EXPERIMENTAL PERFORMANCE EVALUATION OF WIREGUARD PROTOCOL COMPARED TO OPENVPN AND IPsec IN VIRTUAL PRIVATE NETWORKS

Pham Anh Thu
Posts and Telecommunications Institute of Technology

Abstract - In the rapidly evolving landscape of digital communication, Virtual Private Networks (VPNs) have become indispensable for ensuring secure and private connectivity over the Internet. This paper delves into the technical advancements and performance of modern VPN technologies, specifically focusing on the WireGuard protocol—a relatively new technology that has garnered significant attention due to its streamlined architecture and improved efficiency. Through comparative analysis, WireGuard is evaluated against traditional VPN solutions such as OpenVPN and IPsec across various parameters. The research methodology leverages simulation data obtained from labs using standardized tools, analyzing metrics such as bandwidth, latency, and the time of connection setup. The results demonstrate that WireGuard offers significant improvements in performance, thanks to its simple design and use of advanced encryption protocols.

Keywords – Virtual Private Network (VPN), WireGuard, OpenVPN, IPsec.

I. INTRODUCTION

In the context of rapidly advancing technologies and increasingly prevalent cybersecurity threats, Virtual Private Networks (VPNs) have become an essential solution for protecting data, ensuring privacy, and providing secure access for remote users. Two traditional protocols, OpenVPN and IPsec, have played significant roles in establishing VPN connections for many years [1].

OpenVPN is well known for its high flexibility and compatibility across various platforms. However, one of the major drawbacks of OpenVPN is its complexity in configuration and inconsistent performance, especially when operating on high-latency networks. The complex encryption structure of OpenVPN, while delivering a high level of security, increases connection setup time and consumes system resources, ultimately reducing system performance. Meanwhile, IPsec, a long-established VPN protocol, is often used in enterprise networks due to its robust security capabilities and widespread support. However, IPsec faces similar challenges, including long connection setup times, high resource demands, and complicated configurations. These drawbacks not only increase management costs but also pose difficulties in scaling and maintaining VPN systems within modern network environments.

Both protocols have certain drawbacks that prevent them from fully meeting the performance and flexibility requirements of modern networks. Specifically, issues such as limited bandwidth, high latency, and long connection setup time are major challenges that need to be addressed. WireGuard, a next-generation VPN protocol, has garnered significant attention for its ability to overcome these limitations [2-5].

In recent years, WireGuard is emerging as a promising next-generation VPN solution, offering significant advancements over traditional protocols like OpenVPN and IPsec. Its streamlined architecture and modern cryptographic protocols contribute to enhanced performance and security, making it a strong candidate for replacing older VPN technologies. WireGuard's integration into the Linux kernel further facilitates its adoption across various platforms, highlighting its potential as a robust and efficient VPN solution. Several studies on the application suitability of OpenVPN and IPsec in [6-7] have provided a comprehensive suite of security features but can be cumbersome to configure and manage. Data transfer speed and network latency are WireGuard's advantages. Due to its efficient codebase and cutting-edge cryptographic methods, it streamlines VPN processes and reduces overhead compared to OpenVPN and IPsec [8]. Empirical tests have demonstrated WireGuard's ability to outperform OpenVPN and other VPN software in NAT have been done by [9]. Security features such as message secrecy, forward secrecy, and mutual authentication are studied by [10].

Several recent studies have demonstrated that WireGuard outperforms traditional protocols in terms of performance, particularly in network environments with high demands for speed and reliability. Experimental reports indicate that WireGuard not only delivers higher data transfer speeds but also improves user experience with lower latency and stable connection maintenance, even in heterogeneous or noisy network conditions.

Contact author: Pham Anh Thu
Email: thupa@ptit.edu.vn
Manuscript received: 13/02/2025, revised: 24/03/2025, accepted: 31/3/2025.

Additionally, other studies emphasize that WireGuard is easier to deploy and manage compared to OpenVPN and IPsec [11-12]. WireGuard operates based on a "stateless" model, which optimizes packet processing, especially in environments requiring IP address switching or operating over mobile networks. This significantly mitigates common issues encountered with OpenVPN and IPsec, such as network switching failures or connection interruptions. With its concise codebase and performance-focused design, WireGuard not only enhances security but also facilitates seamless integration into modern network systems.

