라즈베리파이를 활용한 개인용 VPN 서버 구축 및 성능 분석

이정연[1], 전상훈[2*]
[1]수원대학교 정보보호학과 학생, [2]수원대학교 정보보호학과 교수

Implementation and Performance Analysis of a Personal VPN Server Using Raspberry Pi

Jung-Yeon Lee[1], Sanghoon Jeon[2*]
[1]Student, Department of Information and Security, The University of Suwon
[2]Professor, Department of Information and Security, The University of Suwon

요 약 VPN(Virtual Private Network)은 네트워크 보안 기술 중 하나로, 공공 네트워크에서의 익명성을 보장하거나 개인 네트워크 환경에 접근하기 위해 사용한다. 현재 널리 사용되는 프로토콜은 OpenVPN과 WireGuard가 있으며, 두 프로토콜은 성능과 보안성 측면에서 각각 다른 특징을 가진다. 본 논문에서는 OpenVPN과 WireGuard의 성능 및 보안성을 비교·분석하여, 사용자가 환경에 따라 최적의 VPN 프로토콜을 선택할 수 있는 기준을 제시하는 것을 목표로 한다. 이를 위해 VPN 사용 시 발생하는 송수신 속도와 지연 시간, CPU 사용률, 패킷 손실을 기준으로 외부망과 내부망의 성능을 측정하고, 구성 파일이 유출되었을 때 발생할 수 있는 보안 문제를 분석하여 보안성을 평가하였다. 실험 결과, 통신 및 처리 성능 측면에서는 WireGuard가 우수하였으며, 보안성 측면에서는 OpenVPN이 더 안전하였다. 이를 바탕으로 본 연구는 사용자에게 맞는 VPN 프로토콜 선택의 방향성을 제안하고 VPN 환경에서 구성 파일의 중요성을 강조한다.

주제어 : 가상 사설망, VPN 보안, 오픈VPN, 와이어가드, 라즈베리파이

Abstract A Virtual Private Network (VPN) is a security technology used to ensure anonymity on public networks or securely access private network environments. Currently, the most widely used VPN protocols are OpenVPN and WireGuard, each of which has distinct characteristics in terms of performance and security. This study aims to compare and analyze the performance and security of OpenVPN and WireGuard to provide users with criteria for selecting the optimal VPN protocol based on their network environment. We evaluate the performance of both protocols in external and internal network environments based on upload speed, download speed, latency, CPU usage, and packet loss rate. Additionally, we analyze security vulnerabilities in cases where configuration files were leaked, emphasizing the importance of securing configuration files. Experimental results show that WireGuard outperforms OpenVPN in terms of communication and computing performance, while OpenVPN provides stronger security, particularly in terms of protecting anonymity and preventing traffic analysis. Based on these findings, this study guides selecting the appropriate VPN protocol according to user needs and highlights the importance of configuration file security in VPN environments.

Key Words : VPN, VPN Security, OpenVPN, WireGuard, Raspberry Pi

본 연구는 원자력안전위원회의 재원으로 한국원자력안전재단의 지원을 받아 수행한 원자력안전연구사업의 연구결과입니다 (No. RS-2024-00403596).
*교신저자 : 전상훈(shjeon@suwon.ac.kr)
접수일: 2025년 03월 08일 수정일: 2025년 03월 28일 심사완료일: 2025년 04월 09일

1. 서론

최근 인터넷 서비스의 발달로 다양한 디지털 서비스가 등장하며 무선 네트워크(Wireless Network)를 통한 다양한 정보가 공유되고 있다. 무선 네트워크는 Wi-Fi (Wireless Fidelity)와 같이 거의 모든 환경에서 시간과 공간의 제약 없이 자유롭게 정보를 송수신할 수 있는 기술이다. 그러나 무선 네트워크는 데이터를 무선으로 송수신하기 때문에 공격자가 신호를 가로챌 수 있으며, 이로 인해 심각한 보안 위협을 초래할 수 있다. 만약 공격자와 피해자가 같은 통신 영역에 위치할 경우, 공격자는 내부 사용자와 같은 권한으로 공격을 실행할 수 있다. 공격에 성공하면, 공격자는 저장된 데이터를 탈취당하거나 암호화시킨 후 사용자에게 금전을 요구하는 등의 피해를 줄 수 있다. 따라서 이러한 보안 위협을 방지하기 위한 네트워크 보안 기술의 적용은 매우 중요하다[1-3].

네트워크 보안 기술의 대표적인 예로 가상 사설 네트워크(VPN, Virtual Private Network)가 있다. VPN은 네트워크를 통해 단말기와 VPN 서버 사이의 통신을 암호화하여 터널을 형성하는 기술이다. 이러한 보안 터널 형성으로 공격자로부터 네트워크 트래픽이 보호된다. 따라서 공용 네트워크 환경에서도 공격자가 신호를 가로챌 수 없으므로 보안을 강화하는 예방 방법이 될 수 있다[4]. VPN을 사용하는 프로토콜은 L2PT(Layer 2 Tunneling Protocol), PPTP(Point-to-Point Tunneling Protocol), IPSec(Internet Protocol Security), SSL(Secure Sockets Layer) 등이 있으며, 2023년 기준, 소비자 대상으로 널리 사용되는 VPN 프로토콜이자 소프트웨어는 OpenVPN과 WireGuard가 있다[5]. VPN은 프로토콜에 따라 성능과 보안성에서 차이가 나타난다.

본 연구의 목적은 OpenVPN과 WireGuard의 성능 및 보안성을 종합적으로 비교하여, 사용 환경에 따라 둘 중 더 적합한 프로토콜을 선택할 수 있는 기준을 제시하는 것이다.

본 논문의 구조는 다음과 같다. 2장에서는 VPN 프로토콜 비교를 위한 성능과 보안성 관련 기술들의 배경적 지식을 설명한다. 3장에서는 라즈베리파이를 활용한 개인용 VPN 서버 구축 및 실험 환경을 설명한다. 4장에서는 실험을 통해 도출된 결과를 분석하여 두 프로토콜의 성능과 보안성을 비교한다. 마지막으로, 5장에서는 본 연구의 결론과 향후 연구 방향을 논의한다.

