Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

VPN SpyGlass: Encrypted Network Traffic Analysis of VPNs Application

First A. Author1, Fellow, IEEE, Second B. Author2, and Third C. Author, Jr.3, Member, IEEE

1National Institute of Standards and Technology, Boulder, CO 80305 USA

2Department of Physics, Colorado State University, Fort Collins, CO 80523 USA

3Electrical Engineering Department, University of Colorado, Boulder, CO 80309 USA

Corresponding author: First A. Author (e-mail: author@ boulder.nist.gov).

This paragraph of the first footnote will contain support information, including sponsor and financial support acknowledgment. For example, “This work was supported in part by the U.S. Depart­ment of Com­merce under Grant BS123456.”

ABSTRACT In the era of the internet, virtual private networks, or VPNs, have become essential tools for protecting online activity as digital privacy and security grow more important. VPNs provide customers with confidentiality and integrity protection against cyber-attacks and possible monitoring by encrypting data flow. But there are drawbacks to using encryption, especially when it comes to digital forensics. Although it strengthens security and privacy, it also makes it harder to track down hackers and makes it more difficult to look into illegal activity online. In order to close this gap, this study develops a forensic analysis framework that can recognize and extract potentially relevant forensic artefacts from encrypted VPN data. Through the examination of many top VPN providers and the use of cutting-edge network forensic methods, this study seeks to identify unique trends and irregularities that indicate malevolent activity in encrypted data streams. The technique presented herein provides law enforcement and cybersecurity experts with a unique strategy to identify, investigate, and attribute digital crimes in an encrypted environment by navigating the encrypted landscapes of VPN services. By using this analytical lens, the study respects the essential requirement for privacy in digital communications while simultaneously improving the toolbox accessible for digital forensics investigations. As such, our study makes a substantial contribution to the current discussion about striking a careful balance between protecting digital privacy and security and maintaining the rule of law in the online sphere.

INDEX TERMS Cybersecurity, Digital forensics, Encrypted traffic analysis, Internet privacy, Network forensics, Virtual Private Network (VPN), Deep Packet Inspection (DPI), PfSense Firewall, Programming Protocol-independent Packet Processors (P4) Language

1. Introduction

With internet access permeating every aspect of life in the modern digital world, privacy and security issues have assumed a central role. Virtual Private Networks (VPNs) are becoming increasingly popular as vital tools for safeguarding online communications as concerns about data breaches, spying, and cyberattacks escalate. VPNs provide a strong barrier against illegal surveillance by masking users' IP addresses and encrypting data transfers, guaranteeing the privacy of personal and company information. In addition to protecting data from possible interceptors, this encryption gives users the ability to browse the internet anonymously, something that was not possible before the advent of digital technology. VPN technology is being used by a diverse range of users, including organizations looking to secure remote access to their networks and individuals want to safeguard their privacy and get around geo-restrictions. A secure tunnel that may be created for data flow across the internet by VPNs effectively prevents other parties, such as internet service providers and potential eavesdroppers, from accessing encrypted data. This is why VPNs are so popular. Because of this, VPNs are now an essential part of any modern digital security and privacy toolset.

But in the field of digital forensics, the encryption that forms the basis of VPN technology poses a conundrum. Although it greatly benefits user privacy and data security, it also presents serious difficulties for the investigation of digital crimes. The actions of users, especially bad actors, can be successfully hidden by encrypted VPN traffic, making it more difficult for forensic investigators to track down cybercriminals and get evidence that can be used in court. The consequences for cybersecurity and law enforcement experts are significant since the very instruments used to safeguard privacy online can equally work as a mask for illegal activity.

There is more evidence than just anecdotal evidence supporting the surge in VPN usage; several research studies and market analysis have confirmed this trend. Several causes, such as increased awareness of digital privacy problems, business cybersecurity policies, and the growing limitations on internet freedom in different countries, are contributing to this spike. VPNs' adaptability, meeting a variety of requirements from data security to gaining access to information that is forbidden, has solidified their place as an essential part of the contemporary digital infrastructure.

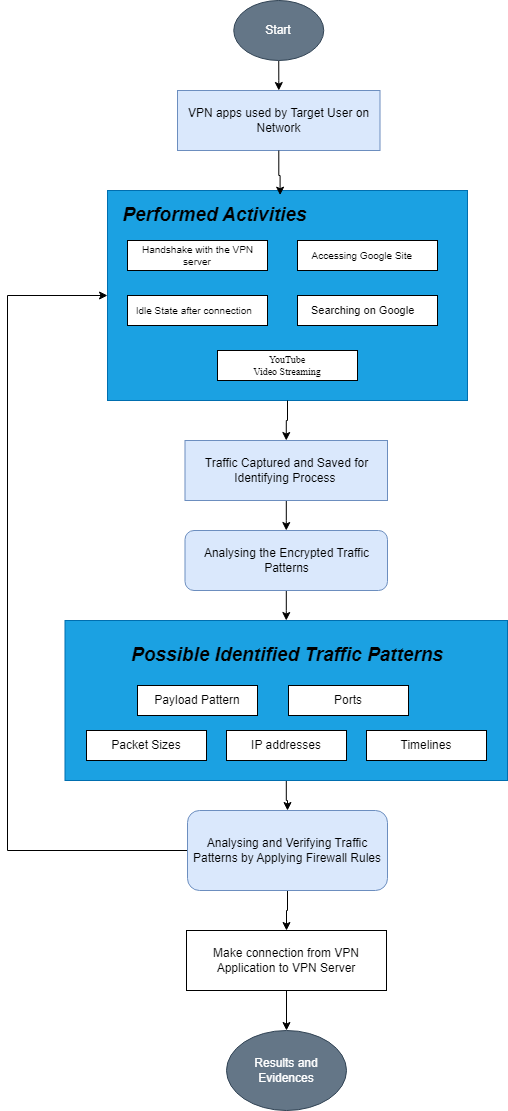


Figure 1: **The flow diagram of proposed strategy.**

Recognizing the need to maintain the rule of law in cyberspace while protecting privacy via VPN use, this research attempts to negotiate the tricky landscape of encrypted VPN traffic from a forensic standpoint. Our research aims to provide a more sophisticated knowledge of how VPN services might be examined to identify and look into the activities of cybercriminals without jeopardizing the privacy safeguards that are the cornerstone of VPN technology. Through an analysis of encrypted traffic produced by many top VPN providers, this research seeks to identify trends, artefacts, and abnormalities that may be signs of nefarious behavior. By carefully dissecting these encrypted data streams using cutting-edge network forensic methods, we want to pave the way for law enforcement and cybersecurity professionals to successfully seek responsibility in the digital sphere. This equilibrium aims to guarantee that the internet stays a safe, confidential, and authorized area for every user.

1. Contributions

Our motivation stems from the realization that there are two competing imperatives at work here:

* Safeguarding individual privacy in the digital age
* Preventing the internet from turning into a lawless black market where bad actors may operate with impunity.

By doing this study, we add to the body of knowledge on digital privacy, security, and forensic science. We do this by putting forth a framework that upholds privacy principles and strengthens the tools available to forensic investigators to fight digital crime in an encrypted world.

1. Organization

The methodology, experimental setup, findings, and analysis of our forensic examination of VPN traffic are presented in this study. We also go over the suggested strategy's possible applications, advantages, drawbacks, and future prospects. Our goal is to improve security and accountability in the digital sphere by furthering the development of forensic techniques in the context of encrypted network communications.

1. Literature Review
2. Experimental Setup

To capture network traffic, Network architecture was created as shown in Figure 2. To show the continuous traffic of VPNs moving across the network, a firewall, and PCs are part of the suggested experimental configuration. The PC under investigation is connected to the internet through the firewall and all the internet traffic is routed through a PC-based PfSense firewall. The PC-based PfSense firewall filtered all internet traffic. Sniffing, capturing, and monitoring all network traffic was the manual job. To facilitate investigation, the trace files produced were stored. The configured firewall was used to block VPN traffic in a controlled environment as Figure 3.

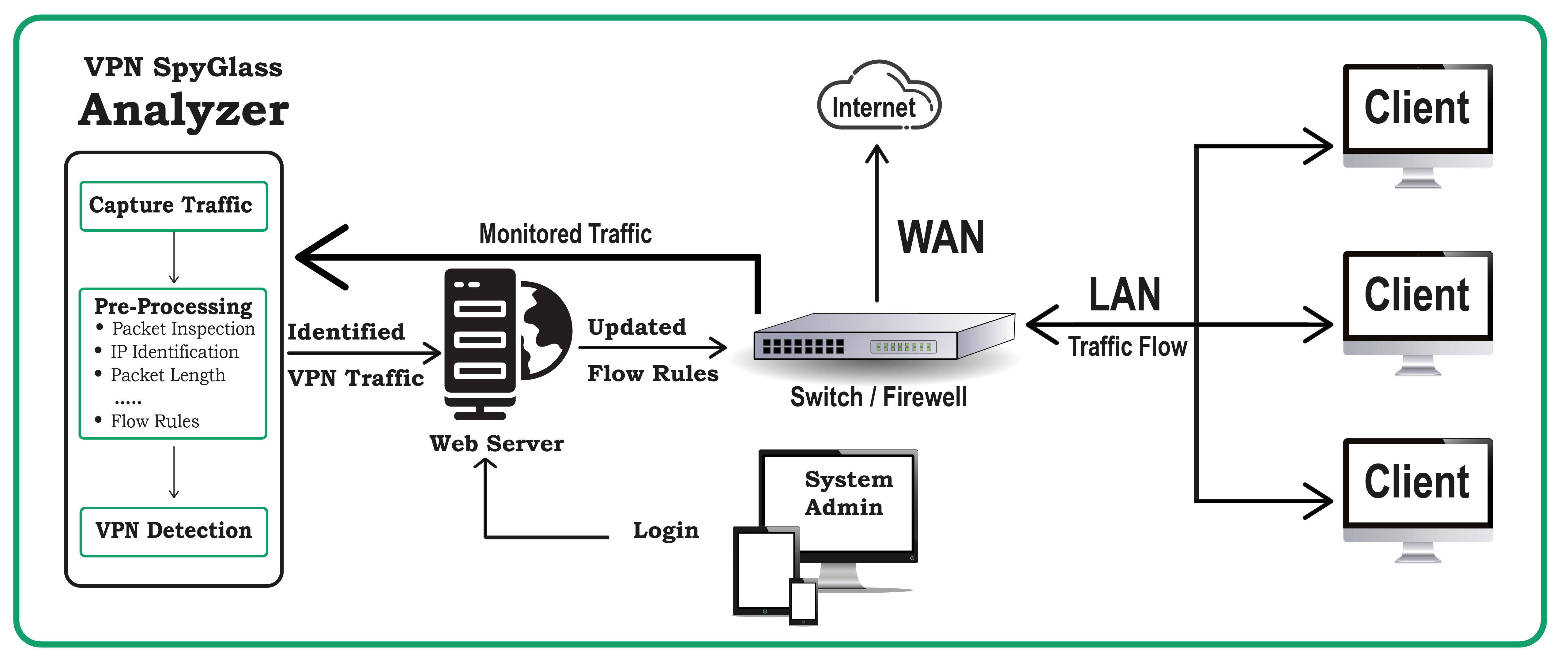


Figure 2: Experimental setup for capturing the network encrypted traffic of VPNs.

