Analysis of ONOS Clustering Performance on Software Defined Network

Kinanthi Audina Septian, Istikmal, Ishak Ginting
School of Electrical Engineering, Telkom University, Bandung, Indonesia
E-mail: kinanthias@student.telkomuniversity.ac.id

2021 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS) | 978-1-6654-2035-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/IoTaIS53735.2021.9628659

***Abstract*—The increasing need for service providers makes the performance of network management must also be improved. In this study, the author builds a clustering controller using the Open Network Operating System (ONOS) controller on an Software Defined Network (SDN) network. ONOS has a cluster feature that allows it to distribute traffic loads to other controllers to increase performance. The performance of clustering controller is compared with single controller and multi controller without clustering by measuring Quality of Service (QoS) on delay, jitter, and throughput parameters for data, voice, and video services. The test is done by adding background traffic. After being tested and analyzed, the performance of the clustering controller has the most superior QoS with an average delay in data services of 0.047026667 ms, jitter of 0.015414943 ms, throughput of 37.56360736 kbit/s. In the voice service, the delay value is 0.03909 ms, the jitter is 0.01828 ms, the throughput is 72.79049 kbit/s. In the video service, the delay value is 0,12072 ms, the jitter is 0,042013 ms, and the throughput is 5314,059 kbit/s. While in single and multi controller the average value obtained is not much different from the clustering controller. This is due to the single controller architecture, the network control center is only centered on 1 controller, as well as multi controllers, so that it affects the performance of service distribution.**

Keywords: ***Software Defined Network, ONOS, clustering controller.***

I. INTRODUCTION

As network traffic increases, it can result in decreased performance on a single controller. Meanwhile, if the single controller fails, the switch cannot plan the routing for the newly arrived packets, which will affect network communication. In this case, the use of multi controllers can be considered in order to achieve scalability and efficiency. The existence of multiple controllers will consequently bring challenges to consistency, reliability and load balancing. From these consequences, clustering controller is present as a solution

Previously, research focused on multi controller surveys concluded that multi controller could be a solution for optimizing large-scale SDN networks [1]. In research [2] apply a clustering controller. This research using Open Daylight controller. The results of this study indicate that the performance of the clustering controller of OpenDaylight.

In this study, the clustering controller consists of 3 ONOS controllers to be able to stabilize and improve network performance and compare network performance without and with cluster. Parameters measured are throughput, delay, and jitter. Performance testing using D-ITG (Distributed Internet Traffic Generator) software.

II. BASIC THEORY

2.1 Software Defined Network

The basic concept of SDN is to separating between the control plane and the data plane [3]. Control Plane is in charge of controlling the distribution of packets in the network. The Data Plane is the component responsible for forwarding packets, parsing packet headers, managing QoS, and packet encapsulation. With the Openflow protocol, devices can connect to the controller and will automatically separate the data plane from the control plane.

2.2 ONOS

Open Network Operating System (ONOS) is an SDN operating system for Service Providers, targeting high availability, high performance, scalability, and abstraction to make it easy to build applications and services [4]. ONOS is written in Java and uses OSGi for functionality management. Every feature in ONOS is enabled via Apache Karaf (OSGi runtime). The main goal of ONOS is a service provider network and ONOS has been equipped with a GUI and complementary display to make it easier to control devices that support the Openflow protocol connected to it. ONOS supports real-time network configuration and control, eliminating the need to run and switch control protocols within the network. With ONOS controller, users can easily build network applications without changing the system of data plane.

2.3 Clustering Controller

Clustering controller is a machine that is used to manage other machines in the cluster. Clustering Controller distributes work to other machines to process and receive output from other machines. Cluster technology has the advantage that it can produce a system with a high level of reliability and a system that has a high level of availability or commonly called high availability. In the clustering controller architecture, each controller is connected and exchanged data to handle the network load, with 1 controller as master and 2 controllers as slaves[5] . The master controller is selected based on periodic polling technology, responsible for combining data from member controllers in 1 cluster and then distributing it, while the slave controller knows the network status and can become the master controller in case of failure[6]. In this study, clusters are managed by the Atomix distributed systems framework using the SWIM protocol. Atomix was a system where all controllers share information and separating memory from controllers and create a new module [7]. When an ONOS node is started, ONOS propagates its identity to all Atomix nodes, and the Atomix node rebroadcasts other known members to join the ONOS node.

