Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11424
e-ISSN : 2541-2019
p-ISSN : 2541-044X

Comparative analysis of software defined network performance and conventional based on latency parameters

Raja Pahlevi Harahap[1)*], Deci Irmayani[2)], Rahma Muti'ah[3)]
[1)2)3)]Universitas Labuhanbatu, Indonesia
[1)]rajaoppo06@gmail.com, [2)]deacyirmayani@gmail.com, [3)]rmuthea5@gmail.com

Submitted : Apr 28, 2022 | **Accepted** : Apr 30, 2022 | **Published** : May 2, 2022

Abstract: Software Defined Network (SDN) offers convenience in managing network devices by simply setting the software control plane so as to reduce the complexity of network configuration. However, the ease of management offered is not always accompanied by an increase in network performance when compared to conventional network architectures. This study will perform a performance comparison analysis between networks with SDN architecture (OpenFlow) and conventional architecture. The research methods applied are: topology implementation on Mininet, network simulation and data collection, data analysis, and comparative analysis. Network performance testing is carried out based on latency parameters that are simulated in one subnet using the Mininet emulator. From the results of the latency test, it is found that the average latency on the SDN network is 0.119ms, while the average latency on the conventional network is 0.09588ms. Based on these results, it can be concluded that the network topology that has better performance based on latency parameters is a conventional network topology with an average latency value of 0.09588ms.

Keywords: Control plane; Latency; Mininet; Network; SDN;

INTRODUCTION

Software Define Networking (SDN) is a new approach to designing, building and managing computer networks by separating the control plane and data plane which is different from conventional network networks where the control plane and data plane are still combined.

The main concept of SDN is network centralization where, where all the settings are in the control plane, especially on the data plane platform that uses an abstraction layer for all network devices on the SDN architecture. So that it is easy to manage network devices by simply setting the control plane software and is able to reduce the complexity of network configuration. The most prominent protocol on SDN is Openflow (Ummah, 2016).

Openflow became the first standard as an interface in communication between the forwarding plane and control plane in SDN architecture where Openflow can configure network devices and choose the optimal path for traffic applications and controllers that allow software to run on various types of hardware (Anam & Adrian, 2017) . However, the advantages offered by the SDN network are not necessarily followed by an increase in performance when compared to conventional networks.

Research related to this topic has been reviewed by a number of previous researchers. Research conducted to compare the performance of SDN with conventional networks based on the parameters of jitter, delay, and throughput showed that SDN had better delay, jitter, and throughput results (Hernandez, Jimenez, Pranolo, & Rios, 2019). In another study, it was found that, by analyzing key network metrics including round-trip-time (RTT) and data transfer rate (DTR), the results show that the location of the controller has a demonstrable effect on network performance (Banjar, Pupatwibul, Braun, & Moulton, 2014). The SDN network has Quality of Services (QoS) results compared to conventional networks and the latency value produced by SDN is between 0.019 – 0.084ms (Nugroho, Irfan, & Faruq, 2019).

Based on the background that has been described, this study aims to perform a comparative analysis of the performance of the SDN network compared to conventional networks with latency parameters. The formulation of the problem in this research is between SDN and Conventional, which network topology is better and what are the results obtained.

*name of corresponding author

This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11424
e-ISSN : 2541-2019
p-ISSN : 2541-044X

LITERATURE REVIEW

Software Defined networking (SDN) is a computer networking approach that allows network administrators to dynamically program, control, modify, and manage network behavior through open interfaces and low-level functional abstractions. SDN is intended to address the fact that traditional networked static architectures do not support the dynamic and scalable compute and storage needs of modern computing environments such as data centers. This is done by separating the system that makes decisions about where the traffic is sent from the system that underlies the traffic to the chosen destination (Haleplidis et al., 2015). The advantages and conveniences offered by the SDN architecture are not necessarily followed by an increase in network performance when compared to conventional network architectures that use the TCP/IP protocol. SDN is commonly associated with the Openflow protocol (for remote communication with network plane elements to determine the path of network packets across network switches) since its second appearance in 2011 (Eissa, Bozed, & Younis, 2019).