2. 관련 연구

일반적으로 클라이언트(Client)는 네트워크와 직접적으로 연결되지만, VPN을 사용할 경우, 암호화된 데이터가 VPN 터널을 통해 VPN 서버로 전송된다. 따라서 Client가 아닌 VPN 서버와 네트워크 간의 통신이 이루어지며 네트워크상의 외부 사용자는 VPN 뒤에 있는 Client에게 직접 접근할 수 없다.

본 절은 OpenVPN과 WireGuard의 기술적 특징을 설명하고, 두 프로토콜 간의 성능 및 보안성 분석을 위한 이론적 배경을 제공한다.

2.1 OpenVPN

OpenVPN[6,7]은 SSL/TLS 기반의 오픈 소스 가상 사설망(VPN) 프로토콜이다. 네트워크를 안전하게 확장하고 Client 인증 기능을 지원한다. 또한 SSL 라이브러리를 활용하여 개인 네트워크 트래픽을 보호하며 SSL/TLS를 통해 세션을 인증하고 TLS 프로토콜을 통해 키를 교환한다. OpenVPN은 UDP(User Datagram Protocol)와 TCP(Transmission Control Protocol)를 모두 지원하며, 다양한 운영체제(예: Linux, Windows, iOS)에서 사용할 수 있어 범용성이 높고, 사용자가 간단한 명령으로 터널을 생성할 수 있어 편리하다. 또한 OpenVPN은 IP 패킷을 안전하게 캡슐화하여 트래픽을 전달하는 구조를 갖는다.

OpenVPN은 데이터를 AES(Advanced Encryption Standard) 방식으로 암호화한다. AES란 미국 정부 표준으로 지정된 대칭키 암호 방식으로, 암호화 키는 128(AES-128), 192(AES-192), 256(AES-256)비트 중 하나를 사용하여 데이터를 암호화한다. AES는 암호화와 복호화 과정에서 같은 키를 사용하며, 복호화 과정에서 상대적으로 많은 연산이 필요하므로 소요 시간이 길다[8,9]. 그러나, AES는 강력한 보안성과 안정성을 제공하여 데이터를 안전하게 보호한다. 사용자의 설정에 따라 AES 대신 ChaCha20과 같은 다른 암호화 방식을 사용할 수도 있다[8-10].

2.2 WireGuard

WireGuard[11]란 사용자가 쉽게 구현할 수 있고 범용 VPN이므로, 다양한 환경에서 사용할 수 있다. OpenVPN과 다르게 TCP를 지원하지 않고 UDP만 사용한다. UDP를 통해 IP 패킷을 안전하게 캡슐화하여 트래픽을 전달한다. 또한 WireGuard는 공개 키 기반 인증

방식을 사용하여 클라이언트와 서버 간 패킷을 교환한다. 이 방식은 구성 파일에서 각 피어가 서로를 쉽게 인증할 수 있도록 설계되었으며, 배포와 관리가 간편하다는 장점이 있다. 패킷 교환 동작을 설명하면, 패킷 전송 과정에서 인터페이스는 공개 키를 통해 IP 패킷을 암호화한 후 UDP를 사용하여 암호화된 바이트를 전송한다. 패킷을 수신할 때는 복호화 및 인증을 수행한 후, 정상적인 패킷만을 수락하고 나머지는 폐기한다.

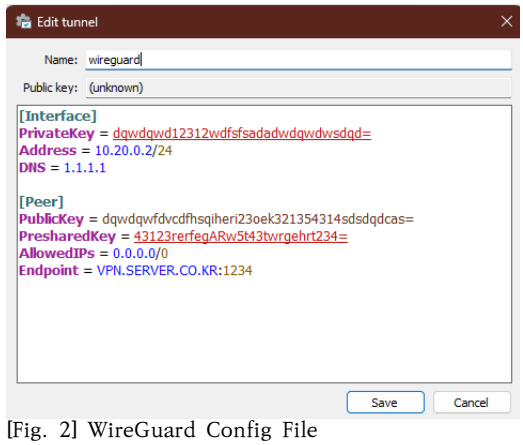
WireGuard는 ChaCha20[12]의 프로토콜을 사용하여 데이터를 암호화한다. 이 프로토콜은 대칭 키 스트림 암호로 기존 Salsa20 알고리즘을 개선한 형태이며, 현재 알려진 보안 결함이 존재하지 않기 때문에 보안성이 높다[10]. WireGuard는 256비트 키와 96비트 랜덤 생성 토큰(nonce)을 사용하여 보안을 강화하며, 소프트웨어 최적화에 특화된 고속 암호화 알고리즘으로 동작한다.

2.3 구성 파일(Config File)