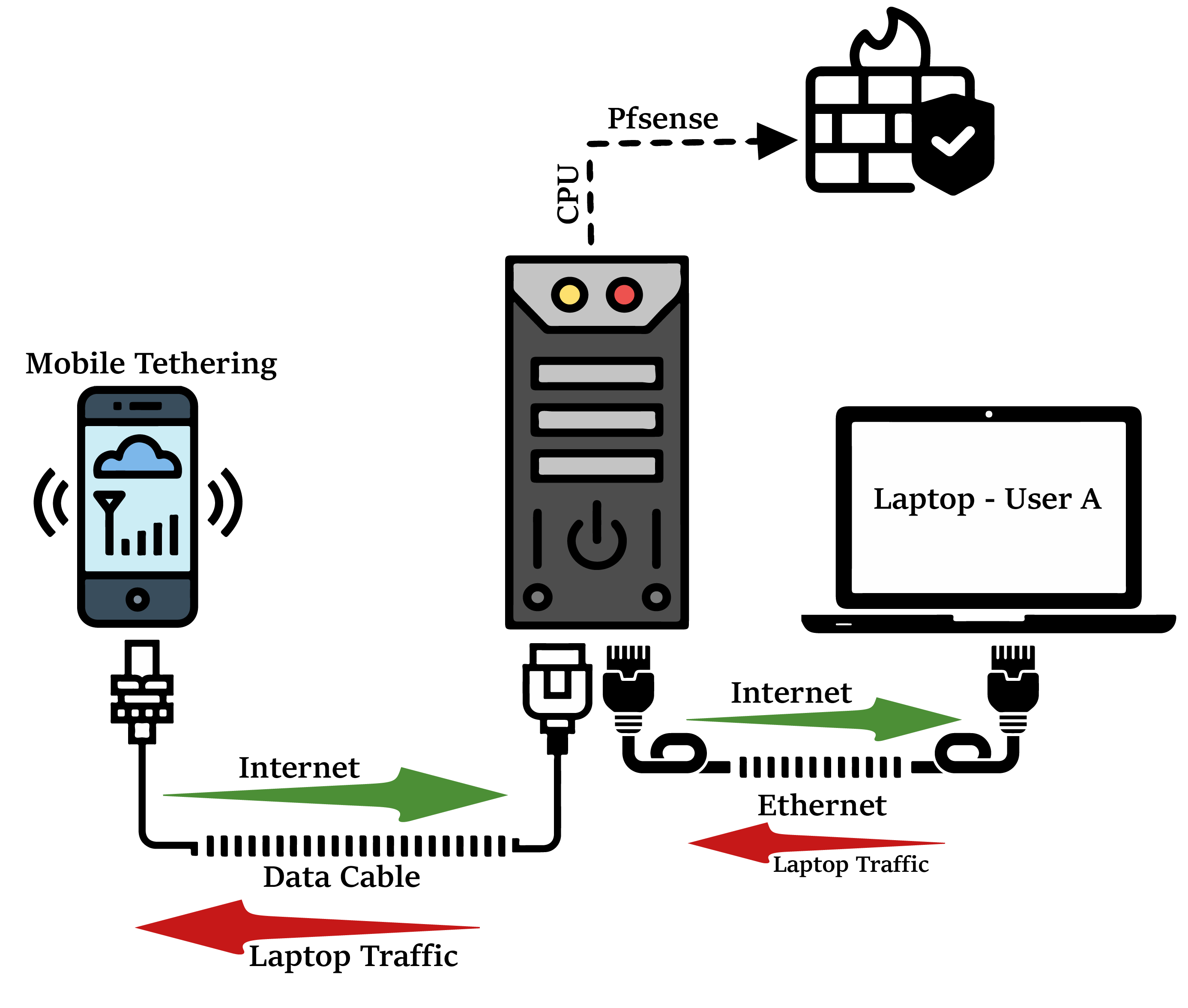
In order to monitor the encrypted traffic through the examination of trace files, Wireshark Software was installed for network traffic analysis. Notably, for privacy and secrecy purposes, the payload was encrypted, but the IP addresses and ports were left in cleartext. The IP addresses and ports made it possible to ascertain the actions taken by VPN applications and how they behaved during the connection. For analytical reasons, we gathered a significant number of trace files during the tests; some of them are accessible on for a clearer understanding. The details of the devices and software tools used in the tests are displayed in Table 1.

**Table 1:**

**DEVICES AND TOOLS SPECIFICATIONS FOR EXPERIMENTAL SETUP**

|  |  |  |  |
| --- | --- | --- | --- |
| **Device/Tool** | **Purpose** | **Company** | **Software/OS** |
| Desktop Computer | To deploy firewall. | Dell | Windows 10, 64- bit Operating system that has an x64-based processor, 4GB RAM, Intel (R) Core (TM) i5-3470 CPU @3.20 GHz. |
| Firewall | Capturing and monitoring packets in the traffic.  Firewall rules | PfSense | pfSense-CE-memstick-2.7.1-RELEASE-amd64 |
| Wireshark | To read the network trace. | Wireshark | Version 4.2.2 |
| Ethernet Wire | Used as bridge. | PTCL |  |
| Laptop | To run VPN applications and perform major activities. | HP | Windows 11, 64-bit operating system, x64-based processor, 16GB RAM, Intel(R) Core (TM) i7-8565U CPU @ 1.80GHz 1.99 GHz |
| Proton VPN | Application under investigation | Proton | Version 2.4.3 |
| Windscribe | Application under investigation | Windscribe Limited | Version 2.8.6 |
| Hide.me | Application under investigation | eVenture Ltd | Version 3.15.1 |
| Tunnel Bear | Application under investigation | McAfee | Version 5.0.0 |
| TurboVPN | Application under investigation | INNOVATIVE CONNECTING PTE. LIMITED | Version 4.0.5.2 |
| Hotspot Shield | Application under investigation | AnchorFree | Version 11.2.1 |
| Cloudflare WARP | Application under investigation | CloudFlare | Version 11.6.0 |

To examine the behavior of the VPN application, such as the characteristics of the packets and how client and server communication functions, the suggested method sniffed the encrypted traffic between the client and its servers, displaying a list of several servers along with their IP addresses and ports. Imposing new network rules is made easier with the identification of ports and IP addresses. For instance, we found that several app servers offer distinct response regions and protocols during our studies. The part on results and analysis goes into depth on user activity.



**Figure 3**: Experimental setup for blocking VPN connections.

1. Justification of Firewall

By putting a firewall on the investigation network, the control may effectively detect the VPN app's behavior. Installing a firewall in the investigation network enables effective detection and control of VPN application behavior. Firewall rules are useful in validating the default VPN program behavior and revealing any hidden or alternate behaviors. The firewall helps forensic examiners comprehensively analyze VPN-related activity by imposing limits and monitoring traffic patterns.

Furthermore, the firewall allows for the monitoring of client-server communication design patterns, such as TCP/UDP ports and server ranges used for VPN connections. This knowledge is extremely useful for understanding how VPN programs create connections and communicate with servers. The firewall's configuration flexibility enables specialized management of network traffic depending on unique requirements for both Wide Area Network (WAN) and Local Area Network (LAN). By default, inbound and outgoing traffic are prohibited, however rules can be established to enable or limit access to internet resources as needed.

The firewall interface's packet capture capability allows for real-time collection of VPN traffic from targeted devices. This direct capture approach lowers dependency on other traffic capture technologies while improving the accuracy of traffic analysis. Captured VPN network traffic is captured as trace files from both the LAN and WAN interfaces, which may subsequently be analyzed using tools such as Wireshark. This analysis provides valuable insights into VPN application behavior, aiding in the identification of patterns, anomalies, and potential security threats.

**Table 2:**

**VPNs AND THEIR USED**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **VPN** | **WireGuard** | **OpenVPN (UDP)** | **OpenVPN (TCP)** | **IKEv2** | **Stealth** | **WStunnel** | **SoftEther** | **SSTP** | **Hydra** | **TCP** | **UDP** |
| Proton VPN | ✔ | ✔ | ✔ | X | X | X | X | X | X | X | X |
| Hide.me VPN | ✔ | ✔ | ✔ | ✔ | X | X | ✔ | ✔ | X | X | X |
| Windscribe VPN | ✔ | X | X | ✔ | ✔ | ✔ | X | X | X | ✔ | ✔ |
| Tunnel Bear VPN | ✔ | X | X | ✔ | X | X | X | X | X | X | X |
| Turbo VPN | ✔ | X | X | X | X | X | X | X | X | X | X |
| Cloudflare WARP | X | X | X | X | X | X | X | X | X | X | X |
| Hotspot Shield VPN | ✔ | X | X | ✔ | X | X | X | X | ✔ | X | X |

1. Results and Analysis

Encryption methods are used by VPN applications to protect data while communicating. Only packet sizes, frequencies, and recurring patterns of packets may be used for analysis because of the encrypted payload. Consequently, distinct traffic may be identified with the use of particular payload lengths, sizes, port numbers, and IP addresses. Extensive analysis of packet sizes (patterns, frequencies) discloses the significant operations of the applications, even while privacy and secrecy remain unaffected.

This study's thorough examination of traffic dumps produced patterns in the bytes and payloads associated with various activities. While patterns did appear, they were predictably ineffective in aiding in the reconstruction of the original data. Regulatory bodies, on the other hand, can ascertain the behaviors of VPN applications by observing constant connectivity patterns, thanks to the experimental setup provided in the previous section. Towards the conclusion of the Results and Analysis section is the comprehensive summary.

The following users were created and given descriptions to facilitate the app operations for better understanding:

* **User A:** target user, whose entire set of activities are monitored.
* **User B:** Network Administrator, who monitors the network traffic.

1. Identification of VPN Traffic

We dumped the network data from the laptop that was the target in order to find the activity related to the VPN apps. Without the cryptographic key, it was impossible to decipher the intercepted traffic since it was encrypted. As a result, in order to identify traffic patterns against the various activities, we thoroughly carried out and recorded the various activities. The target device, User A, is used for a variety of tasks.

To gain a deeper comprehension, we examined and categorized target User A's (device's) app behaviors as follows:

1. Handshake with the VPN server
2. Idle State after connection
3. Accessing Google Site
4. Searching on Google
5. YouTube Video Streaming

For this study we have looked at a variety of the VPNs [Table 2] as listed below:

1. Proton VPN
2. Hide.me VPN
3. Windscribe VPN
4. Tunnel-Bear VPN
5. Turbo VPN
6. Cloudflare WARP
7. Hotspot Shield VPN

Now we move on to individual analysis of each VPN application.

