Comparative Analysis of IPv4 and IPv6 OpenVPN Protocol Performance Based on QoS Parameters

Pradistia Edo Kristianto [1*], Anggyi Trisnawan Putra [1]

[1] Department of Computer Science, Faculty of Mathematics and Natural Sciences, Universitas Negeri Semarang, Semarang, Indonesia
*Corresponding author: pradistiaedo@students.unnes.ac.id

ARTICLE INFO

Article history
Received 11 February 2021
Revised 16 March 2021
Accepted 12 April 2021

Keywords
Virtual Private Network
OpenVPN
IPv4
IPv6
QoS

ABSTRACT

Network security is still a big concern in data exchange because it involves users' data privacy. There are several ways to secure data exchange on the internet. One of them uses a Virtual Private Network (VPN), which functions as a tunnel in the public internet that securely connects users to the local network. This research will analyze the performance of the OpenVPN IPv4 and IPv6 protocols. The method used to determine the performance results is based on the QoS parameters. The performance analysis results obtained are throughput OpenVPN IPv6 is better, namely 198.155 Kbps on ICMP data packets, 35.704 Kbps on FTP data packets, and 17.698 on TCP data packets. The delay value of OpenVPN in IPv4 is superior, namely 1.4s for ICMP data packets, and on IPv6, FTP data packages are superior with 0.1s. Jitter values indicate OpenVPN IPv6 is better with similar results. Packet loss values on OpenVPN for both IPv4 and IPv6 protocols are 0%. Based on these results, throughput IPv6 OpenVPN on ICMP data packets and delay on FTP data packets is better than IPv4 OpenVPN.

1 Introduction

In recent years, internet network security is still a thing that is continually reviewed by researchers, especially in this day and age where internet users and connected devices are increasing, resulting in data security becoming vulnerable to being hacked or misused. One method to secure data traffic on the internet is to use a Virtual Private Network (VPN) (Skendzic & Kovacic, 2017).

VPN is a technology that can create a private network by utilizing public networks to secure the data exchange process (Yang *et al.,* 2019). In VPN, a combination of tunneling and encryption technology makes VPN a reliable technology to solve security problems in the network (Jyothi & Reddy, 2018).

In its implementation, VPN is used to connect between 2 or more places located far away, such as headquarters with branch offices, a company with a partner company, or one campus with separate buildings in other areas (Jing *et al.,* 2017). A VPN used to connect a company with another company (e.g., partners and customers) is called an extranet. Whereas when a VPN is used to connect the head office with a branch office, it is called intranet site-to-site VPN (Iqbal, 2019). Some tunneling that can be used includes Internet Key Exchange (IKEv2), OpenVPN, Point to Point Tunneling Protocol (PPTP), and Layer Two Tunneling Protocol (L2TP) (Alshalan, Pisharody, & Huang, 2016). However, each tunneling on a VPN has a different performance.

According to Dliyauddin and Muslim (2019), in research comparing the performance analysis between IKEv2 and OpenVPN, the IKEv2 protocol performs better at a maximum load of 1536 bytes compared to the performance of the OpenVPN protocol.

Zakari et al. (2019), which compares IPv4 and IPv6 protocols, show that IPv6 is robust and has performed better than IPv4 on both experimental modes (voice and video).

From some of these studies, the IP Address used is IPv4. There is still little research analyzing tunneling protocol VPN¬ performance using IPv6, which is the new standard today. The Internet Assigned Numbers Authority (IANA), an international official IP address management agency, released data that the number of addresses leftover from IPv4 in 2011 was approximately 10 percent of the initial capacity or about 400 million addresses alone. This number is not adequate to anticipate the development of internet users today, which is remarkable, coupled with the development of future Telecommunication technology based on IP. IPv6 is a new internet protocol developed to anticipate the soon-to-be full IPv4 protocol (Zakari *et al.*, 2019).