The need for simulation or experiment the WireGuard protocol in virtual private networks is essential to objectively compare and evaluate its performance against OpenVPN and IPsec. Criteria such as bandwidth, latency, and connection setup time must be accurately measured to determine whether WireGuard truly serves as a comprehensive alternative. Through this evaluation, organizations and businesses can identify the most suitable deployment approach to improve operational efficiency and ensure optimal security in existing VPN systems.

The remainder of the paper is organized as follows. The WireGuard protocol is introduced in Section 2. Section 3 presents the system simulation model. Section 4 discusses the simulation results and provides an evaluation of these findings. Finally, Section 5 concludes the paper.

II. WIREGUARD PROTOCOL

WireGuard is a simple and modern virtual private network security protocol, distinguished by its advanced encryption algorithms. It delivers high speed, ease of configuration, and superior performance compared to other options like IPsec and OpenVPN, proving its efficiency. As a cross-platform VPN protocol, WireGuard can operate on almost all major operating systems, including Linux, Windows, Android, and macOS. Notably, WireGuard functions on a peer-to-peer network model rather than the traditional client-server model. Depending on specific configurations, a peer device can act as either a client or a server.

WireGuard operates by creating a network interface on each peer device, functioning as a secure communication channel. Participants exchange and authenticate through a public key cryptography system, similar to the SSH model. Public keys are mapped to allowed IP addresses within the tunnel, and all VPN traffic is transmitted via the UDP protocol. WireGuard ensures data security by utilizing modern encryption algorithms and a simple design that minimizes security risks and optimizes performance.

WireGuard employs modern cryptographic algorithms such as Curve25519, ChaCha20, Poly1305, BLAKE2, SipHash24, and HKDF [13-14]. These algorithms have been reviewed and highly regarded by cryptographers for their reliability and security. This careful and reasonable selection helps WireGuard build a robust security structure that meets user information protection needs in modern network environments. Specifically, ChaCha20 is used for data encryption, and Poly1305 for message authentication. ChaCha20 is a very fast and secure symmetric encryption algorithm, while Poly1305 ensures data integrity, guaranteeing that messages remain unaltered during transmission. Additionally, WireGuard uses Blake2s for hashing and Curve25519 for public key exchange, ensuring the security of keys and resilience against modern attacks.

WireGuard not only supports cross-platform usage but is also easy to configure, deploy, and operate, as simple as SSH. Designed with the primary purpose of providing a simple, secure, and efficient VPN solution and protocol, WireGuard overcomes many limitations of popular VPN protocols like OpenVPN and IPsec. As a new VPN security protocol that is both simple and effective, WireGuard is built to secure and optimize data transmission over networks. WireGuard's connection models are based on a simple yet robust architecture, incorporating advanced encryption mechanisms and a fast handshake process, protecting transmitted data from potential threats while ensuring integrity, confidentiality, and authenticity of information in network environments. The main connection types that can apply the WireGuard security protocol are illustrated in Figure 1.