### Proton VPN

The firm behind Proton Mail, Proton AG, is based in Switzerland and runs Proton VPN, which is a VPN service. The IPSEC protocol may be used to implement its service, which is compatible with Windows, MacOS, Linux, Android, iOS, and ChromeOS. It also provides a command-line utility for Linux. Proton VPN employs AES-256 encryption with the OpenVPN (UDP/TCP), IKEv2, and WireGuard protocols.

Now we examine and categorize target application behaviors as follows:

#### WireGuard Mode

One VPN protocol, called WireGuard, is a set of guIdleines that controls data encryption and transfer inside virtual private networks (VPNs). Typically, a VPN server and a client—your phone app, for instance—are involved in a WireGuard VPN. Similar to other encryption protocols, WireGuard provides an encrypted tunnel between the client and server through communication with the server. The data that travels between the WireGuard client and server, the two nodes on the network, is encrypted and converted into unintelligible code that requires the right encryption keys to decrypt.

##### Handshake with the VPN server

The Handshake activity of the Proton VPN represents the start of application on the User A’s device. Initially, it was difficult to classify the VPN only traffic from the shared network traffic packets. Therefore, an extensive number of trace files were captured and monitored during the handshake of the Proton VPN application with VPN server.

The process of the above activity is as follows:

* Open the VPN application on User A device and Press Connect.
* Wait for 3 to 4s without interacting with the app.
* Click on Disconnect and close the application.

We observed that a Handshake Initiation packet is sent to the VPN Server from User A. In response, a Handshake Response packet with IP addresses was sent back to the client to establish a connection. In addition, it was observed that User A's Proton VPN was connecting to the server 51820 [Table 3] over a random port on their end.

The following packet patterns with payloads were seen between the server and client endpoints following the negotiation of session keys between the client and server.

* Proton VPN Application (IP: 172.20.10.12) sent packets of (190) bytes with a payload of size (148), to the Proton servers.
* In response, the server (IP: 37.19.205.202) sent packets of (134) bytes to the VPN application with a payload of (92).

To validate the Proton VPN app's connection patterns, the aforementioned action was repeated several times, as seen in Figure 4. Thus, we deduced that the aforementioned byte transfer patterns signify the launch or establishment of the Proton VPN connection.

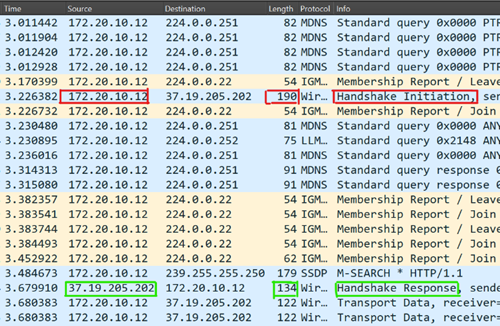


Figure 4: **Proton VPN –** Handshake Packets – WireGuard

##### Idle State after Connection

To notify the patterns of the Proton VPN app when target User A is connected and no activity is being performed, certain flow patterns with fixed payload sizes were noticed. To be assured, these patterns were observed several times in trace files to deduce results as shown in Figure 5. Target User A is highlighted with IP 172.20.10.12. It was noticed that when User A with IP 172.20.10.12 started sending packets to server with IP 37.19.205.202 with 122, 154 and 138 bytes of data packets with 80, 112 and 96 payload size. The server replied by sending the Proton VPN client 138 or 1482 bytes of data packets with payload sizes of 96, 1440. In the meantime, it was observed that even though nothing was being done, the Proton VPN app showed the connected status to the corresponding servers.

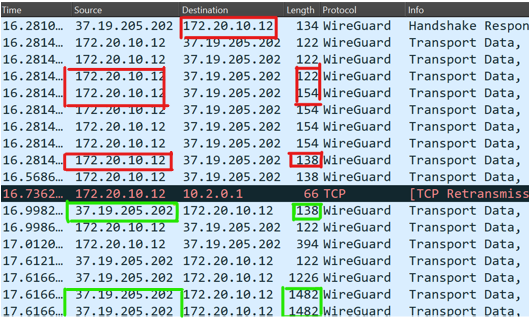


Figure 5: **Proton VPN – Idle State** – WireGuard

##### Accessing Google site

The traffic patterns that User A uses to reach the Google website will be examined in this section. It was always seen that the client at target User A transmitted 138 bytes of the packet in request with 96 bytes of payload to the server after User A accessed the webpage in his or her browser. This request was made to the target server (IP: 45.87.213.226). In answer to the client at target User A's device, the server transmitted 1354 and 122 bytes of packets containing 1312 and 80 bytes of payload. As seen in Figure 6, the server sent the client a series of 1354-byte and 122-byte data packets with payload sizes of 1312 and 80 bytes respectively. This pattern was observed and confirmed repeatedly through multiple iterations.

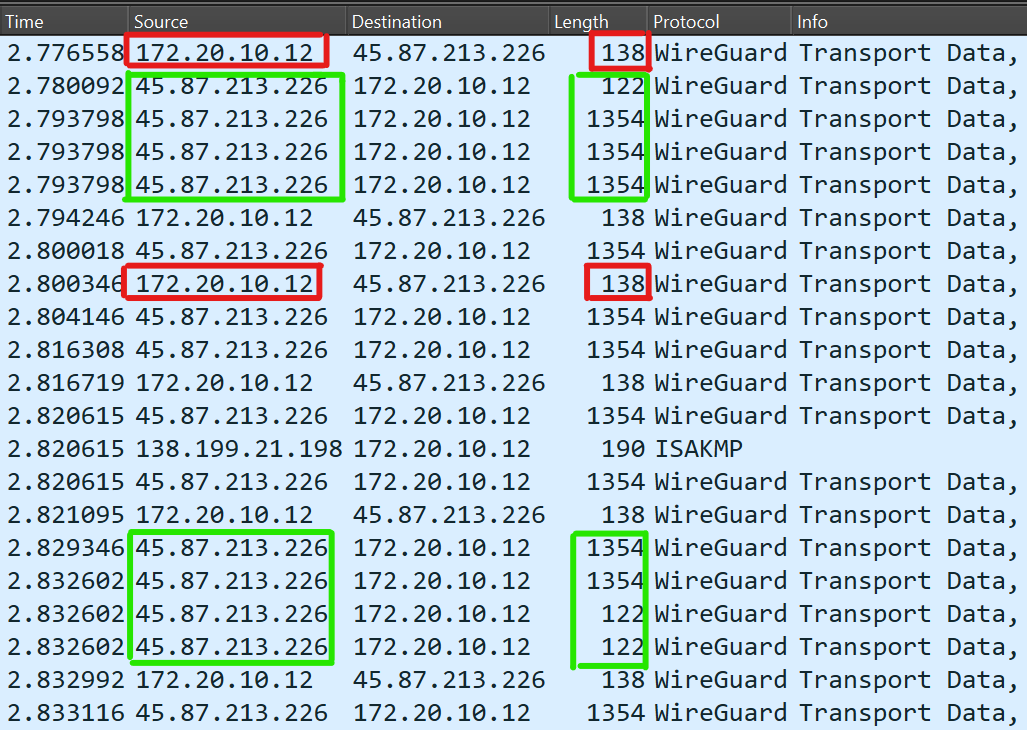


Figure 6: **Proton VPN – Accessing Google Website** – WireGuard

##### Searching on Google

In this section, we will analyze the traffic patterns of User A when he/she performs a search on Google Search Engine. Once User A started searching his/her browser, which would be sent to the target server, it was always noticed that 1354 bytes of packets with 1312 bytes of payload were sent from the server to the client in response to the request. It was also observed that maximum packet length is 1482 byte with 1408 byte of payload. The client at target User A sent 138 bytes of the packet as request with 96 bytes of payload as shown in Figure 7. This pattern was observed and confirmed repeatedly through multiple iterations.

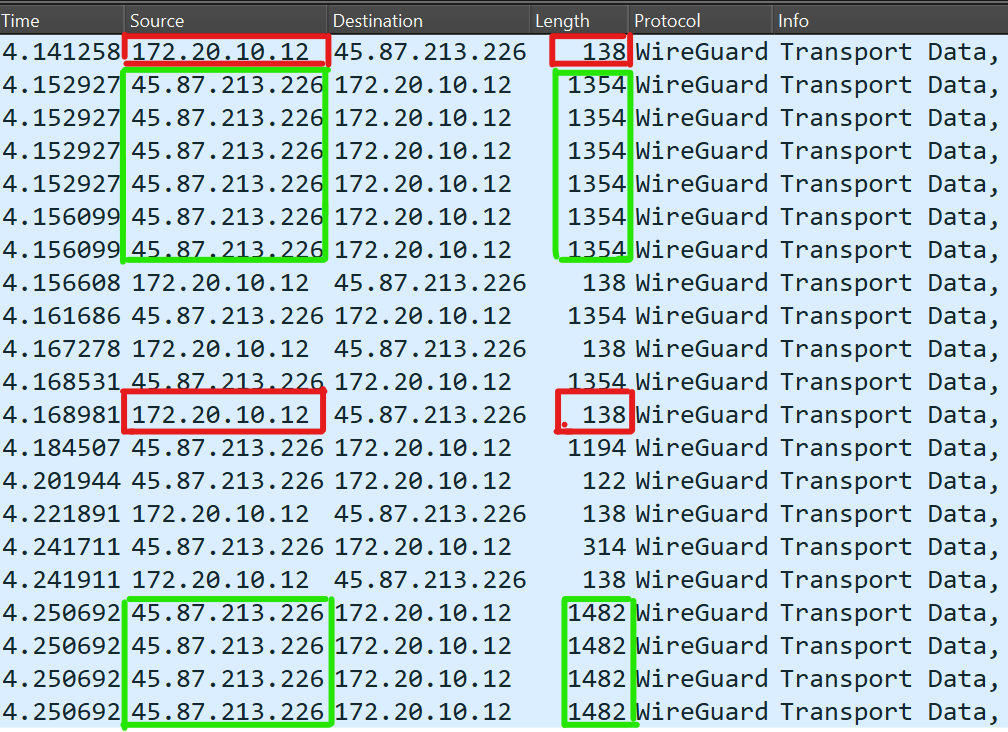


Figure 7: **Proton VPN – Searching on Google** – WireGuard

##### YouTube Video Streaming

To observe the pattern of Proton VPN when a video content was played, as a result certain flow patterns with fixed payload sizes were noticed. Targeted User A sent a request for 138 bytes with payload size of 96 byte for a video. In response, the Proton VPN server responded with 1354-byte packets with 1312 byte of payload. Occasionally, these packets reached the size of 1482 and 138 bytes with payload size of 1408 and 96 bytes. The majority of packets were at size 1354-byte with 1312 byte of payload. This pattern was observed and confirmed repeatedly through multiple iterations as shown in Figure 8.