Based on the problem description above, research is proposed to measure VPNs' performance in IPv4 and IPv6. One of the VPN protocols that support IPv6 is OpenVPN. Therefore, research will be conducted comparing VPN performance using OpenVPN protocol in IPv4 and IPv6 to find out how OpenVPN can run well on IPv6, which will be widely applied in place of IPv4 due to the depletion of IPv4 availability. This analysis can be taken from QoS results in the form of throughput, delay, jitter, and packet loss.

2 Methods

This research was conducted to classify the movie review dataset using the method approach to get better accuracy results. The classification process of sentiment analysis in this study is carried out according to Figure 1. The process consists of several main stages: server-client installation and configuration, sending data from client to VPN server, capturing data, and QoS analysis.

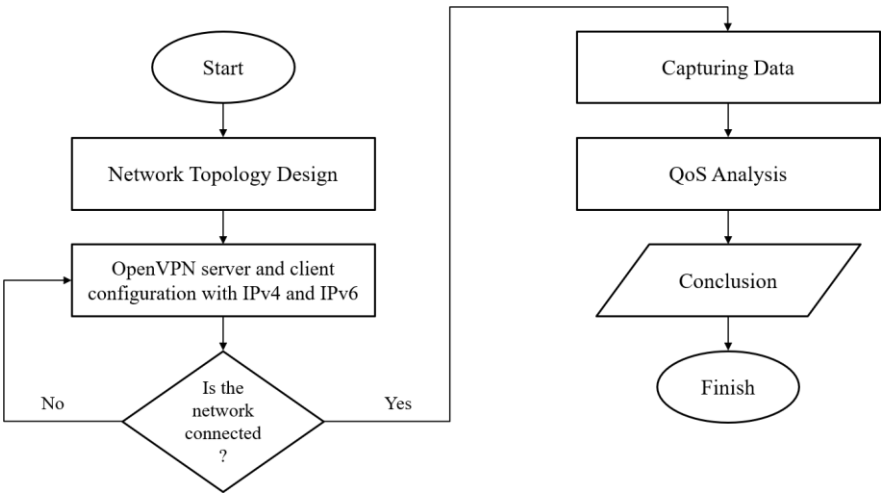

Figure 1. The stages of research

2.1 Network Design

This research was conducted by implementing a type of Remote-Access VPN using a local network in each network configuration tested. The network is not connected to the internet, and only communication between the PCs/laptops configuration used in this study has two kinds, namely IPv4 and IPv6 protocol configuration (Deshmukh & Iyer, 2017). As a test will be done data transfer between the server and the client, the server-side is configured FTP server to put files to be downloaded by the client, while on the client-side installed Wireshark software capture data packets that pass through the network.

The client will use IP address 192.168.1.7, while the Ubuntu server will use IP Address 192.168.1.5. The initial configuration of the client is in the same position when connected to the

OpenVPN server, where the client gets IPv4 10.8.0.6 and IPv6 2001:db8:ee00:abcd::1000 from the OpenVPN server.

2.2 Client-Server Installation and Configuration

After the network concept is designed, the installation and configuration are carried out on the OpenVPN server and client to connect with the server. After the client is connected to the server, the packet will be sent from the client to the server and vice versa. The parameters calculated are based on QoS, namely throughput, jitter, delay, and packet loss (Mushtaq & Patterh, 2018).

2.3 Capturing Data

The process of capturing data using Wireshark application assistance by sending Internet Control Message Protocol (ICMP) client data packets to the server and using File Transfer Protocol (FTP) package to download files from server to client. Capturing data is done on the client. With the help of the Wireshark application calculation, data will appear so that throughput, delay, jitter, and packet loss estimates can be performed. The results of capturing data will be processed at the next stage of data calculation. At the stage of data, analysis occurs in calculating data based on the formula of the standard QoS parameters (Nawej & Du, 2019).