구성 파일은 VPN 소프트웨어의 옵션 설정 정보가 들어있는 파일이다. 이 파일은 서버의 구성 정보, 서버의 보안 키, 클라이언트의 보안 키, 암호화 방식 등 여러 정보가 들어가 있다. 만약 구성 파일이 노출되면 외부에서 익명의 사용자나 공격자가 접근할 수 있고, 서버 접속을 위한 인증서를 가져가서 기업이나 개인의 사설망을 통해 서버, NAS, 프린터 등 여러 네트워크 기기에 접근할 수 있다는 문제가 발생한다. Fig. 1은 OepnVPN 구성 파일(.ovpn)을 나타내며, Fig. 2는 Wireguard 구성 파일(.conf)을 나타낸다.

```
client
dev tun
proto udp
remote MY.SERVER.CO.KR 1194
resolv-retry infinite
nobind
remote-cert-tls server
tls-version-min 1.2
verify-x509-name yoo10035_839498ef-4b59-4bc8-9877-7623110fff11 name
cipher AES-256-CBC
auth SHA256
auth-nocache
verb 3
<ca>
-----BEGIN CERTIFICATE-----
MIIBvzCCAWWgAwIBAgIUNe7SwE7ikNuxBLzGgJ8nrSIvv30wCgYIKoZIzj0EAwIw
FjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0EwHhcNMjUwMjE1MTMwMDAyWhcNMzUwMjEz
MTMwMDAyWjAWMRQwEgYDVQQDDAtFYXN5LVJTQSBDQTBZMBMGByqGSM49AgEGCCqG
SM49AwEHA0IABCvrL7uLt+jjWOFyhuU3ozYxQJxKAuhm/WjXYgFPAMM0jqAKdt+n
r9xBt4FbBEi5yfLdSgeH3AGQPOc+Qd/JBXujgZAwgY0wDAYDVR0TBAUwAwEB/zAd
BgNVHQ4EFgQUe8y6SotGULPAaK3BGGzEU3rg70EwUQYDVR0jBEowSIAUe8y6SotG
ULPAaK3BGGzEU3rg70GhGqQYMBYxFDASBgNVBAMMC0Vhc3ktU1NBIENBghQl7tLA
TuKQ27EEvMaAnyetIi+/fTALBgNVHQ8EBAMCAQYwCgYIKoZIzj0EAwIDSAAwRQIh
AOcRpaoRuLNQZp2OPChAz5EkKdDBKACVx/McaQKU1E7CAiApdVM6f3vY6prnjYSx
YHv15ZGjkzxvlW6AXc7jcDbKVw==
-----END CERTIFICATE-----
</ca>
<cert>
-----BEGIN CERTIFICATE-----
MIIBxzCCAW2gAwIBAgIQGVpFuGiSoUqa8DVZ9OqYDjAKBggqhkjOPQQDAjAWMRQw
EgYDVQQDDAtFYXN5LVJTQSBDQTAeFw0yNTAyMTUxMzAxMzZaFw0yODAxMzExMzAx
MzZaMBkxDjAMBgNVBAMMBXRlc3QxMFkwEwYHKoZIzj0CAQYIKoZIzj0DAQcDQgAE
nrh5zoo6g8jeJskaOy7GtmoHhrA7uyPC7wbjVqcl+qm+V1UcrisB42uB0h85osWg
nGZ954cRIQ5HN9Up8JaBIaOBojCBnzAJBgNVHRMEAjAAMB0GA1UdDgQWBBSIn4JX
PXEyIOJGmmnHcdWUz+PKOzBRBgNVHSMESjBIgBR7zLpKi0ZQs8BorcEYbMRTeuDv
QaEapBgwFjEUMBIGA1UEAwwLRWFzeS1SU0EgQ0GCFDXu0sBO4pDbsQS8xoCfJ60i
L799MBMGA1UdJQQMMAoGCCsGAQUFBwMCMAsGA1UdDwQEAwIHgDAKBggqhkjOPQQD
```

[Fig. 1] OpenVPN Config File



[Fig. 2] WireGuard Config File

2.4 VPN 성능 측정 평가 지표

본 논문에서 성능을 평가하기 위해 송신 속도(Upload Speed), 수신 속도(Download Speed), Ping(Packet Internet Groper), CPU 사용률, 패킷 손실률(Packet Loss Rate), 지연 시간(Latency)을 측정한다. 먼저, 송신은 통신 신호를 보내는 것이다. 따라서 송신 속도는 데이터를 서버에 저장하지 않고 얼마나 빨리 전송할 수 있는지를 의미한다. 수신은 송신과 반대로 통신 신호를 받는 것으로 수신 속도는 서버 안에 저장된 데이터를 얼마나 빨리 받을 수 있는지를 의미한다. 이때 속도는 bps(bit per second) 단위로 측정된다. bps란 1초에 몇 비트를 송신 및 수신할 수 있는지를 나타낸다. 속도는 Kbps(Kilo bits per second), Mbps(Megabit per second), Gbps(Gigabit per second)가 있다. 인터넷 속도의 표준 측정값은 Mbps이며, 송신 및 수신 속도는 모두 Mbps로 측정된다. 다음으로, 지연 시간은 네트워크 응답 속도를 나타내는 지표로, 데이터를 송수신하는데 걸리는 시간을 의미한다. 마지막으로, 패킷은 네트워크에서 전송되는 데이터의 용량 단위이다. 패킷 손실률은 전송 과정에서 용량 초과 등으로 수신되지 못하고 손실된 패킷의 비율을 의미한다.

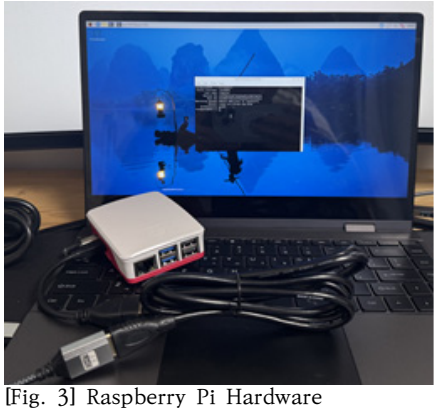
3. 제안 방법

3.1 하드웨어 구축 (Raspberry Pi 5)

본 논문에서는 OpenVPN과 WireGuard 기반의 개인용 VPN 서버를 저사양·저전력 환경에서 구축하고, 사용자의 관점에서 어떤 VPN을 선택하는 것이 좋을지 성능과 보안성을 평가한다. 이를 위해 VPN 하드웨어로

Raspberry Pi 5를 사용하였다. Fig. 3은 본 연구에서 활용하기 위해 Raspberry Pi OS를 설치한 환경을 나타낸다.

Raspberry Pi는 주변 기기와의 연결이 자유로운 소형 컴퓨터 장치로 저비용, 저사양, 저전력이라는 특징으로 개발자와 일반 사용자가 다양한 분야 및 프로젝트에 활용할 수 있다. GPIO(General Purpose Input Output) 핀과 같은 확장성, USB, 블루투스 통해 센서 및 다양한 주변 기기와 연결할 수 있으며 IoT, 개인 NAS 서버, 네트워크 보안 장치, 교육용 도구 등 다방면으로 사용할 수 있다. 특히, Raspberry Pi는 오픈소스 소프트웨어 지원과 강력한 커뮤니티 생태계를 기반으로 많은 사용자가 VPN 서버 구축에 활용하고 있다[13].