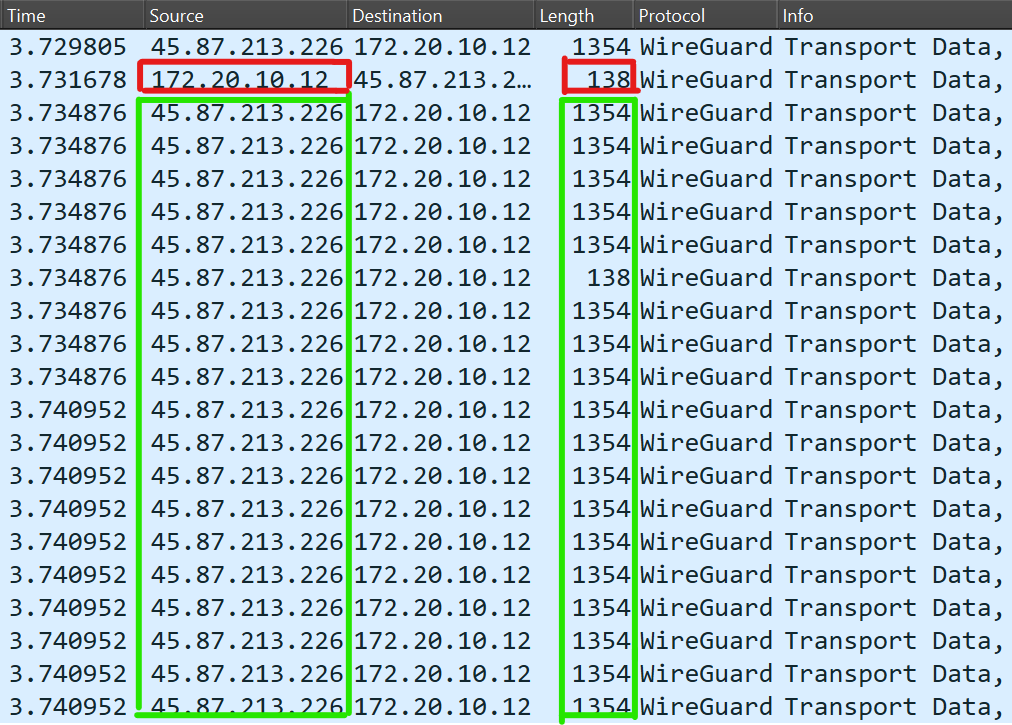


Figure 8: **Proton VPN – YouTube Video Streaming** – WireGuard

#### OpenVPN (UDP) Mode

OpenVPN is a highly customizable protocol that is relatively new. The fact that OpenVPN is open source is its greatest feature. Although the term "open" may not seem like a desirable thing for a privacy tool, it really has a lot of benefits. UDP ports work well for OpenVPN. The receiver cannot request that information be resent or acknowledge receipt of the data via UDP. As a result, UDP can create connections and send data more quickly.

##### Handshake with the VPN server

The establishment of connection activity of the Proton while using OpenVPN (UDP) is relatively different from any other connection formation. As UDP does not require handshake there are no handshake packets, but there must be an exchange.

The process of the above activity is as follows:

* Open the VPN application on User A device and Press Connect.
* Wait for 3 to 4s without interacting with the app.
* Click on Disconnect and close the application.

We observed that the first packets sent from User A to the VPN server uses IAX2 protocol followed by OpenVPN and SIGOMP. In response, server sends a packet of OpenVPN protocol then followed by IAX2 and SIGCOMP packets. In addition, it was observed that User A's Proton VPN was connecting to the server 5060 [Table 3] over a random port on their end.

The following packet patterns with payloads were seen between the server and client endpoints following the negotiation of session keys between the client and server.

* Proton VPN Application (IP: 172.20.10.12) sent packets of (45,80) bytes with a payload of size (3, 38), to the Proton servers.
* In response, the server (IP: 185.107.56.219) sent packets of (45,80) bytes to the VPN application with a payload of (3, 38).

To validate the Proton VPN app's connection patterns, the aforementioned action was repeated several times, as seen in Figure 9. Thus, we deduced that the aforementioned byte transfer patterns signify the launch or establishment of the Proton VPN connection.



Figure 9: **Proton VPN – Handshake Packets – OpenVPN (UDP)**

##### Idle State after Connection

To notify the patterns of the Proton VPN app when target User A is connected and no activity is being performed, certain flow patterns with fixed payload sizes were noticed. To be assured, these patterns were observed several times in trace files to deduce results as shown in Figure 10. Target User A is highlighted with IP 172.20.10.12. It was noticed that when User A with IP 172.20.10.12 started sending packets to server with IP 185.107.56.219 with 106 bytes of data packets with 64 payload size which is more notable. The server replied by sending the Proton VPN client 106 and 1460 bytes of data packets with payload sizes of 64, 1418. In the meantime, it was observed that even though nothing was being done, the Proton VPN app showed the connected status to the corresponding servers.



Figure 10: **Proton VPN – Idle State** – OpenVPN (UDP)

##### Accessing Google site

The traffic patterns that User A uses to reach the Google website will be examined in this section. It was always seen that the client at target User A transmitted 127 bytes of the packet in request with 85 bytes of payload to the server after User A accessed the webpage in his or her browser. This request was made to the target server (IP: 185.107.56.219). In answer to the client at target User A's device, the server transmitted 1344 bytes of packets containing 1302 bytes of payload. As seen in Figure 11 this pattern was observed and confirmed repeatedly through multiple iterations.

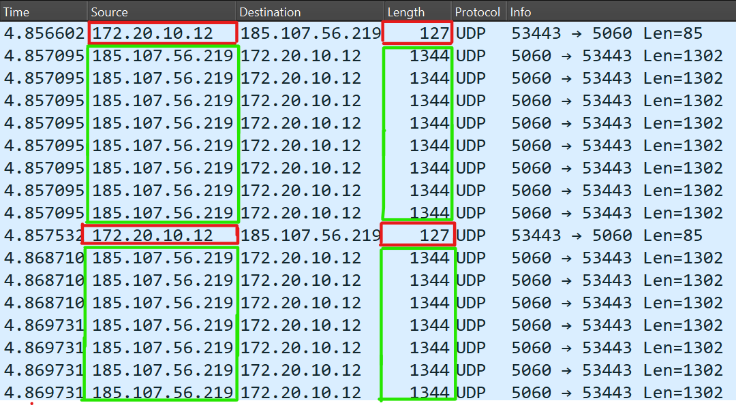


Figure 11: **Proton VPN – Accessing Google Website** – OpenVPN (UDP)

##### Searching on Google

In this section, we will analyze the traffic patterns of User A when he/she performs a search on Google Search Engine. Once User A started searching his/her browser, which would be sent to the target server, it was noticed that 1344 bytes of packets with 1302 bytes of payload were sent from the server to the client in response to the request. It was also observed that maximum packet length is 1460 byte with 1418 byte of payload. The client at target User A sent 126 bytes of the packet as request with 84 bytes of payload as shown in Figure 12. This pattern was observed and confirmed repeatedly through multiple iterations.

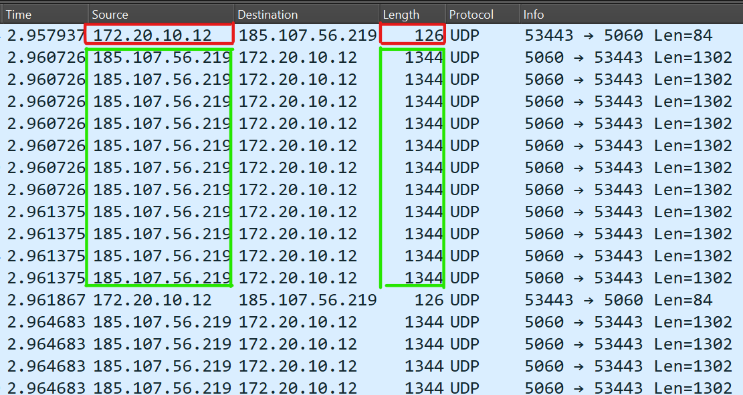


Figure 12: **Proton VPN – Searching on Google** – OpenVPN (UDP)

##### YouTube Video Streaming

To observe the pattern of Proton VPN when a video content was played, as a result certain flow patterns with fixed payload sizes were noticed. Targeted User A sent a request for 128, 126 bytes with payload size of 86, 84 bytes for a video. In response, the Proton VPN server responded with 1344-byte packets with 1302 byte of payload. Occasionally, these packets reached the size of 1460 bytes with payload size of 1418 bytes. The majority of packets were at size 1344 byte with 1302 byte of payload. This pattern was observed and confirmed repeatedly through multiple iterations as shown in Figure 13.

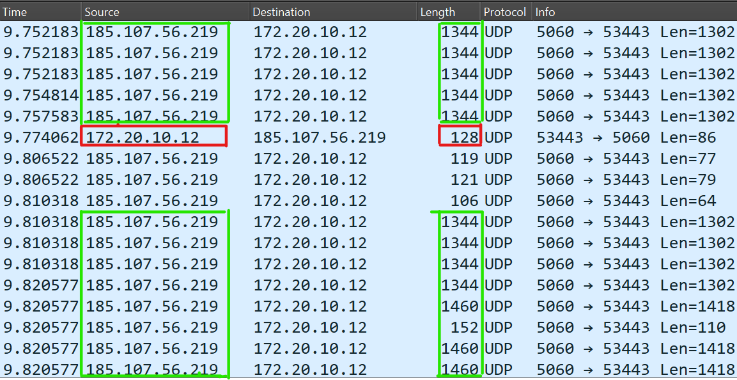


Figure 13: **Proton VPN – YouTube Video Streaming** – OpenVPN (UDP)

#### OpenVPN (TCP) Mode

OpenVPN is a highly customizable protocol that is relatively new. The fact that OpenVPN is open source is its greatest feature. Although the term "open" may not seem like a desirable thing for a privacy tool, it really has a lot of benefits. In contrast to UDP, TCP makes sure that the receiver receives the data in the right format and sequence before allowing the request to be made again. As a result, latency increases at the price of higher dependability.

##### Handshake with the VPN server

The Handshake activity of the Proton VPN represents the start of application on the User A’s device. Initially, it was difficult to classify the VPN only traffic from the shared network traffic packets. Therefore, an extensive number of trace files were captured and monitored during the handshake of the Proton VPN application with VPN server.

The process of the above activity is as follows:

* Open the VPN application on User A device and Press Connect.
* Wait for 3 to 4s without interacting with the app.
* Click on Disconnect and close the application.