2.4 Capturing Data

Analysis of the results is the most crucial stage in the completion of scientific research activity. This study conducted the analysis using QoS parameters that include throughput, jitter, packet loss, and delay. At this stage, throughput, jitter, packet loss, and delay calculations are performed after sending data in IPv4 and IPv6 protocols at a specific time by using the capture packet results from the Wireshark application and then from the data then done manual calculation with QoS calculation formula (Narayan *et al.,* 2016).

The parameters used in QoS in this study consisted of:

1. Throughput

Throughput is the speed (rate) of effective data transfer measured in bps (bits per second). Throughput is the total number of successful packet arrivals observed at a destination during a given time interval divided by the duration of that time interval.

Equation 1 throughput calculation:

$$Throughput = \frac{received\ data\ packet}{delivery\ time} \quad (1)$$

2. Packet Loss

Packet Loss is a parameter that describes a condition that indicates the total number of packets lost that can occur due to collisions and congestion on the network.

Equation 2 packet loss calculation:

$$Packet\ loss = \frac{\text{received data packet}}{Data\ package\ sent}\ x\ 100\% \quad (2)$$

3. Delay (latency)

Delay (latency) is when it takes data to travel the distance from the origin to the destination. Delays can be affected by distance, physical media, or long processing time.

Equation 3 calculation of delay (latency):

$$Delay = \frac{total\ delay}{total\ packets\ received} \quad (3)$$

4. Jitter or Variations of Package Arrivals

Jitters are caused by variations in queue length, data processing time, and recollection of packages at the end of the trip. Jitters are commonly called delay variations, closely related to latency, which indicates the large number of delay variations in data transmission on the network.

Equation 4 calculation of jitter:

$$Jitter = \frac{total\ delay\ variation}{total\ packets\ received} \quad (4)$$

3 Results and Discussion

The results of QoS analysis of ICMP data packets after calculation can be seen in Table 1.

Table 1. ICMP data package QoS analysis results

Protocol	*Throughput (kbps)*	*Delay (s)*	*Jitter (s)*	*Packet Loss*
IPv4	195,89675711	1,430511898	0,051096	0
IPv6	198,15569375	2,247636847	0,051095	0

It is known that the calculation result of throughput on IPv6 protocol is more significant than 2.26 kbps compared to IPv4 protocol throughput result. The IPv4 protocol delay value is smaller than IPv6, with a difference of 0.82s. The jitter value is relatively the same, and the value of the second loss packet protocol is 0, or no packets are missing.

Then in table 2 is shown the results of FTP data package analysis as follows.

Table 2. FTP data packet QoS analysis results

Protocol	*Throughput (kbps)*	*Delay (s)*	*Jitter (s)*	*Packet Loss*
IPv4	17,31364610	0,290162988	0,002503	0
IPv6	35,70472194	0,175572221	0,004064	0

From the FTP data packet calculation table above, the IPv6 protocol throughput and delay values are better because they have large throughput but minor delays with a difference of 18.4 kbps on throughput values and 0.12s in delay values. The jitter values of the two protocols do not differ much where IPv4 is slightly smaller with a difference of 0.001561s, and the packet loss value of both protocols is the same as 0.

The results of the analysis of TCP data packets on the IPv4 protocol generated by D-ITG applications are shown in table 3.

Table 3. QoS data analysis results of TCP IPv4 Protocol

TCP Packet Data Size *(bytes)*	*Throughput (Kbps)*	*Delay (s)*	*Jitter (s)*	*Packet Loss*
64	0,651	0,186461	0,000293	0
128	1,204	0,189430	0,000239	0
256	2,267	0,189430	0,000239	0
512	4,437	0,189430	0,000239	0
1024	8,594	0,185599	0,000235	0

The larger the data packet size, the higher the throughput generated. The delay value at the data package size of 128, 256, and 512 bytes is the same, while the most negligible delay and jitter values are found in the most extensive data packet size tested, which is 1024 bytes. For packet loss, the value on all data packet sizes is the same, which is 0.

Further analysis of TCP data packets on the IPv6 protocol generated by D-ITG can be seen in Table 4.