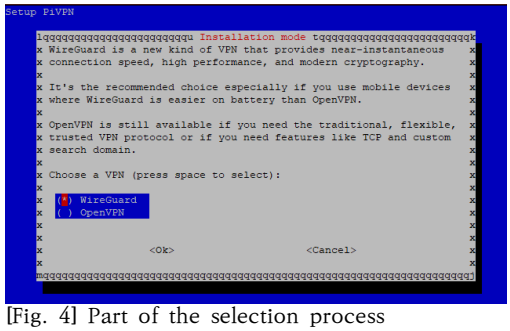


[Fig. 3] Raspberry Pi Hardware

3.2 소프트웨어 구축

본 연구에서는 Raspberry Pi 5를 기반으로 OpenVPN과 WireGuard를 설치하여 VPN 환경을 구축하였다. 구축 과정에서 PiVPN 패키지를 사용하였다. PiVPN은 Raspberry Pi에서 OpenVPN과 WireGuard를 간편하게 설치하고 관리할 수 있도록 설계된 스크립트 기반 패키지로, 명령어 기반의 설치 과정과 GUI 기반의 설정 메뉴를 제공한다. VPN 서버 구축 과정은 다음과 같다. 먼저, PiVPN 설치 스크립트를 실행하여 OpenVPN과 WireGuard 중 설치하고자 하는 프로토콜을 선택하고 네트워크 인터페이스 및 포트 설정을 진행한다. 기본적으로 OpenVPN은 UDP 1194 포트, WireGuard는 UDP 51820 포트를 사용하지만, 보안 강화를 위해 사용자가 원하는 포트를 설정할 수 있다. 이후 DNS 제공자 선택 및 고정 IP 또는 DHCP 설정 후 Raspberry Pi를 재부팅 하여 초기 설정을 마친다. VPN 서버 설정이 정상적으로 완료되면, 클라이언트가 접속할 수 있도록 클라이언트 사용자를 추가하고 Raspberry Pi에서 생성된 클라이언트 구성 파일을 클라이언트 장치로 복사하여 VPN을 사용할 수 있도록 한다. 이때 OpenVPN의 구성 파일은 .ovpn이고 WireGuard의 구성 파일은 .conf이다.

Fig. 4는 VPN 설치 중 나타나는 선택 과정이다. 이곳에서 OpenVPN을 선택하면 OpenVPN 서버를 구축할 수 있고 WireGuard를 선택하면 WireGuard 서버를 구축할 수 있다.



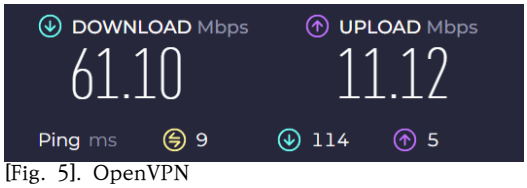
[Fig. 4] Part of the selection process

3.3 성능 및 보안성 평가 방법

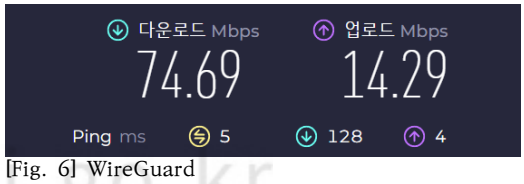
본 연구에서는 OpenVPN과 WireGuard의 성능 및 보안성을 비교 분석하기 위해 다양한 실험을 수행하였다. 본 절에서는 성능 및 보안성 평가 방법을 설명한다.

3.3.1 성능 평가

성능 평가는 두 프로토콜의 송신 속도, 수신 속도, Ping, CPU 사용률, 패킷 손실률, 지연 시간을 기준으로 측정하였다. 송신 속도, 수신 속도, Ping의 경우 Speedtest by Ookla에서 지원하는 인터넷 속도 측정을 이용해 100Mbps를 기준으로 외부 서버와의 속도 및 성능을 측정한다. Fig. 5과 Fig. 6은 측정 결과이다.



[Fig. 5]. OpenVPN



[Fig. 6] WireGuard

또한 속도 성능뿐만 아니라 어느 프로토콜이 다중 처리를 잘하고 패킷을 얼마나 잘 전달하는지 확인하기 위하여 내부망 내에서 서버와 클라이언트 간 트래픽 부하를 인위적으로 생성한 후 측정한다. 실험 환경은 100Mbps 속도와 UDP 프로토콜을 사용하고 네트워크 성능 측정 도구로 iperf3을 사용하였다. 이 평가를 통해 Raspberry Pi 서버와 사용자 PC의 컴퓨터 간의 네트워크 지연 시간, 응답 속도, 트래픽 처리 성능을 측정하며, 이를 바탕으로 OpenVPN과 WireGuard의 성능을 비교·분석한다.

3.3.2 보안성 평가

VPN의 구성 파일이 유출될 경우, 각 프로토콜의 보안 위협이 발생할 수 있다. OpenVPN은 추가적인 사용자 인증 정보를 제공하여 구성 파일만으로 VPN 서버에 접근할 수 없다. 반면 WireGuard는 구성 파일만 있으면 서버에 바로 접근할 수 있어 상대적으로 보안성이 낮다. 이에 본 논문에서는 공격자가 구성 파일을 통해 VPN 서버에 접근했다고 가정하여 두 가지 방법으로 보안성을 평가하였다.

보안성은 두 가지를 통해 평가하였다. 첫 번째로 인터넷 익명성(Anonymity on the Internet) 분석을 통해 실제 IP 주소의 노출 가능성을 확인하였다. VPN을 사용하면 사용자의 실제 IP 주소가 숨겨지기 때문에 다른 사용자가 이를 확인하거나 직접 통신할 수 없다. 그러나 서버의 로그나 구성 파일을 통해 클라이언트의 IP를 확인할 수 있다. 특히, 서버는 연결된 클라이언트 피어 정보를 항상 저장하며 공개키, VPN IP 주소, 실제 공인 IP 주소 등 클라이언트의 정보를 조회할 수 있고 이를 악용할 가능성도 존재한다. 따라서 본 논문에서는 OepnVPN과 WireGuard의 익명성을 비교하기 위해 서버 구성 파일(wg0.conf, server.conf), 서버 로그, 서버의 IP 구성 등을 사용하여 어느 프로토콜이 익명성을 더 보장하는지 확인한다. 두 번째로 Wireshark를 이용하여 패킷 분석을 수행하고 이를 통해 클라이언트의 VPN 사용 여부를 식별할 수 있는지 확인한다. 이 과정에서 패킷이 사용하는 통신 프로토콜, 공인 IP주소 등의 정보를 확인할 수 있는지 분석한다. 이를 통해 OpenVPN과 WireGuard 중 어느 프로토콜이 보안성 측면에서 더 우수한지 비교 및 분석한다.