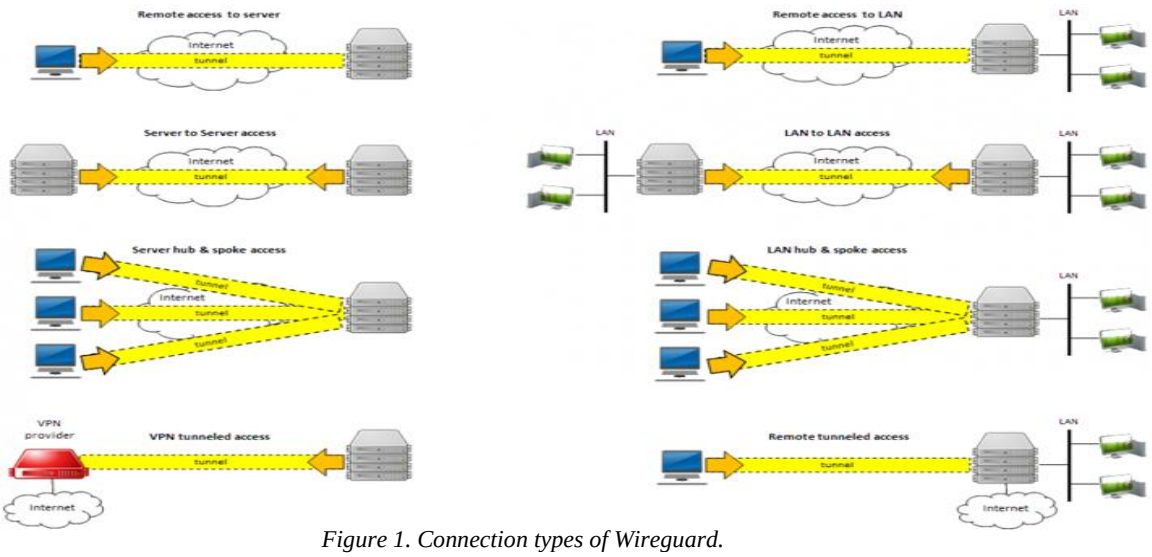

Figure 1. Connection types of Wireguard.

The two most common models are remote access to an internal network and internal network-to-internal network access. Remote access to an internal network is the most popular model for remote workers, especially with the growing demand for remote work. This model allows users to securely and safely connect to the internal network. It is particularly important for companies and organizations that need to protect data when employees access the corporate network from external environments.

The internal network-to-internal network model is widely used between company branches or offices that need to share network resources. This is achieved by creating a shared virtual private network (VPN) for the offices and branches. Using WireGuard for this model securely connects internal networks across different locations over the Internet.

WireGuard offers several outstanding advantages that make it an appealing choice for both individuals and businesses. One of its key benefits is simple configuration. It is easy to set up and use, making it accessible even for those without deep knowledge of network configurations. This simplicity extends to both individual users and organizations. In addition to its ease of use, WireGuard provides fast connection speeds. By utilizing fast encryption algorithms and integrating directly into the Linux kernel, it delivers superior speed compared to other protocols such as IPsec and OpenVPN. This makes it a highly efficient choice for VPN connections.

WireGuard also stands out for its high security. It employs modern cryptographic algorithms like ChaCha20 and Poly1305, along with secure default configurations. The protocol's compact codebase, which consists of around 4,000 lines, allows for easier auditing and reduces the risk of security vulnerabilities, providing a robust solution for secure connections. Another key feature of WireGuard is its "security through simplicity" philosophy. The protocol is designed to be simple, which allows users to quickly become familiar with it and easily manage it. This approach makes WireGuard suitable for a wide range of users, including those without specialized security expertise.

Easy deployment is another major advantage of WireGuard. It is available on multiple operating systems with ready-to-use client applications, making it easy to deploy quickly and efficiently. Even beginners with limited security experience can implement WireGuard without difficulty. Finally, WireGuard ensures secure key exchange by using the Noise_IK handshake process. This mechanism prevents impersonation attacks, replay attacks, and ensures the secure forwarding of traffic, making it a highly secure option for VPN connections.

Although WireGuard offers many advantages, there are also some drawbacks that should be considered. One such issue is suboptimal IP security. WireGuard stores the IP addresses of connections to the server indefinitely or until the server is restarted. This can impact user privacy because the IP addresses are not automatically deleted, unlike in other VPN security protocols where this happens more routinely.

Another limitation is that WireGuard only supports the UDP protocol. This reliance on UDP can create difficulties when connecting over networks that do not support UDP or when passing through firewalls and NAT devices that block this protocol. This can limit its usability in certain network environments. In terms of privacy, WireGuard prioritizes security, but it may not fully address privacy concerns. Those responsible for managing the server need to take additional steps to ensure user privacy, such as configuring the system to regularly delete IP address logs, which is not automatically handled by the protocol itself. Finally, WireGuard is still under development. While it is a promising solution, it is still being enhanced in terms of features and compatibility with various systems. As a result, some features are still being finalized, and it may not yet offer the full range of capabilities that users might expect in a mature protocol.

III. SYSTEM SIMULATION SCENARIO

In this section, the system simulation scenario, installation steps, and configuration details of WireGuard protocol will be discussed. Additionally, the simulation will also include a comparison to highlight the features of the WireGuard protocol in relation to OpenVPN and IPsec.

The simulation scenario is presented in Figure 2.