We observed that a standard DNS query was sent to access the Proton VPN server from the client end, i.e., vpn-api.proton.me OPT. Between the client and server, a TCP connection was formed in three stages: SYN, SYN ACK, and ACK. Following that, TLSv1.3 was used for the TLS handshake. The client and server exchanged certificates and session keys. The client then sends the server an encrypted handshake message after exchanging the key change cypher specifications. The server responded by sending the client device a new session ticket, modifying the cypher specs, and sending an encrypted handshake message over TLS. In response, a server’s IP address was sent back to the client to establish a connection. We observed that a Handshake Initiation packet is sent to the VPN Server from User A. In response, a Handshake Response packet with IP addresses was sent back to the client to establish a connection. In addition, it was observed that User A's Proton VPN was connecting to the server 443 [Table 3] over a random port on their end.

The following packet patterns with payloads were seen between the server and client endpoints following the negotiation of session keys between the client and server.

* Proton VPN Application (IP: 172.20.10.12) sent packets of (66, 54) bytes with a payload of size (34, 34), to the Proton servers.
* In response, the server (IP: 185.159.159.148) sent packets of (66, 54) bytes to the VPN application with a payload of (34, 34).

To validate the Proton VPN app's connection patterns, the aforementioned action was repeated several times, as seen in Figure 14. Thus, we deduced that the aforementioned byte transfer patterns signify the launch or establishment of the Proton VPN connection.

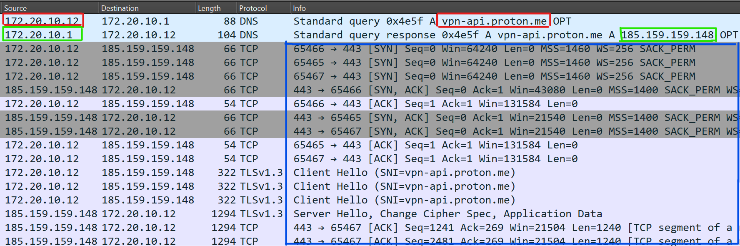


Figure 14: **Proton VPN –** Handshake Packets – OpenVPN (TCP)

##### Idle State after Connection

To notify the patterns of the Proton VPN app when target User A is connected and no activity is being performed, certain flow patterns with fixed payload sizes were noticed. To be assured, these patterns were observed several times in trace files to deduce results as shown in Figure 15. Target User A is highlighted with IP 172.20.10.12. It was noticed that when User A with IP 172.20.10.12 started sending packets to server with IP 185.159.159.148 with 54 bytes of data packets with 34 payload size which is more notable. The server replied by sending the Proton VPN client 54 bytes of data packets with payload sizes of 34. In the meantime, it was observed that even though nothing was being done, the Proton VPN app showed the connected status to the corresponding servers using SSL packets.

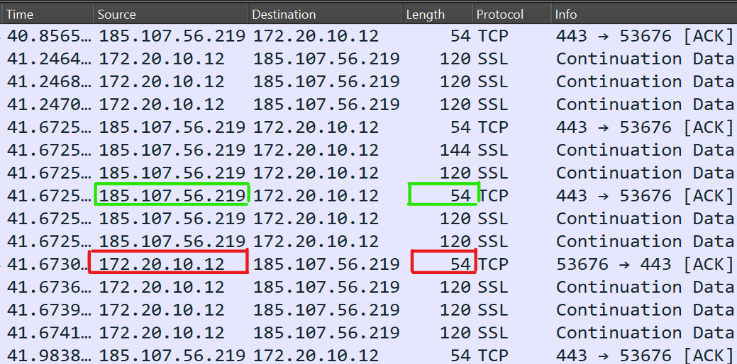


Figure 15: **Proton VPN – Idle State** – OpenVPN (TCP)

##### Accessing Google site

The traffic patterns that User A uses to reach the Google website will be examined in this section. It was always seen that the client at target User A transmitted 54 and 120 bytes of the packet in request with 34 and 66 bytes of payload to the server after User A accessed the webpage in his or her browser. This request was made to the target server (IP: 193.148.16.2). In answer to the client at target User A's device, the server transmitted 1349 bytes of packets containing 1295 bytes of payload. As seen in Figure 16 this pattern was observed and confirmed repeatedly through multiple iterations.



Figure 16: **Proton VPN – Accessing Google Website** – OpenVPN (TCP)

##### Searching on Google

In this section, we will analyze the traffic patterns of User A when he/she performs a search on Google Search Engine. Once User A started searching his/her browser, which would be sent to the target server, it was noticed that 1348 and 54 bytes of packets with 1294 and 34 bytes of payload were sent from the server to the client in response to the request. It was also observed that maximum packet length is 1358 byte with 1304 byte of payload. The client at target User A sent 54 bytes of the packet as request with 34 bytes of payload as shown in Figure 17. This pattern was observed and confirmed repeatedly through multiple iterations.

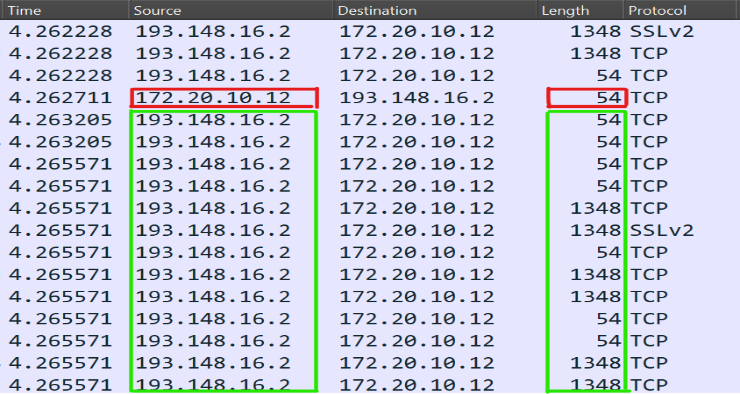


Figure 17: **Proton VPN – Searching on Google** – OpenVPN (TCP)

##### YouTube Video Streaming

To observe the pattern of Proton VPN when a video content was played, as a result certain flow patterns with fixed payload sizes were noticed. Targeted User A sent a request for 54 bytes (which is the most prominent packet size) with payload size of 34 bytes for a video. In response, the Proton VPN server responded with 1358 and 1448 bytes packets with 1304 and 1394 bytes of payload. Occasionally, these packets reached the size of 1448 bytes with payload size of 1439 bytes. The majority of packets were at size 1358 byte with 1304 byte of payload. This pattern was observed and confirmed repeatedly through multiple iterations as shown in Figure 18.

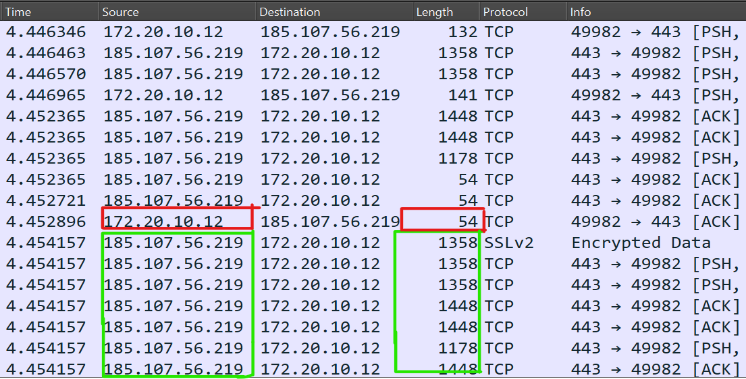


Figure 18: **Proton VPN – YouTube Video Streaming** – OpenVPN (TCP)

### Hide.me VPN

Hide.me VPN provides encryption, wi-fi security, and privacy protection for a genuinely private online browsing experience, no matter where you are. It makes use of SSTP, WireGuard, OpenVPN, SoftEther VPN, and IKEv2 [Table 2].

In this study, we thoroughly examined Hide.me VPN across a range of criteria, adhering to the same methods used to assess ProtonVPN. We carefully watched and categorised our target user, User A,'s application behaviours on their device in order to have a thorough insight. These included the handshake procedure with the VPN server, the idle state following the formation of a connection, browsing Google websites, using Google to search, and watching YouTube videos. We have condensed the in-depth observations into Table 3 for convenience of reference and interpretation in order to simplify the presenting of our findings.

### Windscribe VPN

Based in Canada, Windscribe is a cross-platform, commercial virtual private network (VPN) service provider with operations throughout the globe. In its manual setups and applications, Windscribe makes use of IKEv2, WireGuard, and OpenVPN protocols [Table 3]. P2P file sharing is supported on Windscribe servers, who advertise themselves as a no-log VPN service in their privacy statement.

Following up on our research into VPN services, similar to the methodology used for Proton VPN and Hide.me VPN, we carefully evaluated Windscribe VPN based on a number of performance indicators. We examined User A's device app behaviors across several important factors in order to gain further insight into the operation and effectiveness of Windscribe VPN. To improve reader accessibility and understanding and to enable a concise presentation of our findings, we have summarized the in-depth observations in Table 3.

### Tunnel-Bear VPN

The simplest VPN in the world to use is Tunnel-Bear. The way Tunnel-Bear operates is by enabled to connect to any place worldwide over an encrypted tunnel. Users may browse the internet as if they were physically in the nation they are connected to, and their true IP address stays disguised while you are connected. It uses WireGuard and IKEv2.

We thoroughly investigated Tunnel-Bear VPN using the same approach we used to evaluate ProtonVPN, Hide.me VPN and Windscribe VPN. Our research included a detailed examination of the VPN service's functionality according to several different criteria. We have condensed the in-depth observations into Table 3 to simplify the presentation of our findings and offer a clear and concise summary for further examination and comparison.

### Turbo VPN

Turbo VPN, created in 2018 by Innovative Connecting, a Singaporean firm, has received about 180,000 ratings on the Apple App Store and over 100 million downloads on the Google Play Store. It uses WireGuard Protocol.

Following the established process that we used to assess Previous VPN, we looked more closely at Turbo VPN. We meticulously examined the VPN service's performance in relation to a number of different metrics as part of our study. We did this by carefully observing and classifying our assigned user, User A,'s application behaviors on their device. In order to improve readability and enable comparison, we have condensed our in-depth observations into Table 3, which provides a concise synopsis for thorough comprehension and analysis.

### Cloudflare WARP

Much attention was aroused when Cloudflare defined WARP as a fast, thin, mobile-only VPN that doesn't drain your phone's battery. People may enjoy the internet quicker, more securely, and privately with the help of the Cloudflare WARP client. The WARP client, which stands in between your device and the Internet, offers a variety of connection types to accommodate various requirements. It does not use the traditional protocols like other VPNs.

Following the established process that we used to assess Previous VPN, we looked more closely at Cloudflare WARP. In order to improve readability and enable comparison, we have condensed our in-depth observations into Table 3, which gives a clear summary for further study and assessment.