4. 실험 결과

본 절에서는 OpenVPN과 WireGuard의 성능 및 보안성을 분석한 실험 결과를 통해 비교·평가하여 사용 환경에 따른 적합한 프로토콜 선택 기준을 제시한다.

4.1 VPN 통신 성능 분석

본 절에서는 OpenVPN과 WireGuard의 외부망과 내부망의 통신 성능을 비교·분석하였다. 이를 위해 각각 VPN 프로토콜의 송신 속도, 수신 속도, 네트워크 측정 서버로부터의 지연 시간을 확인하였다. 네트워크 속도는 측정 서버의 부하 상태에 따라 변동될 수 있으므로 같은 시간대에 측정하여 트래픽의 문제나 외부 변수의 영향을 최소화하고 VPN 서버 소프트웨어와 하드웨어가 일정한 성능을 유지하는지 확인하기 위해 5일 동안 매일 같은 시간이라는 일정한 조건에서 반복 측정하였다. 이는 매일 달라지는 서버 상태의 영향을 최소화하고, 일정한 시간대에서 성능을 평가하는 것이 중요하다는 기존 연구의 분석을 반영하였다.[14]. Table 1과 Table 2는 OpenVPN과 WireGuard의 성능 측정 결과이다.

〈Table 1〉 OpenVPN Performance Measurement

Trial	Upload Speed [Mbps]	Download Speed [Mbps]	Latency [ms]
1st	12.04	63.56	5
2nd	10.82	61.50	5
3rd	11.12	61.10	9
4th	10.06	61.05	6
5th	9.21	62.33	7
Average	10.7	61.9	6.4

〈Table 2〉 WireGuard Performance Measurement

Trial	Upload Speed [Mbps]	Download Speed [Mbps]	Latency [ms]
1st	11.99	76.16	6
2nd	10.64	74.37	6
3rd	14.29	74.69	5
4th	12.75	74.16	6
5th	11.42	76.04	5
Average	12.2	75.1	5.6

외부망에서의 성능뿐만 아니라 내부망에서의 성능을 확인하기 위해 네트워크 트래픽을 임의로 발생시켜 과부하가 되었을 경우와 네트워크에서 많은 양의 데이터를

처리할 시간이 길어질 경우를 가정하여 VPN 프로토콜에 따른 CPU 사용률, 패킷 손실률, 60초 동안 트래픽을 부하시켰을 때 클라이언트와 VPN 서버 사이에 발생하는 지연 시간을 확인하였다. CPU 사용량과 네트워크 성능 역시 트래픽 부하, 백그라운드 프로세스 등의 영향을 받으며 상황에 따라 측정값이 변동될 수 있으므로 일관성 있는 결과를 만들기 위해 5회 측정 후 평균값을 내어 표로 나타내었다. Table 3은 OpenVPN과 WireGuard의 네트워크 트래픽 부하 측정 결과이다.

〈Table 3〉 Measuring network traffic load

Category	OpenVPN	WireGuard
CPU Usage [%]	4.31	3.44
Packet Loss Rate [%]	0.00031	0.00022
Latency [ms]	0.546	0.424

측정 결과, WireGuard가 OpenVPN보다 송신 속도와 수신 속도가 빠르고 지연 시간이 더 낮아 성능이 우수한 것으로 나타났다. 내부망에서의 성능 또한 WireGuard가 CPU 사용률이 낮고 패킷 손실도 적었으며, 지연 시간 역시 더 짧았다. 이를 통해 WireGuard가 OpenVPN보다 데이터 전송 속도가 더 빠르고 트래픽 간 처리 효율이 높다는 것을 알 수 있었다. 따라서 통신 속도와 트래픽 처리 성능 측면에서는 WireGuard가 OpenVPN보다 우수하다는 것을 확인하였으며, 속도와 성능을 중요시하는 사용자에게 WireGuard 사용을 권장한다.

4.2 VPN 프로토콜 보안성 분석

본 절에서는 OpenVPN과 WireGuard의 보안성을 구성 파일 유출 시의 보안 위험성, 사용자 익명성, 통신 트래픽 암호화 및 사용자 식별 가능성 측면에서 비교·분석한다.

4.2.1 구성 파일 유출 시의 보안 위험성 분석

OpenVPN의 구성 파일인 .ovpn 파일은 신원을 검증하는 인증 기관인 CA(Certificate Authority), 클라이언트를 식별하는 공개 키(Client Certificate), 그리고 해당 클라이언트 개인 키(key, Client Private Key)를 포함한다[15]. 따라서 구성 파일이 있다면 유출될 경우, 누구나 VPN 서버에 접속할 수 있다. 하지만 OpenVPN은 기본적으로 사용자 인증 방식(User Authentication)을 사용하므로, Fig. 7과 같이 추가적인 인증 절차를 거쳐야 접속할 수 있다.

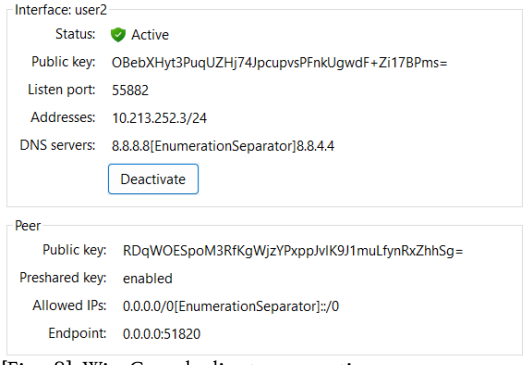
[Fig. 7] OpenVPN User Authentication Process

클라이언트 생성 시 'nopass' 옵션을 사용하면 개인 키를 암호 없이 저장되므로, 구성 파일만으로도 즉시 접속할 수 있다. 그러나 OpenVPN은 세션(Session) 관리 기능을 제공하므로, 서버 관리자가 Peer를 삭제하면 바로 접속이 종료되므로 추가적인 접근이 차단된다.