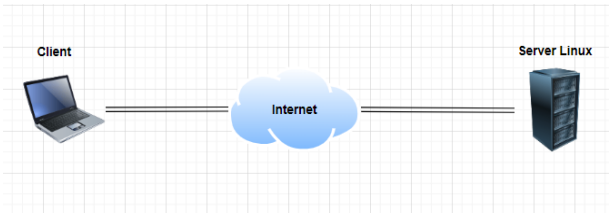

Figure 2. System simulation scenario.

The basic system scenario includes the components (as shown in Figure 2):

+ Client: The user device (which can be a laptop or computer) that connects to the server to send requests and receive data.
+ Internet: The intermediary network through which data is transmitted from the client to the server and vice versa.
+ Linux Server: The server running the Linux operating system, responsible for processing and providing data or services to the client.

Three labs will be implemented based on the model in Figure 2. Each lab will provide a different configuration method to establish a virtual private network (VPN) connection between the Client and Server. The implementation of these 3 labs will help compare the performance and configuration of different VPN protocols. Ultimately, it will highlight the superiority of the WireGuard security protocol.

Lab 1: Using WireGuard

The objective of this lab is to establish a fast and highly secure VPN connection between the Client and Server using the WireGuard protocol. By the end of the lab, you will have successfully set up a reliable and encrypted communication channel.

For the configuration, WireGuard needs to be installed on both the client and the server. The server will be configured to accept connections from the client,

utilizing public keys for authentication. The client will then use its private key along with the server's public key to initiate and establish the secure connection.

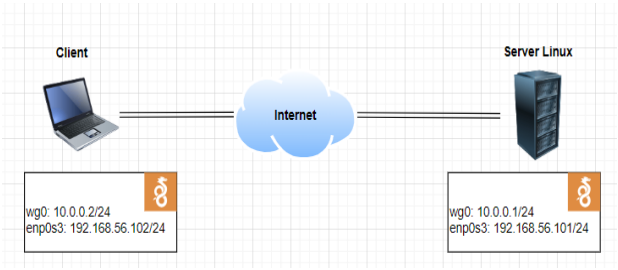

Figure 3. Simulation system using Wireguard

Lab 2: Using OpenVPN

The objective of this lab is to establish a secure VPN connection between the Client and Server using OpenVPN. This setup will ensure encrypted communication between the two endpoints.

For the configuration, both the client and server will have OpenVPN installed. The server will function as an OpenVPN server, listening for incoming connections from the client over the Internet. The client will configure its OpenVPN client to connect to the server, using certificates and encryption keys for authentication and secure communication.

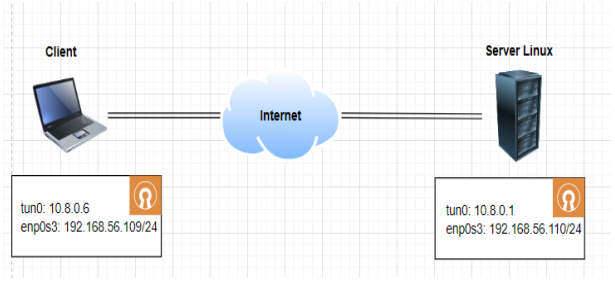

Figure 4. Simulation system using OpenVPN

Lab 3: Using IPsec VPN

The objective of this lab is to create a secure encrypted tunnel between the Client and Server using IPsec VPN. This setup will ensure that all communication between the two endpoints is securely transmitted.

For the configuration, the server will be set up to function as an IPsec server, with parameters such as AH (Authentication Header) and ESP (Encapsulating Security Payload) configured. The client will need to configure its system accordingly to establish a connection with the IPsec server, ensuring secure data transfer across the network.

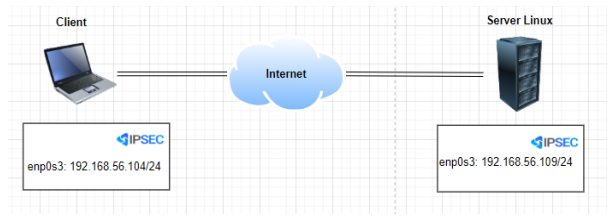

Figure 5. Simulation system using Ipsec VPN

VirtualBox is used as the environment to create virtual machines, supporting the execution of the lab tasks. VirtualBox is an open-source virtualization software developed by Oracle that allows users to create and manage virtual machines on their computers. It supports multiple operating systems such as Windows, macOS, Linux, and Solaris, enabling users to run various operating systems on a single computer without needing to reinstall. In these simulation lab tasks, two Ubuntu virtual machines will be used on VirtualBox: one will act as the client, and the other will serve as the server.