Openflow is a protocol that functions in bridging communication between the Data plane Layer (routers, openflow switches) and the Control plane Layer (Controller) on the SDN architecture. Openflow allows direct access and manipulation of the data plane (routers, openflow switches). Openflow based on SDN architecture requires configuration of network devices every time, service policies can change, so as to reduce network configuration errors and inconsistent policies (Heryanto & Afrilia, 2016). In addition, openflow allows switches from different vendors - often each with their own proprietary interface and scripting language - to be managed remotely using a single open protocol. The inventors of the protocol considered openflow as an enabler of the software-defined SDN network (IEEE, 2017).

QoS is a technique for managing bandwidth, delay, jitter, and packet loss for flows in a network. The purpose of the QoS mechanism is to influence at least one of the four basic QoS parameters that have been determined. Meanwhile, according to Ningsih et al, Quality of Service is the ability of a network to provide better services for traffic services that pass through it. QoS is an end-to-end architectural system and is not a feature of the network. The purpose of QoS is to meet the needs of different services, which use the same infrastructure (Iskandar & Hidayat, 2015).

Latency is the time it takes for data to travel the distance from origin to destination. Latency can be affected by distance, physical media, congestion or also long processing times (Gallenmüller, Naab, Adam, & Carle, 2020). Latency is divided into several categories as follows (ETSI, 2018):

Table 1. Latency Category

Latency Category	Delay Quantity
Very Good	<150ms
Good	150 s/d 300ms
Medium	300 s/d 450ms
Bad	>450ms

$$\text{Average latency} = \frac{Total\ Latency}{Total\ packages\ received} \quad (1)$$

METHOD

This research was carried out from November 2021 to February 2022. The research location was carried out at the Computer Network Laboratory, Labuhanbatu University. The data source is obtained from the simulation results of SDN and Conventional networks using mininet which is repeated 25 times for each network topology. Figure 1 is an illustration of the research stages.

*name of corresponding author

This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11424
e-ISSN : 2541-2019
p-ISSN : 2541-044X

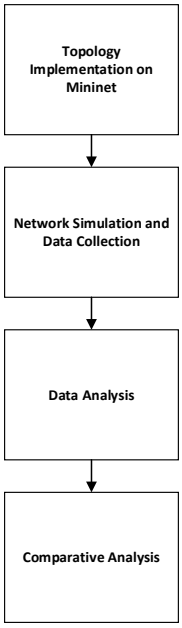

Fig. 1 Research Stages

The system built in this research is based on a simulation using a Mininet emulator. For SDN architecture, the controller will be connected directly to the Mininet emulator which consists of two interconnected OpenFlow switches. Each OpenFlow switch will be connected to two hosts respectively. As for the conventional architecture, the Controller is not used. The scope of the system represents the network that is still in 1 subnet.

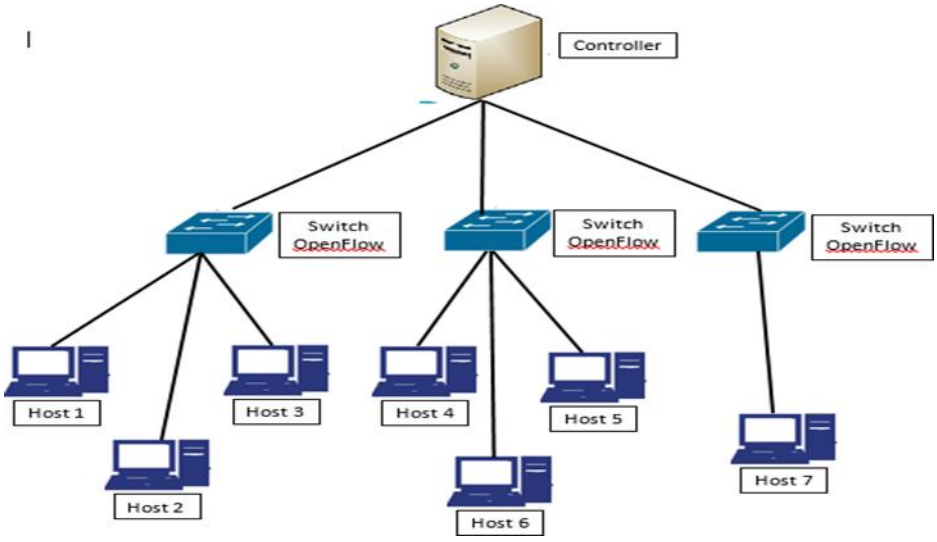

Fig. 2 SDN Network Topology

*name of corresponding author

This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11424
e-ISSN : 2541-2019
p-ISSN : 2541-044X