반면, WireGuard의 구성 파일인 .conf 파일은 클라이언트의 개인 키(Private Key), 서버의 공용 키(Public Key), 그리고 VPN 내부 IP(Allowed IPs) 정보가 포함되어 있다. WireGuard는 사용자 인증 방식이 아닌 고정된 키 교환(Key Exchange) 방식을 사용하여 클라이언트를 인증한다. 즉, 별도의 ID나 비밀번호 입력 없이 사전에 등록된 공개 키와 개인 키의 일치 여부를 검증하여 접속을 허용한다.

따라서 키가 유출될 경우, 누구나 추가적인 별다른 인증 없이 해당 클라이언트로 변장하여 VPN에 접속할 수 있다. 따라서 WireGuard를 사용할 경우, 개인 키를 필수적으로 보호해야 한다. Fig. 8은 인증 없이 연결이 성공한 결과이다. 만약, 공격자가 사용자의 VPN에 접근하여 서버 관리자가 이를 차단하기 위해 Peer를 삭제하더라도, OpenVPN과 다르게 WireGuard는 즉시 연결이 종료되지 않는다. 이는 최소 몇 초에서 최대 몇 분 동안 연결이 유지될 수 있으며, WireGuard는 상태 저장형(Stateful) 프로토콜이므로 항상 연결을 유지하는 구조가 아니다.

요약하면, OpenVPN과 WireGuard 모두 구성 파일이 유출된다면 보안 위협에 노출될 수 있다. 그러나 OpenVPN은 추가적인 사용자 인증 과정을 제공하므로, WireGuard보다 상대적으로 보안성이 높다.



[Fig. 8] WireGuard client connection

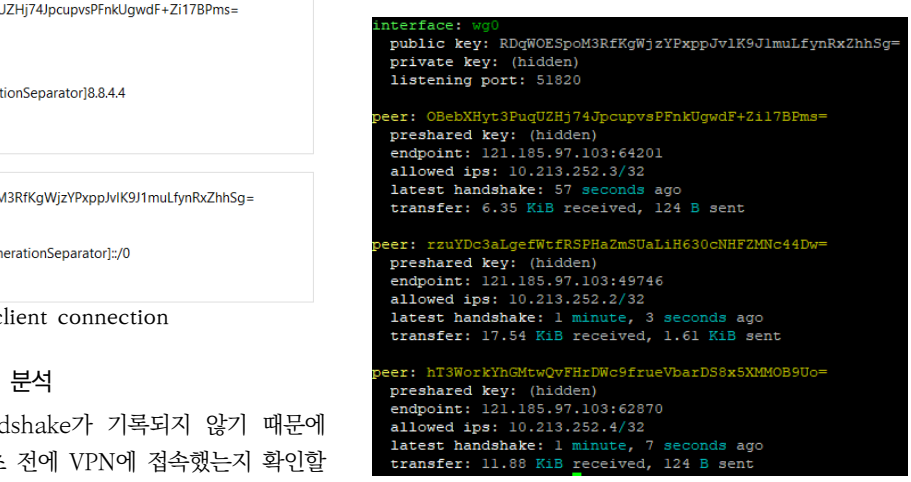
4.2.2 VPN 익명성 분석

OpenVPN은 handshake가 기록되지 않기 때문에 사용자가 몇 분, 몇 초 전에 VPN에 접속했는지 확인할 수 없다. 따라서 특정사용자를 식별하기 어렵다. 실제로, OpenVPN은 고정 IP를 사용하지 않고 DHCP를 사용하여 IP를 동적으로 할당하므로 개별 사용자를 특정하는 것이 불가능하다. OpenVPN이 기본적으로 VPN 사용자를 특정할 수 없도록 설계되었지만, 로그 파일(openvpn-status.log)에 접근하면 사용자의 공인 IP(End Point)를 확인할 수 있다. 따라서 접속 시간과 내부 IP는 숨길 수 있지만 로그 파일을 통해 사용자의 공인 IP가 노출될 수 있으므로 익명성이 완벽하게 보장되지는 않는다.

반면, WireGuard는 공격자가 서버에 'sudo wg show'라는 명령어를 실행하면 해당 서버의 구성 정보를 확인할 수 있다. Fig. 9는 명령어를 통해 확인한 WireGuard의 서버 구성 정보를 나타낸다.

WireGuard는 OpenVPN과 다르게 latest handshake가 기록되기 때문에 사용자가 몇 분, 몇 초 전에 사용했는지 확인할 수 있다. 이를 통해 내부 인터넷과 비교하여 특정 시간대에 접속한 사용자를 특정하고 추적할 수 있다. 또한 WireGuard는 고정된 내부 IP를 사용하므로, 공격자가 특정 IP와 트래픽 패턴을 분석하여 사용자를 특정할 수 있다. 서버 구성 정보를 통해 가장 중요한 정보인 사용자의 공인 IP와 VPN 서버를 확인할 수 있으며 이에 따라 공인 IP를 사용하는 네트워크 기기가 공격자에게 노출될 위험이 존재한다.

요약하면, OpenVPN이 세션 정보를 기록하지 않으므로 WireGuard보다 상대적으로 익명성이 높다. 그러나 OpenVPN은 로그 파일을 통해서, WireGuard는 handshake 기록을 통해 사용자를 특정할 수 있으므로 두 프로토콜 모두 완전한 익명성을 보장하지 않는다.

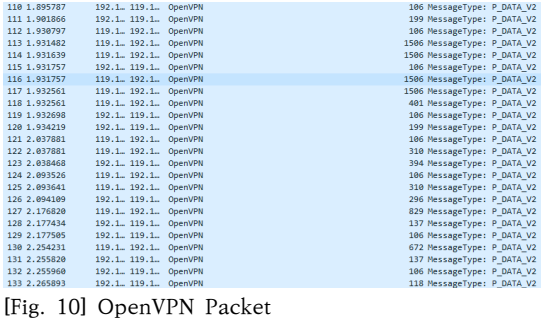


[Fig. 9] WireGuard server configuration information

4.2.3 통신 트래픽 암호화 및 사용자 식별 가능성 분석

VPN 패킷에는 출발지(Source), 목적지(Destination), 프로토콜(Protocol) 등의 정보가 포함되며 이를 통해 VPN을 통한 통신 과정을 확인할 수 있다.