### Hotspot Shield VPN

AnchorFree, Inc. is the company behind Hotspot Shield, a free VPN service. Through one of its supported public VPN servers, the Hotspot Shield client creates an encrypted VPN connection that allows the user to access the Internet. It makes use of IKEv2, WireGuard, and Hydra protocols.

Following the same protocol that was used to assess ProtonVPN, Windscribe VPN, Tunnel-Bear VPN, and Turbo VPN, we also looked at Hotspot Shield VPN. We performed a comprehensive analysis of the VPN service's performance across several factors by using a methodical methodology. These metrics included the handshake with the VPN server, the idle state following the formation of the connection, the use of Google sites, Google searches, and YouTube video streaming. We condensed our in-depth findings into Table 3 to ensure clarity and make comparison easier. This gives a clear summary for further study and assessment.

**Table 3:**

**Identification of VPN**

|  |  |  |  |
| --- | --- | --- | --- |
| **VPN** | **Port No.** | **Packets Length** | **IP Ranges** |
| Proton VPN | 51820, 1224, 88, 4500, 4569, 500, 1194, 5060, 7770, 443, 8443 | 122, 138, 154, 1306, 1354, 1482, 127, 128, 1344, 1428, 1433, 1460, 1354, 1358, 1448, 1507 | Annex A |
| Hide.me VPN | 432, 443, 30xx, 4000xx (with 444 on User A) | 122, 138, 170, 1050, 1354, 1466, 174, 1358, 1486, 126, 1344, 1346, 1458, 1499, 1514, 54, 66, 1296, 1362, 1406 | Annex A |
| Windscribe VPN | 80, 53, 123, 1194, 65142, 54784, 587, 21,22, 3306, 54786, 1194, 8443, 443, 4500 | 122, 138, 154, 1354, 1358, 1494, 118, 1478, 1514, 4250, 60, 1384, 1344, 1392 | Annex A |
| Tunnel-Bear VPN | 51820 (with 51820 on User A) and 4500 (with 4500 on User A) | 114, 122, 138, 1352, 1354, 1494, 126, 1358, 1486 | Annex A |
| Turbo VPN | 443, 8080 | 56, 66, 1514 | Annex A |
| Cloudflare Warp | 2408, 1029, many | 74, 114, 126, 651, 1352, 1354 | 162.159.192.2 |
| Hotspot Shield VPN | 5000, 4500 (with 4500 on User A), 443 | 114, 122, 1318, 1354, 1450, 1494, 126, 142, 222, 398, 1486, 1494, 1044, 1279, 1328, 1389, 1514 | Annex A |

1. Discussion and Use Cases

In this section, we will discuss various use cases to validate and verify the identified evidence from the proposed forensic strategy regarding the VPN traffic analysis.

## Blocking Tunnel Bear VPN

While performing all the above-mentioned activities, certain results are deduced. For example, during multiple activities, we observed that tunnel bear operates on the port 4500 on both server and client side. While monitoring the network packets during the establishment of connection between the User A and Tunnel Bear VPN app, it was noted that the app always connects to one of the servers mentioned in the DNS query sent, i.e., www.tunnelbear.com. It was also observed that an application establishes a connection with one of these servers using 4500 ports. The details of all those servers can be found in packets details in the Wireshark as shown in Figure 19 & 20.

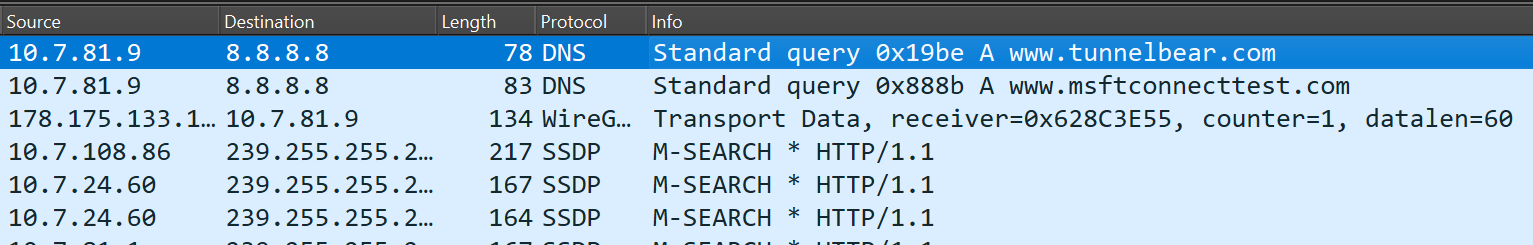


Figure 19: **Wireshark Packets**

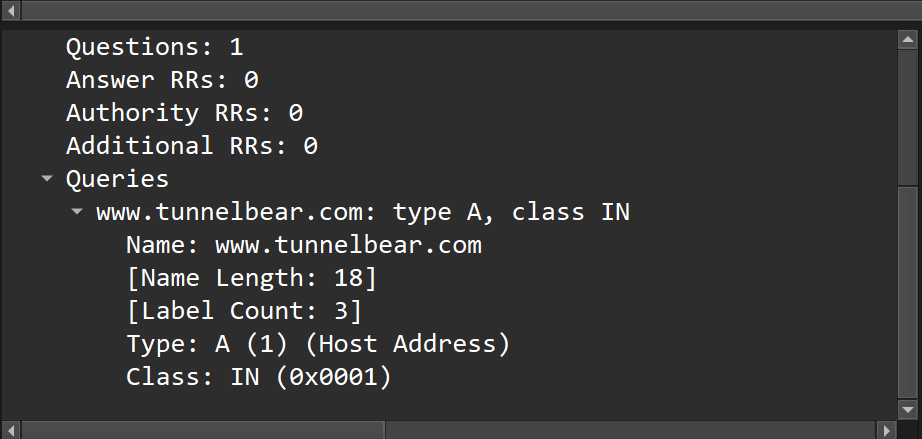


Figure 20: **DNS Packet**

Now, we update the firewall rules such that it blocks any UDP from LAN Subnets from 4500 to any destination at 4500 port, Figure 21. After which, when the Tunnel Bear VPN application attempts to connect with its servers but PfSense Firewall blocks the traffic because the firewall always provides an edge to control and monitor the ongoing activities within the network.