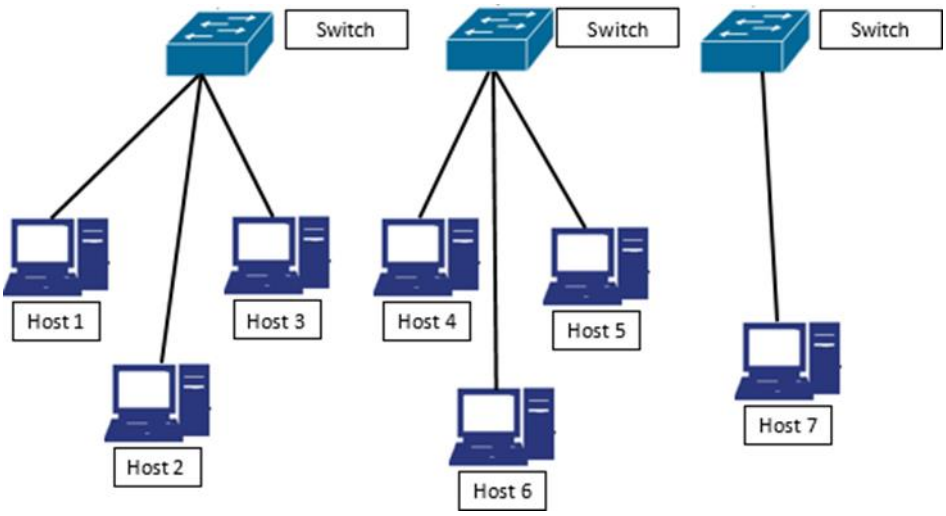

Fig 3. Conventional Network Topology

RESULT

In the test, measurements of latency were carried out by sending ICMP packets by paying attention to changes in packet size, namely 64, 128, 512, 1024, 2048, and 4096 bytes. The test is carried out using the ping tool with the command scheme given in Figure 4. In this scheme, an interval size of 10 is used with a total packet that is generated randomly. As for the packet size, it depends on the size of the ICMP packet to be sent.

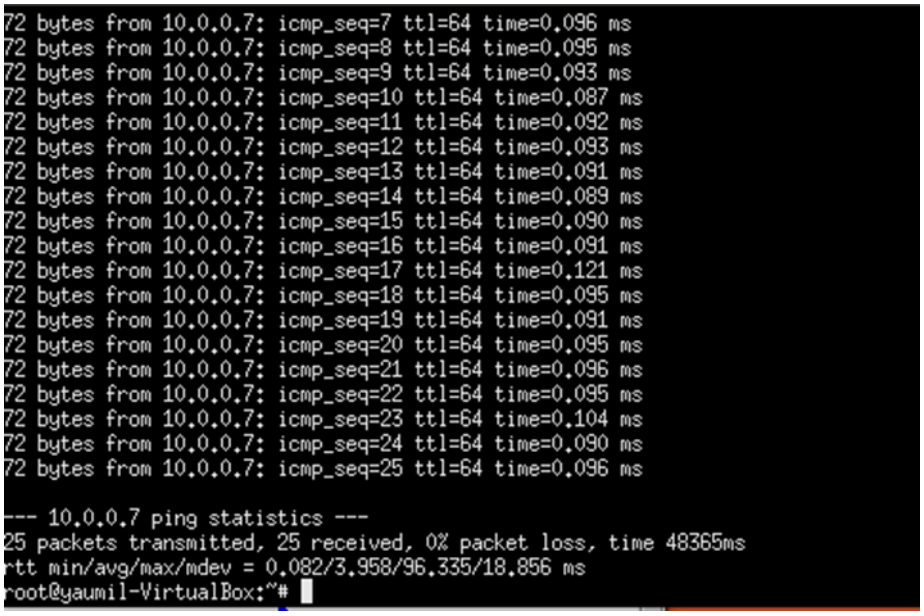

Fig. 4 SDN Architecture Latency Test

*name of corresponding author

This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

Sinkron : Jurnal dan Penelitian Teknik Informatika
Volume 6, Number 2, April 2022
DOI : https://doi.org/10.33395/sinkron.v7i2.11424
e-ISSN : 2541-2019
p-ISSN : 2541-044X

