

References

1. Spoof host / hackers creating a fake network.
Radhika Vyas , *CouldRadius,* accessed Nov 18 2023, <https://www.cloudradius.com/wi-fi-spoofing-a-major-threat-to-network-security/#:~:text=The%20hacker%20creates%20a%20rogue,activity%20and%20obtain%20confidential%20information>

2. History of VPNS.
Alexander S. Gillis 2021, TechTarget, accessed Nov 17 2023, <https://www.techtarget.com/searchnetworking/definition/virtual-private-network#:~:text=VPN%20technology%20was%20first%20used,with%20and%20used%20by%20businesses>

3. What's a VPN?
NordVPN, accessed Nov 16 2023, <https://nordvpn.com/what-is-a-vpn/>

4. Server hosting that I used for my VPN.
<https://www.interserver.net/>

5. Wireshark network tool.
<https://www.wireshark.org/>

6. Sprend
<https://sprend.com/large-file-transfer-encrypted-online-free#:~:text=Sprend%20is%20one%20of%20the,most%20secure%20encryption%20technologies%20available.&text=Sprend%20is%20available%20in%20two,The%20free%20version>

7. OpenVPN
<https://openvpn.net/>

8. WANS
<https://www.paloaltonetworks.com/cyberpedia/what-is-sd-wan#:~:text=SD%2DWAN%20(software%2Ddefined,wide%20area%20networks%20(WANs) >