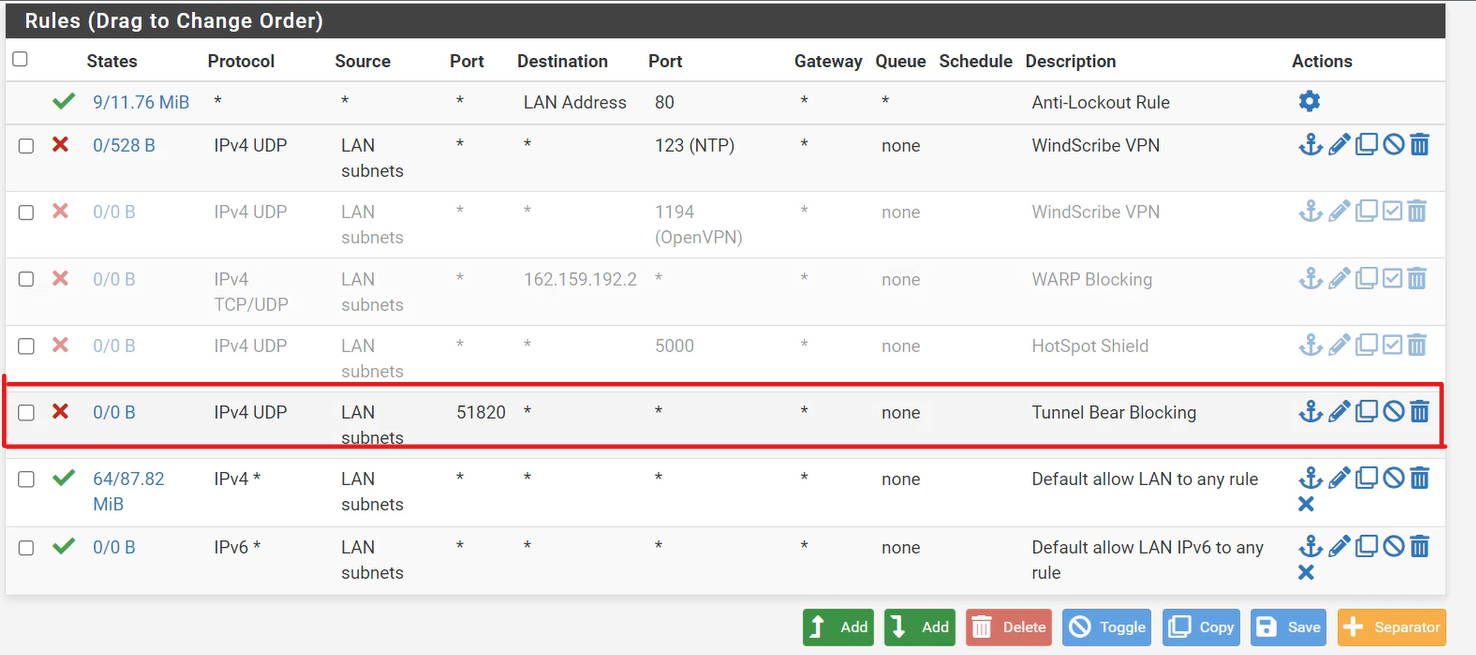


Figure 21: **PfSense - Tunnel Bear Block**

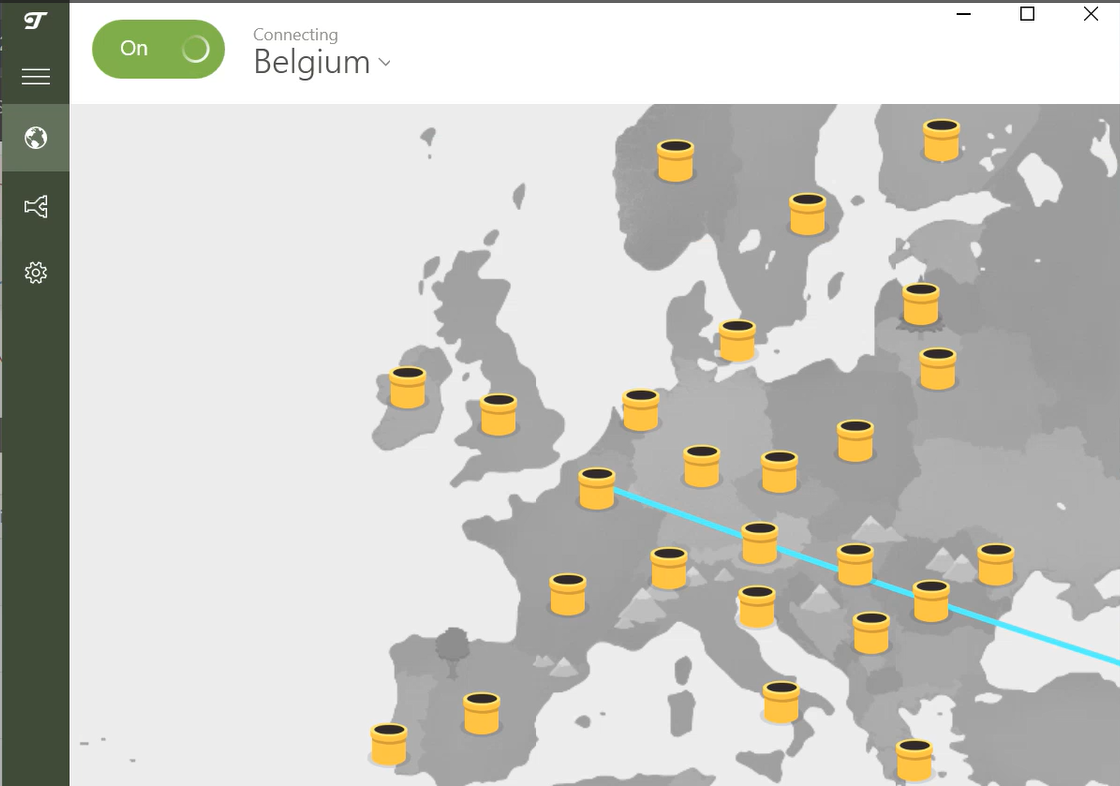


Figure 22: **Blocked Tunnel Bear**

The rules are as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Protocol** | **Source Port** | **Destination Port** | **Action** | **Observation** |
| 1 | TCP | Any | Any | Default allow LAN to any rule. | All services work smoothly |
| 2 | UDP | 4500 | 4500 | Block the entire traffic which work on 4500 ports on both ends. | Tunnel Bear Connection Failed |

## Blocking Cloudflare WARP

While performing all the above-mentioned activities, certain results are deduced. However, during the monitoring of network packets for the establishment of connections involving Cloudflare WARP, it was noticed that the application connects to IP address 162.159.192.2. Unlike Tunnel Bear, which consistently utilizes port 4500, Cloudflare WARP demonstrates flexibility in port usage, connecting to different ports during its operation.

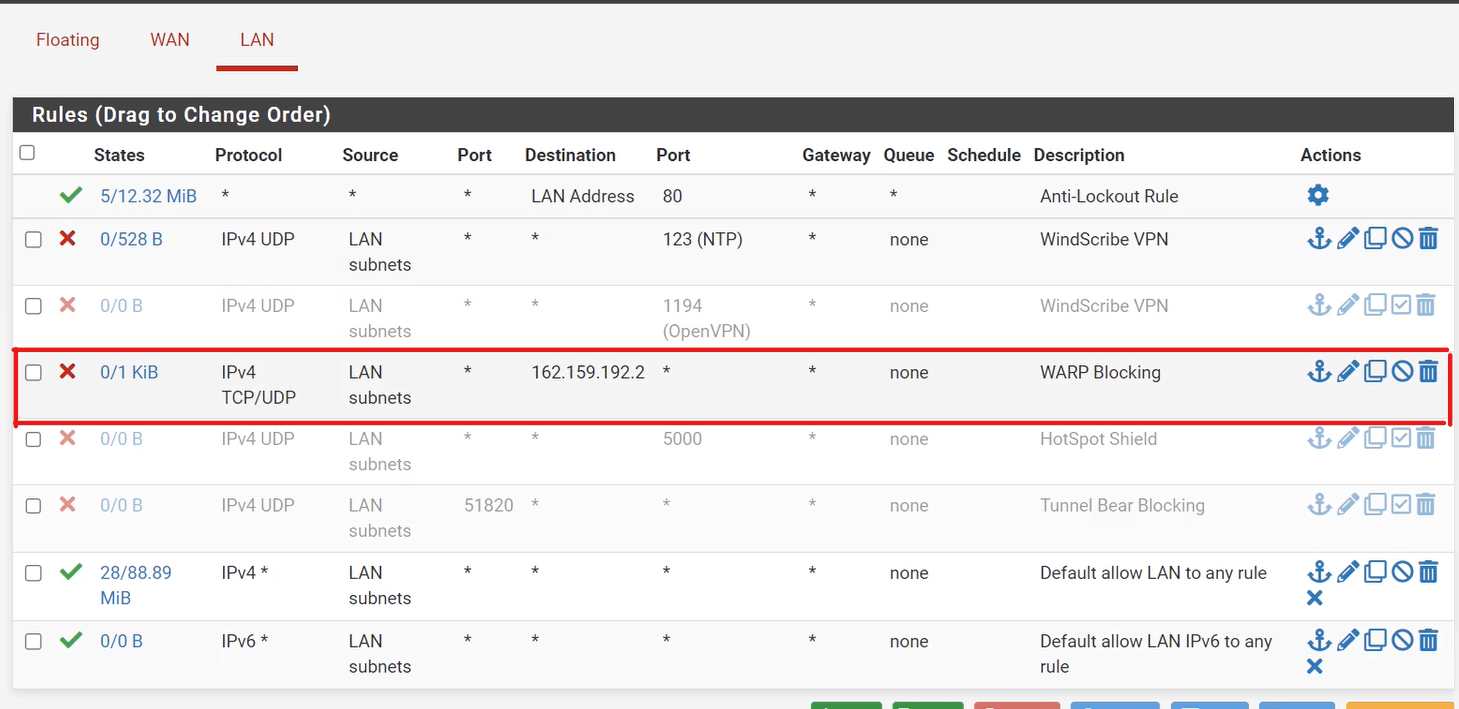


Figure 23: **PfSense - WARP Blocking**

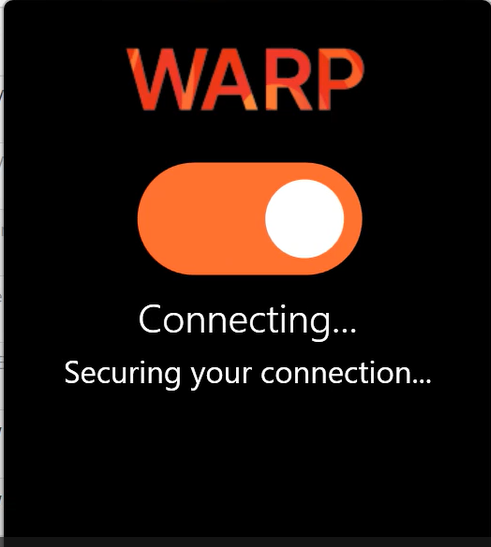


Figure 24:  **Blocked Cloudflare WARP**

Therefore, we utilize the PfSense Firewall to block any traffic connection Figure 23 from Lan subnet to 162.159.192.2 destination on any ports. The firewall rules are as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Protocol** | **Source Port** | **Destination Port** | **Action** | **Observation** |
| 1 | TCP | Any | Any | Default allow LAN to any rule. | All services work smoothly |
| 2 | UDP  /TCP | Any | Any | Block the entire traffic moving to 162.159.192.2 port. | Cloudflare Warp connection Failed |

## Other VPN Blocking

While conducting network analysis for Other VPNs services, it was observed that these VPNs exhibit distinct patterns in their network traffic behavior. Although they may not have the same port or IP address as Tunnel Bear or Cloudflare WARP, they still possess identifiable characteristics that can be leveraged for blocking as shown in Table 3.

To block all Other VPN services, we can adopt a strategy similar to the approach used for Tunnel Bear and Cloudflare WARP. Once these patterns are identified, firewall rules can be configured on PfSense to block traffic associated with services. This involve blocking specific ports, IP addresses, or even applying more advanced filtering techniques based on packet inspection and analysis.

1. Benefits of Proposed Strategy

The proposed strategy outlined in this study offers several significant benefits:

* **Comprehensive Insight**: By capturing and analyzing network traffic, the strategy provides a comprehensive understanding of VPN application behavior, including handshake processes, idle states, and specific user activities such as accessing websites and streaming content.
* **Effective Traffic Identification**: Despite encryption, the strategy effectively identifies VPN traffic through analysis of packet sizes, frequencies, and patterns, facilitating the differentiation of VPN-related activities from other network traffic.
* **Enhanced Security Monitoring**: Through continuous monitoring of network traffic, the strategy enables proactive detection of potential security threats or anomalies, thereby enhancing overall network security posture.
* **Informed Network Management**: The insights gained from the analysis inform network management decisions, such as optimizing network configurations, implementing new security measures, or addressing performance issues related to VPN usage.
* **Privacy Preservation**: While providing valuable insights into VPN behavior, the strategy respects user privacy by focusing on network traffic analysis rather than attempting to decrypt encrypted payloads, ensuring compliance with privacy regulations and ethical standards.

1. Limitations and Future Work

The analysis of encrypted payloads in VPN traffic is the main source of constraint for this study. The capacity to completely comprehend the content and purpose of connections is restricted by the encrypted nature of payload data, despite the fact that packet sizes, ports, and patterns provide insights about VPN behavior. The firewall rules and network settings that are put in place will have a significant impact on how well the suggested approach works. The effectiveness of the proposed strategy heavily relies on the specific network configuration and firewall rules implemented. Variations in network setups may result in different traffic patterns and behaviors, potentially impacting the generalizability of findings. Scalability and performance concerns may also surface as network traffic volumes increase, scalability and performance issues may arise in capturing, storing, and analyzing large datasets.

The version dependence is the main drawback. Forensics analysis of VPN applications is dependent on the specific version of these VPN applications. As a result, the suggested work also depends on version. The network traffic patterns of the VPN may alter and diverge from the findings reported in this paper if a new version is released that significantly alters the network's behavior or structure.

In the future, we will work on exploring techniques to adapt to evolving encryption standards while maintaining effective traffic analysis capabilities by leveraging on machine learning algorithms for traffic analysis which could enhance the ability to identify and classify VPN behaviors accurately. We will work incorporate P4 language to provide more detailed DPI based on the P4 Switch with configurable attributes including but not limited to ports and associated IPs, etc., to generalize the proposed strategy and possibly make it portable to new versions of the application.

Conclusion

In conclusion, our study provides a thorough approach to deciphering encrypted VPN data and provides insightful information about how different VPN providers operate. We have shown the efficacy of our technique in comprehending VPN application behavior by methodically analyzing idle states, handshake protocols, and user actions on various VPN platforms. We improve network security and administration by putting firewall rules into place to prohibit particular ports and IP addresses connected to VPN services. This allows for the proactive identification of possible threats and well-informed decision-making.

Our technique offers a strong basis for future research in network security and digital forensics, even with its inherent constraints (e.g., payload data encryption and reliance on certain network configurations). Future research projects may use machine learning techniques to adjust to changing encryption standards and analyze communications more accurately. Through constant improvement and modification of our approach, we can remain ahead of new obstacles in the field of encrypted VPN traffic analysis.