IV. SIMULATION RESULTS

In this section, the system performance, including bandwidth, latency, and connection setup time, will be investigated.

A. Bandwidth Measurement

To test bandwidth and throughput between virtual machines in each lab using the WireGuard, OpenVPN, or IPsec protocols, the iperf3 tool must first be installed on the virtual machines. This tool is a popular network performance measurement tool that includes data transfer speed and latency.

Once the installation and setup are completed, the testing process will be conducted by sending data through each VPN protocol, such as WireGuard, OpenVPN, and IPsec, to measure bandwidth, latency, and throughput under real-world conditions. These results will help evaluate the performance and suitability of each protocol in different network environments. The measurement results for each lab are as follows:

Lab 1: Using WireGuard

```
server@server-VirtualBox:~$ iperf3 -c 10.0.0.2
Connecting to host 10.0.0.2, port 5201
[  4] local 10.0.0.1 port 41142 connected to 10.0.0.2 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.00   sec   134 MBytes  1.12 Gbits/sec   62    398 KBytes
[  4]   1.00-2.00   sec   130 MBytes  1.09 Gbits/sec   34    279 KBytes
[  4]   2.00-3.00   sec   129 MBytes  1.08 Gbits/sec    0    399 KBytes
[  4]   3.00-4.00   sec   133 MBytes  1.12 Gbits/sec    8    385 KBytes
[  4]   4.00-5.00   sec   136 MBytes  1.14 Gbits/sec    0    426 KBytes
[  4]   5.00-6.00   sec   138 MBytes  1.16 Gbits/sec   23    367 KBytes
[  4]   6.00-7.00   sec   123 MBytes  1.03 Gbits/sec   46    386 KBytes
[  4]   7.00-8.00   sec   135 MBytes  1.13 Gbits/sec   20    327 KBytes
[  4]   8.00-9.00   sec   129 MBytes  1.09 Gbits/sec   10    350 KBytes
[  4]   9.00-10.00  sec   135 MBytes  1.13 Gbits/sec    0    434 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec  1.29 GBytes  1.11 Gbits/sec  203             sender
[  4]   0.00-10.00  sec  1.29 GBytes  1.11 Gbits/sec                  receiver
```

Figure 6. Bandwidth in WireGuard lab

Lab 2: Using OpenVPN

```
client@client-VirtualBox:~$ iperf3 -c 10.8.0.1
Connecting to host 10.8.0.1, port 5201
[  4] local 10.8.0.6 port 47170 connected to 10.8.0.1 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.00   sec  27.2 MBytes   228 Mbits/sec   13    110 KBytes
[  4]   1.00-2.00   sec  27.2 MBytes   228 Mbits/sec   18    106 KBytes
[  4]   2.00-3.00   sec  28.9 MBytes   243 Mbits/sec   22    116 KBytes
[  4]   3.00-4.00   sec  29.9 MBytes   251 Mbits/sec   26   90.6 KBytes
[  4]   4.00-5.00   sec  29.4 MBytes   247 Mbits/sec   11    117 KBytes
[  4]   5.00-6.00   sec  30.2 MBytes   253 Mbits/sec   18   99.8 KBytes
[  4]   6.00-7.00   sec  28.8 MBytes   242 Mbits/sec    9    114 KBytes
[  4]   7.00-8.00   sec  28.6 MBytes   240 Mbits/sec   20   82.7 KBytes
[  4]   8.00-9.00   sec  28.9 MBytes   243 Mbits/sec   19   93.3 KBytes
[  4]   9.00-10.00  sec  22.2 MBytes   186 Mbits/sec   10   95.9 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec   281 MBytes   236 Mbits/sec  166             sender
[  4]   0.00-10.00  sec   281 MBytes   236 Mbits/sec                  receiver

iperf Done.
```