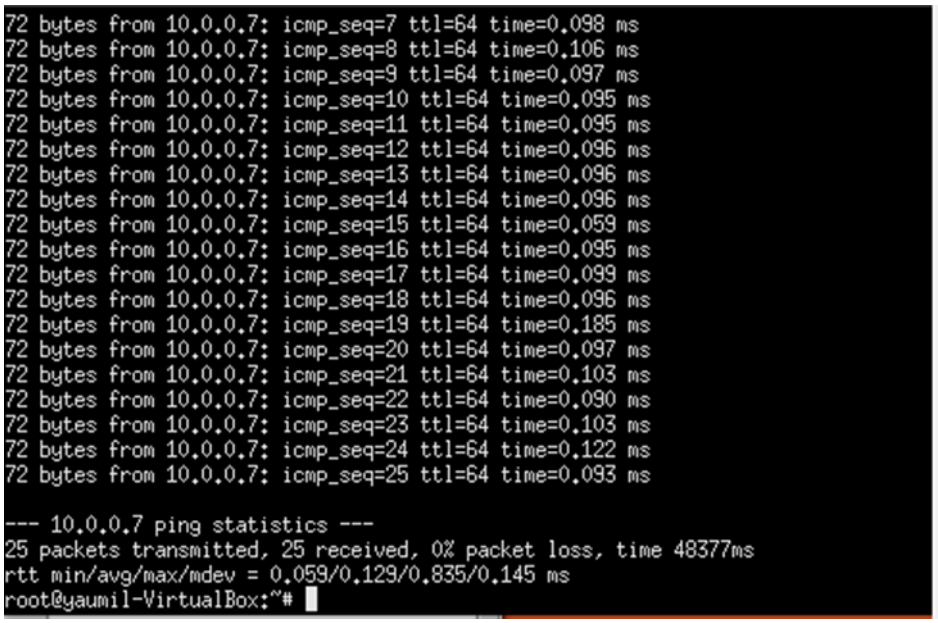

Fig. 5 Conventional Architecture Latency Test

From the results of the tests carried out, the latency test values for SDN and conventional architectures are presented in the graph in Figure 6.

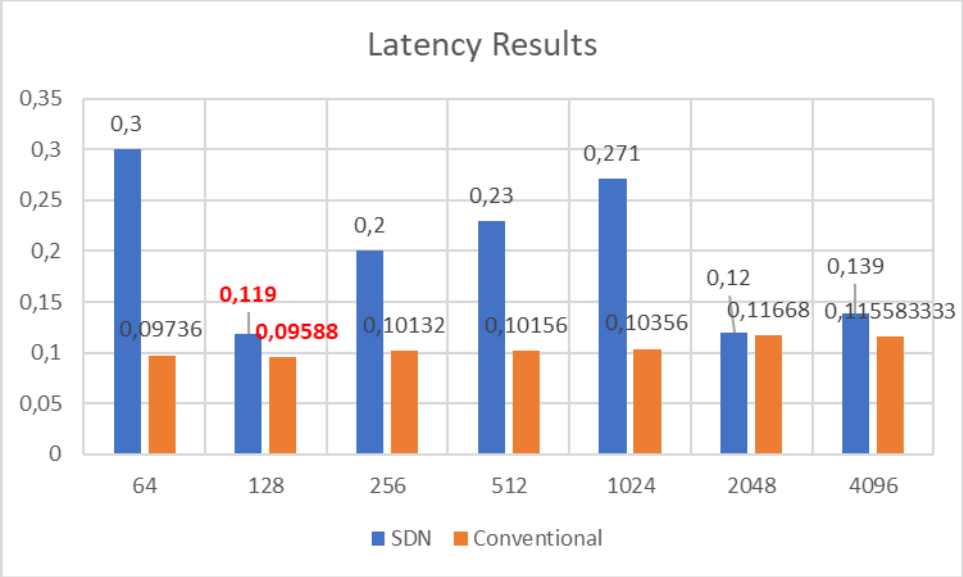

Fig. 6 Latency Test Results

From these results, it is known that the best latency value for SDN and conventional network architectures is obtained when the ICMP packet size is 128 bytes. This is because at the size of 128 bytes the smallest latency value is obtained following the formula (1).