Table 4. IPv6 protocol TCP data QoS analysis results

TCP Packet Data Size*(bytes)*	*Throughput (Kbps)*	*Delay (s)*	*Jitter (s)*	*Packet Loss*
64	0,677	0,185929	0,000264	0
128	1,269	0,185929	0,000264	0
256	2,361	0,190114	0,000230	0
512	4,582	0,188875	0,000211	0
1024	8,809	0,194041	0,000229	0

The table above shown the variation of throughput that is getting higher following the size of the data package. The slightest delay value is found in the data packet sizes of 64 and 128 bytes and the jitter value. Then the packet loss value is 0 for all data packet sizes.

Overall based on the protocol comparison table above, it is known that the calculation result of ICMP and FTP data packet throughput calculation on IPv6 protocol is more remarkable compared to the throughput generated by IPv4 protocol. TCP data packets on the IPv6 protocol also produce greater throughput. The comparison can be seen in Figure 2.

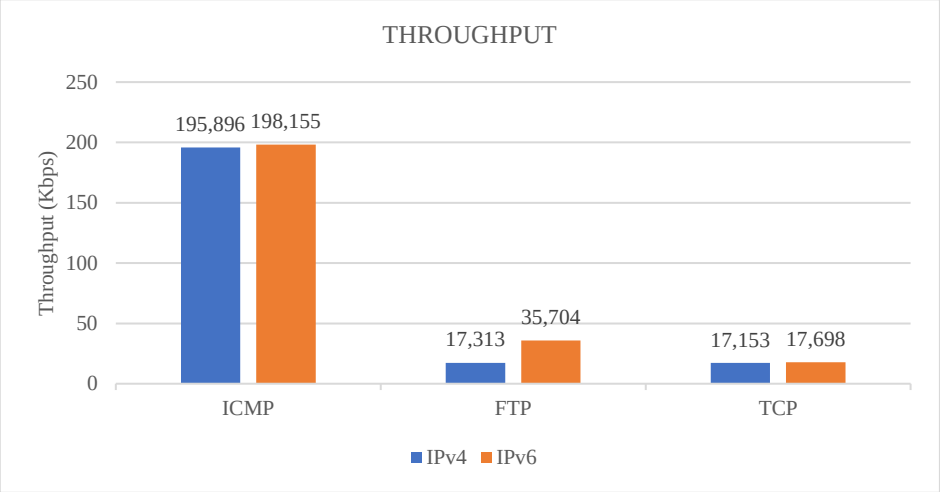

Figure 2. Throughput comparison

Then, there is a delay difference from the calculation results where ICMP data packets have more minor delays on IPv4 protocol than on IPv6 protocol. At the same time, the FTP data packet delay on the IPv6 protocol is smaller than the IPv4 protocol. Then the TCP data packet differs slightly, as shown in Figure 3. The smaller the delay shows, the better the network performance and protocols in data transmission.

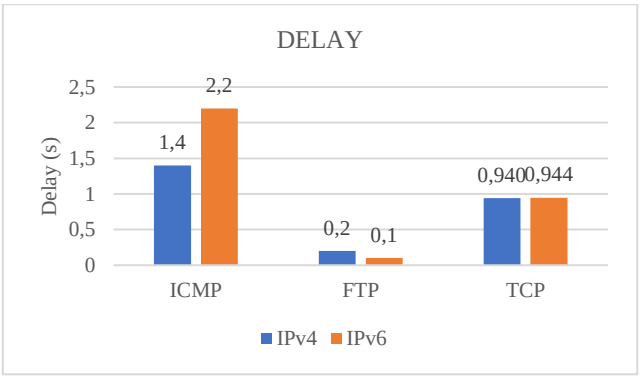

Figure 3. Delay comparison

Then for the jitter value obtained, results are not much different on both IPv4 and IPv6 protocols. The JITTER value of ICMP data packets on the IPv6 protocol is smaller than the IPv4 protocol. On the contrary, the Jitter value of FTP data packets on the IPv4 protocol is smaller than that of the IPv6 protocol. As for TCP data packets, the difference is fragile where the IPv6 protocol has a more negligible jitter. The comparison of jitter values can be seen in Figure 4.