Figure7. Bandwidth in OpenVPN lab

Lab 3: Using IPsec VPN

```
server2@server2-VirtualBox:~$ iperf3 -c 192.168.56.104
Connecting to host 192.168.56.104, port 5201
[  4] local 192.168.56.103 port 45782 connected to 192.168.56.104 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.02   sec  25.2 MBytes   208 Mbits/sec    0    110 KBytes
[  4]   1.02-2.00   sec  23.8 MBytes   202 Mbits/sec    0    110 KBytes
[  4]   2.00-3.02   sec  24.7 MBytes   204 Mbits/sec    0    127 KBytes
[  4]   3.02-4.02   sec  24.8 MBytes   208 Mbits/sec    0    156 KBytes
[  4]   4.02-5.00   sec  46.4 MBytes   396 Mbits/sec    0   1.07 MBytes
[  4]   5.00-6.00   sec  36.4 MBytes   306 Mbits/sec  818   1.05 MBytes
[  4]   6.00-7.00   sec  31.6 MBytes   265 Mbits/sec    0   1.15 MBytes
[  4]   7.00-8.00   sec  32.5 MBytes   273 Mbits/sec    0   1.23 MBytes
[  4]   8.00-9.00   sec  25.3 MBytes   212 Mbits/sec    0   1.29 MBytes
[  4]   9.00-10.00  sec  31.3 MBytes   263 Mbits/sec    2    968 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-10.00  sec   302 MBytes   253 Mbits/sec  820             sender
[  4]   0.00-10.00  sec   301 MBytes   252 Mbits/sec                  receiver
```

Figure 8. Bandwidth in IPsec VPN lab

The results from Figures 6 to 8 show that: WireGuard has a bandwidth of aproximately 1.11 Gbits/sec, while OpenVPN has a bandwidth of 236 Mbits/sec and IPsec has a bandwidth of 253 Mbits/sec. This demonstrates that WireGuard has significantly superior bandwidth compared to the other two protocols.

B. Measuring Average Latency

We used the ping command to check and evaluate the latency between the virtual machines when connecting through each VPN protocol in each lab. The ping command sends ICMP packets to the IP address of the opposite machine and measures the response time, which allows for the determination of the average, minimum, and maximum latency.

```
--- 10.0.0.1 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 101379ms
rtt min/avg/max/mdev = 0.267/0.433/0.894/0.088 ms
```

a, Average latency in WireGuard lab

```
--- 10.8.0.1 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 101271ms
rtt min/avg/max/mdev = 0.308/0.500/1.448/0.178 ms
```

b, Average latency in OpenVPN lab

```
--- 192.168.56.103 ping statistics ---
100 packets transmitted, 100 received, 0% packet loss, time 101263ms
rtt min/avg/max/mdev = 0.246/0.518/7.447/0.854 ms
```

c, Average latency in IPsec VPN lab

Figure 9. Average latency

The results from figure 9 show that WireGuard has an average latency of 0.433ms, while OpenVPN has an average latency of 0.500ms and IPsec has an average latency of 0.518ms. Therefore, WireGuard has lower latency compared to OpenVPN and IPsec.

C. Measuring Connection Setup Time

The connection setup time for the three models was measured, and the results are shown in Figure 10.

The results in Figure 10 show that WireGuard has a connection setup time of 1.776s, while OpenVPN has a connection setup time of 3.229s and IPsec has a connection setup time of 2.407s. Therefore, WireGuard has a shorter connection setup time compared to OpenVPN and IPsec.

```
real    0m1.776s
user    0m0.014s
sys     0m0.000s
```

a , Connection setup time for WireGuard lab

```
real    0m3.229s
user    0m0.007s
sys     0m0.006s
```

b, Connection setup time for OpenVPN lab

```
real    0m2.407s
user    0m0.010s
sys     0m0.002s
```

c, Connection setup time for IPsec VPN lab

Figure 10. Connection setup time

Table 1 presents the measurement results of the performance parameters of the three VPN protocols, including WireGuard, OpenVPN, and IPsec. The metrics measured include bandwidth, average latency, and connection setup time, which help compare the performance of each protocol under the same environmental conditions. Based on these results, we can assess the performance of each protocol and provide insights into the factors influencing the choice of a suitable VPN protocol for different usage needs.