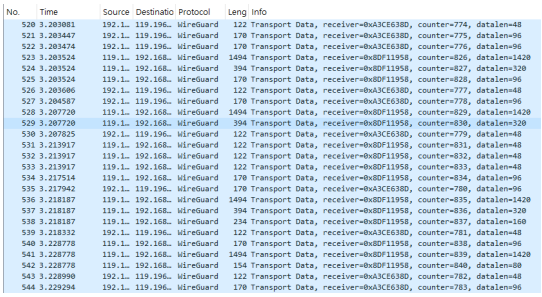
OpenVPN의 패킷은 'P_DATA_V2' 메시지로 표시되며, 이는 암호화된 페이로드가 포함된 패킷을 의미한다. 해당 패킷의 페이로드는 암호화되어 있으므로 내부 데이터를 복호화할 수 없다. Wireshark에서 OpenVPN을 인식하고 있으므로 패킷 식별이 가능하지만, 기본적인 전송 정보(IP, 포트 등)만 확인할 수 있고 상세한 트래픽 정보는 암호화된 페이로드로 인해 확인할 수 없다. Fig. 10은 OpenVPN 패킷 일부를 캡처한 사진으로 OpenVPN이 패킷을 완전히 보호하는 구조를 갖추고 있어 트래픽 패턴을 분석하기 어려운 특징을 가진다.



[Fig. 10] OpenVPN Packet

반면, WireGuard의 패킷은 'Transport Data'로 페이로드가 암호화 되어 있어 내부 데이터를 복호화할 수

없다. 하지만 피어를 식별할 수 있는 receiver 값과 패킷의 순서를 나타내는 counter 값이 노출되어 있다. 이를 통해 패킷의 순서와 패킷의 패턴을 쉽게 감지할 수 있다. 따라서 특정사용자의 트래픽 패턴을 분석할 수 있다. Fig. 11은 실험 중 캡처한 WireGuard 패킷 일부를 캡처한 사진으로, 네트워크 환경에서 WireGuard 트래픽의 흐름을 쉽게 추적할 수 있으며, 특정 사용자의 트래픽 패턴이 분석 가능하다는 것을 보여준다.



[Fig. 11] WireGuard Packet

요약하면, 두 프로토콜 모두 페이로드가 암호화 되어 있으므로 복호화는 불가능했으나, 암호화 수준의 차이를 통해 패킷의 패턴 분석 가능 여부가 달라졌다. OpenVPN은 'P_DATA_V2'로 암호화되어 패킷 식별 및 기본적인 정보 확인이 가능했지만, 패킷의 패턴을 분석할 수 없었다. 반면 WireGuard는 'Transport Data'로 암호화되어 있지만 receiver 값과 counter 값이 노출되어 있으므로 기본적인 정보 확인뿐만 아니라 패킷의 패턴 분석이 가능하다.

4.2.4 성능 및 보안성 측정 정리

Table 4는 OpenVPN과 WireGuard의 성능 및 보안성 평가 결과를 종합하여 정리한 것이다. 측정 결과, WireGuard가 OpenVPN보다 송신 속도는 약 14.02%, 수신 속도는 약 21.32% 빠르며 외부망에서의 지연 시간은 약 12.5% 더 낮다. 이는 응답 속도가 더 빠르다는 것이다. 마찬가지로 WireGuard가 OpenVPN보다 CPU 사용률은 약 20.19%, 패킷 손실률은 약 29.03%, 내부망에서의 지연 시간은 약 22.34% 더 낮다. 따라서 비교 결과, 전반적으로 WireGuard가 OpenVPN보다 성능 면에서 우수하다는 것을 확인하였다.

보안성 측면에서 구성 파일이 유출될 경우, OpenVPN은 사용자 인증 절차를 요구하므로 구성 파일만으로는 VPN 접속이 불가능하다. 반면, WireGuard는 별도의 인증 절차를 제공하지 않으므로 구성 파일만으로도 접속할 수 있어 보안 위험이 상대적으로 높다. 익명성 측면에서도 OpenVPN은 handshake 기록을 남기지 않아 접속 시간과 내부 IP가 노출되지 않지만, 로그 파일을 통해 공인 IP가 확인될 가능성이 있다. 반면 WireGuard는 서버 명령어를 통해 해당 서버의 구성 정보를 확인할 수 있어 접속 시간과 공인 IP가 쉽게 노출될 수 있다. 트래픽 분석 측면에서도 OpenVPN과 WireGuard 모두 패킷 내 데이터는 암호화되어 직접적인 복호화는 불가능하지만, WireGuard의 경우 receiver 값과 counter 값이 노출되어 있으므로 패킷의 패턴 분석이 가능하다.

따라서 구성 파일 보안과 사용자 익명성 및 트래픽 패턴 분석 여부를 고려할 경우, WireGuard는 상대적으로 보안 위험이 높으므로, OpenVPN이 보다 높은 보안성을 제공한다.

정리하면, 성능 측면에서는 WireGuard가 OpenVPN보다 빠른 속도와 낮은 지연 시간을 제공하지만 OpenVPN이 추가적인 인증 절차를 통해 보안을 강화하며, WireGuard보다 높은 익명성을 제공하고 트래픽 분석이 불가능하여 보안성 측면에서는 OpenVPN이 더 우수한 보안성을 제공한다.