DISCUSSIONS

Tests by transmitting ICMP packets which aim to measure the latency value for both network architectures, show that there is a significant additional latency value for the SDN architecture. This is because the communication between the switch and the controller in handling packets to be sent or received. If a comparison is made between the two architectures, the results show that the conventional architecture is still better than the SDN architecture with an average delay value of 0.09588. This is because in the SDN architecture every decision to transmit packets is made by the controller. This is because if the first echo request packet requires more time

*name of corresponding author

This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

for the address resolution process by the Controller and at the same time the Controller enters flow entries into the flow tables of the switch so that the next packet forwarding does not require the resolving address process again.

The high latency value caused by communication between controllers and switches occurs in 2 stages, namely packet arrival and forwarding table updates. In the first stage, when receiving the first ICMP packet. The switch sends a packet_in message to the Controller containing the metadata and the first 128B of the packet. In the second stage, the controller sends a flow_mods message to enter or update flow entries into the flow tables of the switch. This stage is called inbound and outbound latency. As the test results show that the first ICMP packet of each packet size sent always shows a very large latency value for the SDN network architecture.

CONCLUSION

Based on the results of research and network performance testing of SDN and conventional network architectures that have been carried out, it can be concluded that a better network topology based on latency parameters is a conventional network topology with an average latency value of 0.09588ms.

Some suggestions that can be considered for future research are, testing can involve more than one network subnet, using more than two controllers with a more complex network topology..

REFERENCES

Anam, K., & Adrian, R. (2017). Analisis Performa Jaringan Software Defined Network Berdasarkan Penggunaan Cost Pada Protokol Ruting Open Shortest Path First. *CiITEE*, 1–8.

Banjar, A., Pupatwibul, P., Braun, R., & Moulton, B. (2014). *Analysing the performance of the OpenFlow standard for software-defined networking using the OMNeT++ network simulator*. https://doi.org/10.1109/APCASE.2014.6924467

Eissa, H. A., Bozed, K. A., & Younis, H. (2019). Software Defined Networking. *2019 19th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, 620–625. https://doi.org/10.1109/STA.2019.8717234

ETSI. (2018). Achieving the lowest latency for delay-sensitive traffic. Retrieved April 15, 2022, from etsi.org website: https://www.etsi.org/newsroom/blogs/entry/achieving-the-lowest-latency-for-delay-sensitive-traffic

Gallenmüller, S., Naab, J., Adam, I., & Carle, G. (2020). 5G QoS: Impact of Security Functions on Latency. *NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium*, 1–9. https://doi.org/10.1109/NOMS47738.2020.9110422

Haleplidis, E., Pentikousis, K., Denazis, S., Salim, J., Meyer, D., & Koufopavlou, O. (2015). RFC 7426: Software-Defined Networking (SDN): Layers and Architecture Terminology. *IRTF*.

Hernandez, L., Jimenez, G., Pranolo, A., & Rios, C. U. (2019). Comparative Performance Analysis Between Software-Defined Networks and Conventional IP Networks. *2019 5th International Conference on Science in Information Technology (ICSITech)*, 235–240. https://doi.org/10.1109/ICSITech46713.2019.8987493

Heryanto, A., & Afrilia. (2016). Software Defined Network Menggunakan Simulator. *Kntia*, (33), 5–8. Retrieved from https://www.seminar.ilkom.unsri.ac.id/index.php/kntia/article/view/1142

IEEE. (2017). Overview of RFC7426: SDN Layers and Architecture Terminology. Retrieved from sdn.ieee.org website: https://sdn.ieee.org/newsletter/september-2017/overview-of-rfc7426-sdn-layers-and-architecture-terminology

Iskandar, I., & Hidayat, A. (2015). Analisa Quality of Service (QoS) Jaringan Internet Kampus (Studi Kasus: UIN Suska Riau). *Jurnal CoreIT*, *1*(2), 67–76.

Nugroho, H., Irfan, M., & Faruq, A. (2019). Software Defined Networks: a Comparative Study and Quality of Services Evaluation. *Scientific Journal of Informatics*, *6*, 181–192. https://doi.org/10.15294/sji.v6i2.20585

Ummah, I. (2016). Perancangan Simulasi Jaringan Virtual Berbasis Software-Define Networking. *Indonesian Journal on Computing (Indo-JC)*, *1*(1), 95–106. https://doi.org/10.21108/indojc.2016.1.1.20

*name of corresponding author

This is an Creative Commons License This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.