Table 1. Performance parameters of the three VPN protocols

Protocol	Bandwidth	Average latency	Connection setup time
WireGuard	1.11 Gbits/sec	0.433 ms	1.776s
OpenVPN	236 Mbits/sec	0.500 ms	3.229s
IPsec	253 Mbits/sec	0.518 ms	2.407s

From table 1, it is can be seen that WireGuard shows exceptional performance with a bandwidth of 1.11 Gbits/sec, vastly surpassing OpenVPN (236 Mbits/sec) and IPsec (253 Mbits/sec). This demonstrates WireGuard's powerful data handling capabilities, optimizing data transfer speeds. The significant difference in bandwidth can be attributed to WireGuard's modern, simplified design that focuses on performance.

WireGuard exhibits the lowest average latency at just 0.433 ms, faster than OpenVPN (0.500 ms) and IPsec (0.518 ms). This is a critical factor for applications that require low latency, such as video streaming, online meetings, or real-time services. WireGuard’s low latency reflects its lightweight and optimized packet processing structure.

WireGuard continues to outperform with the fastest connection setup time of 1.776s, showcasing its streamlined process for initiating sessions. IPsec ranks second with 2.407s, while OpenVPN takes as long as 3.229s, nearly twice the time of WireGuard. The longer setup time for OpenVPN can be attributed to its more complex architecture, which requires additional authentication steps.

The hands-on lab tests further confirm that WireGuard is not only faster but also easier to configure compared to OpenVPN and IPsec. Traditional protocols like OpenVPN and IPsec require multiple complex setup steps, whereas WireGuard simplifies this process with a clean and easy-to-use syntax. This not only saves time but also reduces the potential for errors during implementation.

WireGuard excels not only in performance metrics (bandwidth, latency, setup time) but also offers a user-friendly configuration experience, making it a perfect fit for modern systems that demand high speed and stability.

V. CONCLUSIONS

In the era of continuously evolving digital communication, choosing an optimal Virtual Private Network (VPN) solution plays a crucial role in protecting data and ensuring secure connections. In this paper, a detailed analysis and performance comparison between the WireGuard protocol and traditional VPN protocols like OpenVPN and Ipsec have been conducted. The simulation results show that WireGuard outperforms in terms of bandwidth, lower latency, and faster connection setup time, thanks to its lightweight architecture and advanced encryption algorithms.

WireGuard's improvements not only address the limitations of OpenVPN and IPsec but also open up new opportunities for implementing VPNs in modern network environments, where performance and simplicity are increasingly prioritized. Moreover, WireGuard's concise codebase helps minimize potential security vulnerabilities and allows for easy integration into existing systems, further increasing its value as a comprehensive alternative solution.

However, the real-world deployment of Wireguard presents several limitations. These include challenges with complex configurations, as it lacks advanced features found in other VPN protocols, and key management difficulties, particularly in dynamic or large-scale environments. Additionally, its reliance on static keys for authentication may not support sophisticated authentication methods, and compatibility issues with legacy systems or hardware can arise. Performance may also degrade under heavy loads or fluctuating network conditions, and there may be difficulties with firewall and NAT traversal. Furthermore, scalability in large networks can be a concern, requiring additional infrastructure or customization. These factors highlight the need for careful consideration when deploying WireGuard outside of a lab setting.

REFERENCES

[1] K. Ghanem, S. Ugwuanyi, J. Hansawangkit, R. McPherson, R. Khan and J. Irvine, "Security vs Bandwidth: Performance Analysis Between IPsec and OpenVPN in Smart Grid", *2022 International Symposium on Networks Computers and Communications (ISNCC)*, pp. 1-5, 2022.

[2] J. A. Donenfeld, "Wireguard: Next Generation Kernel Network Tunnel", June 1, 2020.

[3] S. Taylor, "WireGuard VPN: Secure and Fast, But Bad for Privacy?," Restore Privacy, Sep. 05, 2023.

[4] Z. Xu and J. Ni, "Research on network security of VPN technology", *2020 International Conference on Information Science and Education (ICISE-IE)*, pp. 539-542, 2020.

[5] P.N. Phan Hai, H. Nguyen Hong, B.B. Quoc and T. Hoang, "A Comparative Research on VPN Technologies on Operating System for Routers", *2021 International Conference on Advanced Technologies for Communications (ATC)*, pp. 89-93, 2021.