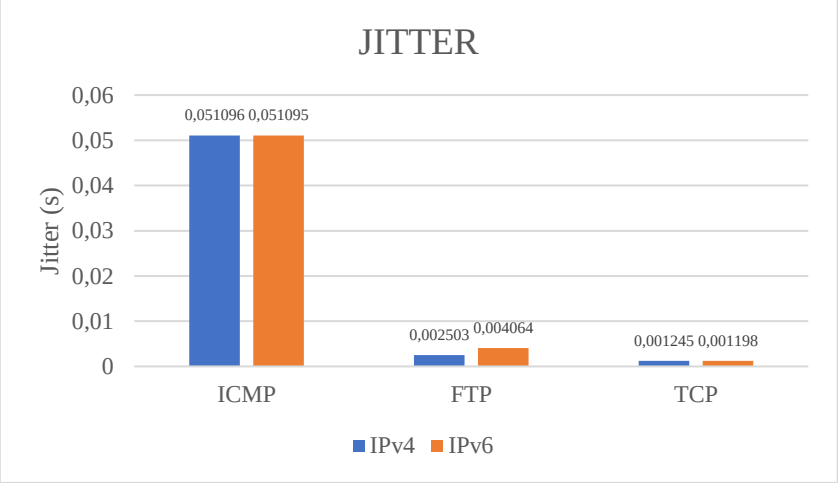

Figure 4. Jitter comparison

For packet loss value obtained from all data packets and protocols is 0%, or no packets are lost during data transmission, as shown in Figure 5.

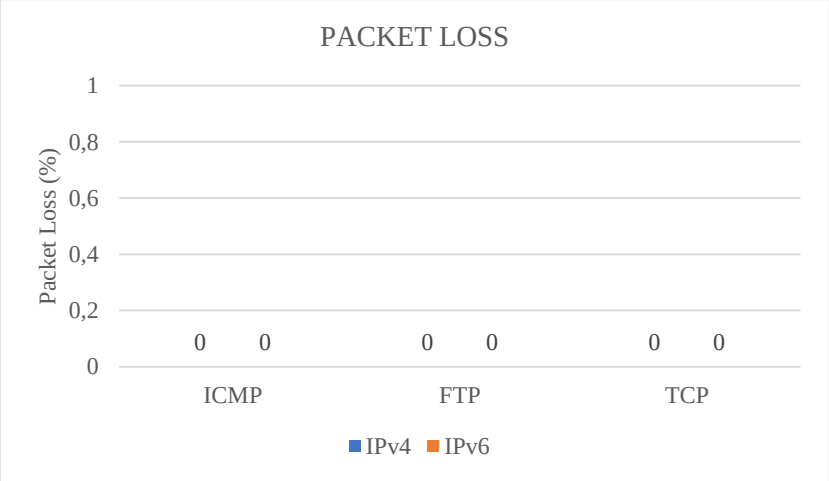

Figure 5. Packet loss comparison

4 Conclusion

Based on the comparison between IPv4 and IPv6 OpenVPN, IPv6 protocol produces a higher throughput value than the IPv4 protocol in data transmission, which is 198.15 Kbps on ICMP and 35.70K Kbps on FTP data packets. Then for the delay value, ICMP data packets have a smaller delay on the IPv4 protocol, which is 1.43s than in the IPv6 protocol. At the same time, the FTP data packet delay on the IPv6 protocol is more negligible than the IPv4 protocol, which is 0.17s. Then for the resulting jitter value is not much different on both IPv4 and IPv6 protocols. The jitter value of ICMP data packets on the IPv6 protocol is smaller than the IPv4 protocol. On the contrary, the jitter value of FTP data packets on the IPv4 protocol is smaller than that of the IPv6 protocol.