APPENDIX

Note: In IPs “x” is any number from 0-9.

1. Proton VPN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 103.125.235.**xx** | 185.107.80.**xxx** | 212.38.97.**xxx** | 89.187.177.**50-100** | 93.190.138.**150-200** |
| 109.236.81.**xxx** | 185.159.156.**xxx** | 212.8.243.**xxx** | 89.187.179**.xxx** | 93.190.140.**xxx** |
| 138.199.21.**xxx** | 185.159.158.**xxx** | 212.8.253.**xxx** | 89.187.180.**xxx** |  |
| 138.199.22.**xxx** | 185.177.124.**xxx** | 217.138.206.**xxx** | 89.187.185.**xxx** |  |
| 138.199.50.**xxx** | 185.177.125.**xxx** | 217.23.3**.xxx** | 89.38.97.**xxx** |  |
| 138.199.52.**xxx** | 185.177.126.**xxx** | 37.120.217.**xxx** | 89.38.99.**xxx** |  |
| 138.199.7.**xxx** | 185.182.193.**xxx** | 37.120.244.**xxx** | 89.39.104.**xxx** |  |
| 143.244.44.**xxx** | 185.183.33.**xxx** | 37.19.200.**xxx** | 89.39.106.xxx |  |
| 146.70.147.**xxx** | 185.183.34**.xxx** | 37.19.201.**xxx** | 89.39.107.**xxx** |  |
| 146.70.174.**xxx** | 185.230.126.**xx** | 37.19.205.**xxx** | 89.45.4.**xxx** |  |
| 146.70.202**.xxx** | 185.236.200**.xxx** | 37.19.221.**xxx** | 93.190.138.**150-200** |  |
| 146.70.45.**xxx** | 190.2.130**.xxx** | 38.132.103.**xxx** | 93.190.140.**xxx** |  |
| 149.34.244.**xxx** | 190.2.133.**xxx** | 45.14.71.**xxx** | 93.190.138.**xxx** |  |
| 156.146.51.**xxx** | 193.148.18**.xxx** | 45.87.214.**xxx** | 93.190.140.**xxx** |  |
| 156.146.54.**xxx** | 192.2.132.**xxx** | 45.89.173.**xxx** | 93.190.138.**xxx** |  |
| 165.150.169.**xxx** | 195.181.162.**xxx** | 46.166.182.**xxx** | 93.190.140.**xxx** |  |
| 169.150.196.**xxx** | 195.181.163.**xxx** | 77.247.178.**xxx** | 93.190.138.**xxx** |  |
| 169.150.218.**xxx** | 198.148.18.**xxx** | 79.110.55.**xxx** | 93.190.140.**xxx** |  |
| 185.107.56.**xxx** | 212.102.35.**xxx** | 87.249.134.**xxx** | 93.190.138.**xxx** |  |
| 185.107.57.xxx | 212.102.51.**xxx** | 89.187.170.**xxx** | 93.190.140.**xxx** |  |

1. Windscribe VPN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 103.10.197.xxx | 143.244.44.xxx | 162.222.198.xxx | 193.27.14.xxx | 212.102.xxx.xxx |
| 104.129.xxx.xxx | 146.70.xxx.xxx | 169.150.19x.xxx | 194.59.249.xxx | 212.103.xxx.xxx |
| 104.223.xxx.xxx | 149.102.229.xxx | 172.98.68.xxx | 198.55.126.xxx | 217.138.25x.xxx |
| 104.233.xxx.xxx | 149.36.xx.xxx | 173.44.36.xxx | 198.8.85.xxx | 223.123.88.xxx |
| 104.245.146.xxx | 149.50.208.xxx | 185.120.147.xxx | 198.96.95.xxx | 23.105.1xx.xxx |
| 107.150.xx.xxx | 149.57.xx.xxx | 185.156.173.xxx | 2.58.44.xxx | 27.122.1x.xxx |
| 107.161.86.xxx | 154.47.26.xxx | 185.189.113.xxx | 204.44.122.xxx | 37.120.2xx.xxx |
| 107.7.60.xxx | 155.94.xxx.xxx | 185.217.68.xxx | 207.244.91.xxx | 45.87.21x.xxx |
| 138.199.xx.xxx | 155.97.217.xxx | 185.236.200.xxx | 208.77.22.xxx | 68.235.3x.xxx |
| 139.199.47.xxx | 161.129.70.xxx | 185.253.97.xxx | 208.78.41.xxx | 68.235.4x.xxx |
| 68.174.103.xxx | 71.19.25x.xxx | 77.81.136.xxx | 84.17.43.xxx | 84.17.50.xxx |
| 86.106.87.xxx | 89.41.26.xxx | 89.47.62.xxx | 91.2xx.xxx.xxx | 92.119.117.xxx |

1. Tunnel-Bear VPN

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 103.50.33.xxx | 134.209.xx.xxx | 143.110.1xx.xxx | 159.2xx.xxx.xxx | 165.2xx.xxx.xxx |
| 104.131.xx.xxx | 137.184.xx.xxx | 143.198.26.xxx | 159.65.xxx.xxx | 167.172.xxx.xxx |
| 104.248.162.xxx | 138.197.xxx.xxx | 146.190.xxx.xxx | 159.89.xxx.xxx | 167.71.xxx.xxx |
| 107.170.xx.xxx | 139.180.xxx.xxx | 149.28.164.xxx | 161.35.16x.xxx | 174.138.xxx.xxx |
| 134.122.xx.xxx | 139.59.xxx.xxx | 157.2xx.xxx.xxx | 162.243.xxx.xxx | 188.166.xxx.xxx |
| 192.241.xxx.xxx | 209.97.xxx.xxx | 45.55.xxx.xxx | 65.225.56.xxx | 68.183.58.xxx |
| 200.25.50.xxx | 37.120.234.xxx | 46.101.xxx.xxx | 67.205.185.xxx | 106.189.xx.xxx |

1. Hide.me VPN

|  |  |
| --- | --- |
| 146.70.106.xx | 31.13.189.xxx |
| 146.70.118.xx | 37.120.192.xxx |
| 146.70.128.xx | 45.141.152.xxx |
| 185.216.33.xxx | 45.152.18x.xxx |
| 217.138.194.xxx | 72.10.160.xxx |
| 217.138.195.xxx | 72.10.162.xxx |
| 217.138.208.xxx | 95.174.67.xxx |
| 217.138.215.xxx |  |

1. Turbo VPN

|  |
| --- |
| 134.209.212.xxx |
| 139.99.90.xxx |
| 157.245.218.xxx |
| 159.89.180.xxx |
| 162.243.x.xxx |
| 165.227.xxx.xxx |
| 51.79.xxx.xxx |

1. Hotspot Shield VPN

|  |  |
| --- | --- |
| 103.105.164.xxx | 198.145.2xx.xxx |
| 103.216.198.xxx | 204.14.xx.xxx |
| 104.232.xxx.xxx | 217.151.xxx.xxx |
| 107.182.231.xxx | 23.249.xxx.xxx |
| 146.70.1xx.xxx | 45.56.1xx.xxx |
| 173.244.217.xxx | 64.141.xx.xxx |
| 185.208.152.xxx | 89.11x.xxx.xxx |
| 192.119.160.xxx | 92.119.xxx |
| 20.190.147.xxx | 13.38.12x.xxx |
| 63.141.48.xx | 51.44.41.xxx |

1. Cloudflare WARP

|  |
| --- |
| 162.159.192.2 |

REFERENCES AND FOOTNOTES

1. REFERENCES

do not use automatic endnotes in Word, rather, type the reference list at the end of the paper using the “References” style.

1. FOOTNOTES

ASD

FIRST A. AUTHOR (M’76–SM’81–F’87) and all authors may include biographies. Biographies are often not included in conference-related papers. This author became a Member (M) of IEEE in 1976, a Senior Member (SM) in 1981, and a Fellow (F) in 1987. The first paragraph may contain a place and/or date of birth (list place, then date). Next, the author’s educational background is listed. The degrees should be listed with type of degree in what field, which institution, city, state, and country, and year the degree was earned. The author’s major field of study should be lower-cased.

The second paragraph uses the pronoun of the person (he or she) and not the author’s last name. It lists military and work experience, including summer and fellowship jobs. Job titles are capitalized. The current job must have a location; previous positions may be listed without one. Information concerning previous publications may be included. Try not to list more than three books or published articles. The format for listing publishers of a book within the biography is: title of book (publisher name, year) similar to a reference. Current and previous research interests end the paragraph.

The third paragraph begins with the author’s title and last name (e.g., Dr. Smith, Prof. Jones, Mr. Kajor, Ms. Hunter). List any memberships in professional societies other than the IEEE. Finally, list any awards and work for IEEE committees and publications. If a photograph is provided, it should be of good quality, and professional-looking. Following are two examples of an author’s biography.

**SECOND B. AUTHOR** was born in Greenwich Village, New York, NY, USA in 1977. He received the B.S. and M.S. degrees in aerospace engineering from the University of Virginia, Charlottesville, in 2001 and the Ph.D. degree in mechanical engineering from Drexel University, Philadelphia, PA, in 2008.

From 2001 to 2004, he was a Research Assistant with the Princeton Plasma Physics Laboratory. Since 2009, he has been an Assistant Professor with the Mechanical Engineering Department, Texas A&M University, College Station. He is the author of three books, more than 150 articles, and more than 70 inventions. His research interests include high-pressure and high-density nonthermal plasma discharge processes and applications, microscale plasma discharges, discharges in liquids, spectroscopic diagnostics, plasma propulsion, and innovation plasma applications. He is an Associate Editor of the journal *Earth*, *Moon*, *Planets*, and holds two patents.

Dr. Author was a recipient of the International Association of Geomagnetism and Aeronomy Young Scientist Award for Excellence in 2008, and the IEEE Electromagnetic Compatibility Society Best Symposium Paper Award in 2011.

**THIRD C. AUTHOR, JR.** (M’87) received the B.S. degree in mechanical engineering from National Chung Cheng University, Chiayi, Taiwan, in 2004 and the M.S. degree in mechanical engineering from National Tsing Hua University, Hsinchu, Taiwan, in 2006. He is currently pursuing the Ph.D. degree in mechanical engineering at Texas A&M University, College Station, TX, USA.

From 2008 to 2009, he was a Research Assistant with the Institute of Physics, Academia Sinica, Tapei, Taiwan. His research interest includes the development of surface processing and biological/medical treatment techniques using nonthermal atmospheric pressure plasmas, fundamental study of plasma sources, and fabrication of micro- or nanostructured surfaces.

Mr. Author’s awards and honors include the Frew Fellowship (Australian Academy of Science), the I. I. Rabi Prize (APS), the European Frequency and Time Forum Award, the Carl Zeiss Research Award, the William F. Meggers Award and the Adolph Lomb Medal (OSA).