[6] Parenreng, Jumadi Mabe. "Network security analysis based on internet protocol security using virtual private network (VPN)." IOTA Journal 3, no. 3 (2023): 239-249.

[7] Davie, Bruce S., and Adrian Farrel. "Virtual Private Networks," 351–69. Morgan Kaufmann, 2008. https://doi.org/10.1016/B978-0-12-374400-5.00012-6.

[8] Остроух, А. В., Cesar Borisovich Pronin, А. А. Podberezkin, J. V. Podberezkina, et A. M. Volkov. 2024. « Enhancing Corporate Network Security and Performance: A Comprehensive Evaluation of WireGuard as a Next-Generation VPN Solution », juillet, 1-5. https://doi.org/10.1109/synchroinfo61835.2024.10617501.

[9] Choon Hoe Chua and S. C. Ng, "Open-Source VPN Software: Performance Comparison for Remote Access," August 26, 2022, https://doi.org/10.1145/3561877.3561882.

[10] Lipp, Benjamin, Bruno Blanchet, and Karthikeyan Bhargavan. "A Mechanised Cryptographic Proof of the WireGuard Virtual Private Network Protocol." IEEE European Symposium on Security and Privacy, June 17, 2019, 231–46. https://doi.org/10.1109/EUROSP.2019.00026.

[11] K. -F. Krentz and M. -I. Corici, "Poster: Multipath Extensions for WireGuard", *2021 IFIP Networking Conference (IFIP Networking)*, pp. 1-3, 2021.

[12] Prof. Dr. Tanja Lange, Jacob Appelbaum, Jason A. Donenfeld, "Analysis of the WireGuard protocol," thesis at Eindhoven University of Technology, June 17, 2019.

[13] Andrew He, Baula Xu, Jerrry Wu, "Security Analysis of WireGuard", MIT 6.857 Project, Spring 2018.

[14] B. Lipp, B. Blanchet and K. Bhargavan, "A Mechanised Cryptographic Proof of the WireGuard Virtual Private Network Protocol," *2019 IEEE European Symposium on Security and Privacy (EuroS&P)*, Stockholm, Sweden, 2019, pp. 231-246, doi: 10.1109/EuroSP.2019.00026..

ĐÁNH GIÁ HIỆU NĂNG THỬ NGHIỆM CỦA GIAO THỨC WIREGUARD SO VỚI OPENVPN VÀ IPSEC TRONG MẠNG RIÊNG ẢO

Tóm tắt – Trong bối cảnh truyền thông số đang phát triển nhanh chóng, mạng riêng ảo (VPN) đã trở nên không thể thiếu để đảm bảo kết nối qua mạng Internet một cách an toàn và riêng tư. Bài báo này đi sâu vào những tiến bộ kỹ thuật và hiệu quả của các công nghệ VPN hiện đại, cụ thể là giao thức WireGuard, một công nghệ tương đối mới đã thu hút được sự chú ý đáng kể nhờ kiến trúc hợp lý và hiệu năng được cải thiện. Thông qua phân tích so sánh, WireGuard được đánh giá so với các giải pháp VPN truyền thống như OpenVPN và IPsec, trên nhiều khía cạnh. Phương pháp nghiên cứu trong bài báo sử dụng dữ liệu mô phỏng thu được từ các bài lab sử dụng các công cụ chuẩn hóa, phân tích các số liệu như băng thông, độ trễ, thời gian thiết lập kết nối. Các kết quả cho thấy WireGuard cung cấp những cải tiến đáng kể về các tham số hiệu năng, nhờ sử dụng thiết kế đơn giản và các giao thức mã hóa tiên tiến.

Từ khóa – Mạng riêng ảo (VPN), Wireguard, OpenVPN, IPsec.

Pham Anh Thu received her Bachelor's degree in Telecommunications Engineering from the Posts and Telecommunications Institute of Technology (PTIT), Vietnam, in 2003, and her Master's degree in Telecommunications Engineering from the Royal Melbourne Institute of Technology, Australia, in 2008. She obtained her Ph.D. in Telecommunications Engineering from PTIT in 2019. Currently, she is a lecturer at the Telecommunications Department, Posts and Telecommunications Institute of Technology. Her main research areas include communication networks, radio wave transmission over fiber optics, and information network security.