For TCP data packets, the throughput generated by both protocols is higher following the size of the data packet being sent. Then the smallest TCP data packet delay value is found in the IPv4 protocol, with a data package size of 1024 bytes with a value of 0.185s. The smallest jitter value on

TCP data packets is 0.000211s on IPv6 protocols with a data packet size of 512 bytes and only has a tiny difference with IPv4. The packet loss value of all data packets and protocols is 0%, or no packets are lost when sending data. This research shows that the OpenVPN IPv6 protocol has better performance because it produces large throughput but with a slight delay, which is a need to transfer files at a better speed and secure and overcome the limitations of existing IPv4.

References

Alshalan, A., Pisharody, S., & Huang, D. (2016). A Survey of Mobile VPN Technologies. *IEEE Communications Surveys and Tutorials*, *18*(2), 1177–1196. doi: 10.1109/COMST.2015.2496624

Deshmukh, D., & Iyer, B. (2017). Design of IPSec Virtual Private Network for Remote Access. *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, 716–719. doi: 10.1109/CCAA.2017.8229894

Dliyauddin, M., & Muslim, M. A. (2019). Comparison Analysis of Ikev2 And OpenVPN Protocol Performance on Wired Network. *Scientific Journal of Informatics, 6(1), 2–9.*

Iqbal, M. (2019). Analysis of Security Virtual Private Network (VPN) Using OpenVPN. *International Journal of Cyber-Security and Digital Forensics*, *8*(1), 58–65. doi: 10.17781/p002557

Jing, S., Qi, Q., Sun, R., & Li, Q. (2017). Study on VPN Solution Based on Multi-Campus Network. *Proceedings - 2016 8th International Conference on Information Technology in Medicine and Education, ITME 2016*, 777–780. doi: 10.1109/ITME.2016.0180

Jyothi, K. K., & Reddy, B. I. (2018). Study on Virtual Private Network (VPN), VPN's Protocols, and Security. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *3*(5), 919–932.

Mushtaq, A., & Patterh, M. S. (2018). QoS Parameter Comparison of DiffServ-aware MPLS Network using IPv4 and IPv6. *International Conference on Recent Innovations in Signal Processing and Embedded Systems, RISE 2017*, 113–118. doi: 10.1109/RISE.2017.8378136

Narayan, S., Ishrar, S., Kumar, A., Gupta, R., & Khan, Z. (2016). Performance Analysis of 4to6 and 6to4 Transition Mechanisms Over Point to Point and IPSec VPN protocols. *IFIP International Conference on Wireless and Optical Communications Networks, WOCN*, *2016-Novem*, 0–6. doi: 10.1109/WOCN.2016.7759027

Nawej, C. M., & Du, S. (2019). Virtual Private Network's Impact on Network Performance. *2018 International Conference on Intelligent and Innovative Computing Applications, (ICONIC 2018)*, 1–6. doi:10.1109/ICONIC.2018.8601281

Skendzic, A., & Kovacic, B. (2017). Open Source System OpenVPN in a Function of Virtual Private Network. *IOP Conference Series: Materials Science and Engineering*, *200*(1). doi: 10.1088/1757-899X/200/1/012065

Yang, D., Wei, H., Zhu, Y., Li, P., & Tan, J. C. (2019). Virtual Private Cloud-Based Power-Dispatching Automation System-Architecture and Application. *IEEE Transactions on Industrial Informatics*, *15*(3), 1756–1766. doi: 10.1109/TII.2018.2849005

Zakari, A., Musa, M., Bekaroo, G., Bala, S. A., Hashem, I. A. T., & Hakak, S. (2019). IPv4 and IPv6 Protocols: A Comparative Performance Study. *IEEE ICSGRC 2019 10th Control and System Graduate Research Colloquium, Proceeding*, 1–4. doi: 10.1109/ICSGRC.2019.8837050