〈Table 4〉 Summary of Performance Evaluation

Category	OpenVPN	WireGuard
Encryption Algorithm	AES-256	ChaCha20
Transmission Protocol	TCP, UDP	UDP
Upload Speed	10.7Mbps	12.2Mbps
Download Speed	61.9Mbps	75.1Mbps
Ping	6.4ms	5.6ms
CPU Usage	4.31%	3.44%
Packet Loss Rate	0.00031%	0.00022%
Latency	0.546ms	0.424ms
VPN Access via Config File	Not Possible	Possible
User Identification (Anonymity)	Not Possible	Possible
Traffic Pattern Analysis	Not Possible	Possible

5. 결론

본 논문은 네트워크 보안 환경에서 널리 사용되는 OpenVPN과 WireGuard의 성능과 보안성을 비교 및 분석하여 사용자가 네트워크 환경에 따라 적절한 VPN 프로토콜을 선택할 수 있는 기준을 제시하였다. 이를 위해 저사양 환경인 Raspberry Pi 5를 기반으로 개인용

VPN 서버를 구축하고 OpenVPN과 WireGuard의 성능과 보안성을 분석하였다. 실험 결과, 성능 측면에서는 WireGuard가 빠른 데이터 전송 속도와 CPU 사용률에서 OpenVPN보다 우수한 성능을 보여 주었으나, 구성 파일 유출 시에는 OpenVPN이 WireGuard보다 높은 보안성을 유지하고 익명성 및 트래픽 분석이 상대적으로 불가능한 것으로 확인되었다. 이를 통해 성능이 중요한 환경에서는 WireGuard가 적합하고 보안성이 중요한 환경에서는 OpenVPN이 적합하다는 결론을 도출하였다. 추가로 OpenVPN을 사용하려는 사용자는 보안성을 고려하여 사용자 인증 절차를 사용하기 위해 클라이언트 추가 시 'nopass' 옵션을 사용하지 않을 것을 권장한다.

본 연구는 OpenVPN과 WireGuard 프로토콜 선택에 대한 방향성을 제시하였다. 본 연구의 결과가 VPN 프로토콜의 선택 과정에서 사용자의 결정에 도움 될 것으로 기대한다. 향후 연구에서는 WireGuard의 보안성을 강화하기 위한 추가적인 보안 방법을 탐색하여 후속 연구를 진행할 계획이다.

REFERENCES

[1] Sagn-Hwi Kim, Hyoun-Jong Kim, "Market trend and perspectives of network security," The Journal of Korean Institute of Communications and Information Sciences, pp.1266-1269, 2001.

[2] Yun Dong Sic, "Intrusion detection agents on the wireless network design," Journal of convergence security, Vol.13, No.1, pp.59-70, 2013.

[3] Moon Jaeyeon, Chang Younghyun, "Ransomware Analysis and Method for Minimize the Damage," International Agency for Culture and Technology Promotion, Vol.2, No.1, pp.79-85, 2016.

[4] Kim Inhwan, Kim Dukyun, Cho Sungkuk, Jeon Byungkook, "A Method for Original IP Detection of VPN Accessor," The Journal of The Institute of Internet, Broadcasting and Communication, Vol.21, No.3, pp.91-98, 2021.

[5] Ray Walsh, "Which VPN protocol is best? How to choose the right VPN protocol," Available: Bleeping Computer, https://www.bleepingcomputer.com/vpn/guides/vpn-protocols/, Accessed: Jan, 2025.

[6] OpenVPN, "Community Wiki and Tracker", Available: https://community.openvpn.net/openvpn/wiki/OverviewOfOpenvpn#OpenVPNOSS, Accessed: 2025.

[7] Yang Seung-Eui, Kim Hong-Kyun, "Implementation of OpenWRT based VPN server system for the IoT," The Institute of Electronics and Information Engineers, pp.1844-1848, 2016.

[8] Kim Dongjoo, Lee SangKye, Park Junho, Seong Dongook, Yoo Jaesoo, "A Differential AES Encryption Scheme for Wireless Sensor Networks", Korean Institute of Information Scientists and Engineers, pp.282-285, 2011.

[9] Lee Kwang Kyu, "File Protection Technique using AES Encryption," Korean Institute of Smart Media, Vol.13, No.12, pp.69-75, 2024.

[10] Jeon Hyeongseok, Lee SungKee, "Analysis and Implementation of Encryption Algorithms for Remote Update Security of IoT Healthcare Device." Korean Institute of Information Technology, Vol.19, No.7, pp.91-99, 2021.

[11] WireGuard, "WireGuard FAST, MODERN, SECURE VPN TUNNEL", Available: https://www.wireguard.com/#conceptual-overview, Accessed: Feb, 2025.

[12] Yoav Nir, Adam Langley, "ChaCha20 and Poly1305 for IETF Protocols," RFC 7539, 2015.

[13] Hong Seok Jin, Jeong Min Su, Kim Su A, Jeong Eui Rim, "Raspberry pi based Smart braile converter for visally impaired", The Korean Institute of Communications and Information Sciences, pp.994-995, 2023.

[14] Kim Jaehoon, Ok Tomin, Lee Youngseok, Choi Yanghee, "End-to-End Internet Performance Measurement and Analysis Using the Active Probing, Korean Institute of Information Scientists and Engineers", Korean Institute of Information Scientists and Engineers, Vol.26, No.2, pp.241-243, 1999.

[15] Park Seung-Kyun, "Research onSecurity and Quality of Service (QOS) in OpenVPN, The Society of Convergence Knowledge Transactions, Vol.12, No.4, pp.2219-227, 2024.

이 정 연(Jung-Yeon Lee) [준회원]

■ 2023년 3월 ~ 현재 : 수원대학교 정보보호학과 학사과정

〈관심분야〉
정보보호

전 상 훈(Sanghoon Jeon) [정회원]

- 2012년 2월 : 경북대학교 IT대학 심화 전자공학 공학사
- 2014년 2월 : 대구경북과학기술원 정보통신융합공학전공 공학석사
- 2020년 8월 : 대구경북과학기술원 정보통신융합전공 공학박사
- 2020년 3월 ~ 2020년 8월 : 한양대학교 산학협력단 선임연구원
- 2020년 9월 ~ 2022년 9월 : 한양대학교 의과대학 응급의학과 포닥연구원
- 2022년 10월 ~ 2023년 9월 : 한양대학교 의과대학 응급의학과 연구조교수
- 2023년 10월 ~ 현재 : 수원대학교 지능형SW융합대학 정보보호학과 조교수

〈관심분야〉
웨어러블컴퓨팅, 의료인공지능, CPS보안