With the clustering controller, it can reduce the time when configuring into the network, only by configuring one controller, other controllers in one cluster will be configured according to the command. If one of the controllers on the multi controller without clustering goes down, causing the network under the controller to not run, but different from the clustering controller, the load from the network will be handled by other controllers in 1 cluster [8].

2.4 D-ITG (Distributed Internet Traffic Generator) Software

DITG (Distributed Internet Traffic Generator) is a platform that can generate data traffic according to a pattern that is determined by the time between packet-to-packet delivery (IDT) and the stochastic packet size. DITG can measure one-way delay and round trip time, evaluate packet loss, and measure jitter and throughput [9].

III. System Design and Implementation

3.1 System Design

Network performance testing is carried out on three topologies that have been designed, namely a mesh topology with 8 switches and 8 hosts connected to each other. In research [10], when the number of switches is more than 300 using a single controller, the control channel is at risk of being disconnected so a clustering controller is designed with the aim of increasing network performance. In cluster mode, three controllers can accommodate up to 1000 switches. In this test, there are limitations on the testing tool so that it only focuses on comparing performance, not focusing on the capacity of the control channel between single controllers, multi controllers without clustering and clustering controllers.

The topology design consists of 8 switches connected to each host.:

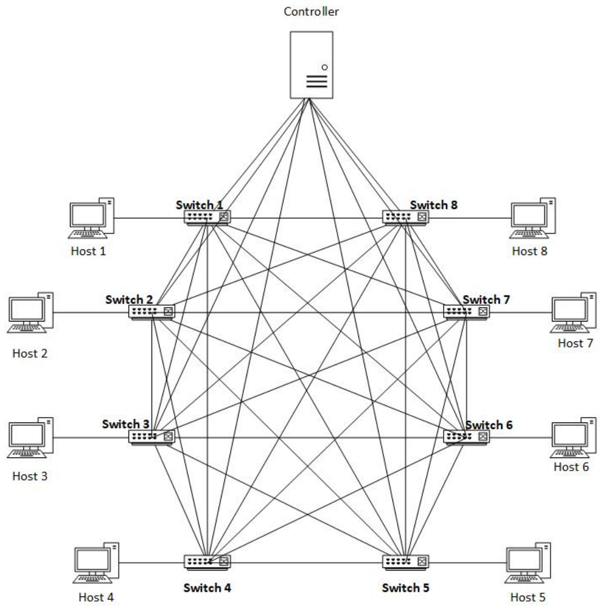


Figure 3.1 Single Controller Topology

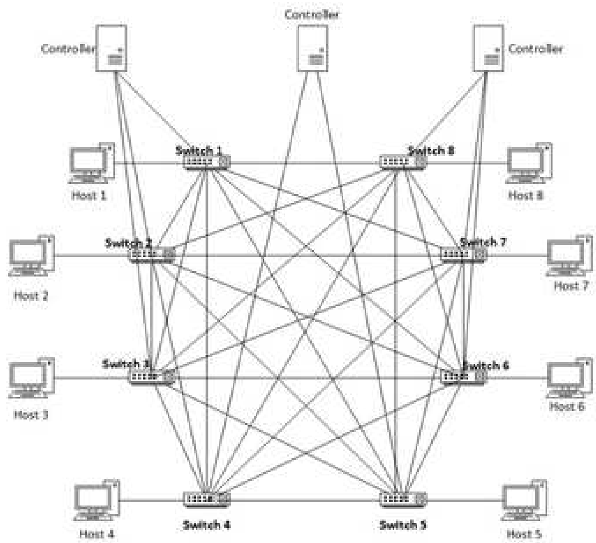


Figure 3.2 Multi Controller without Clustering Topology

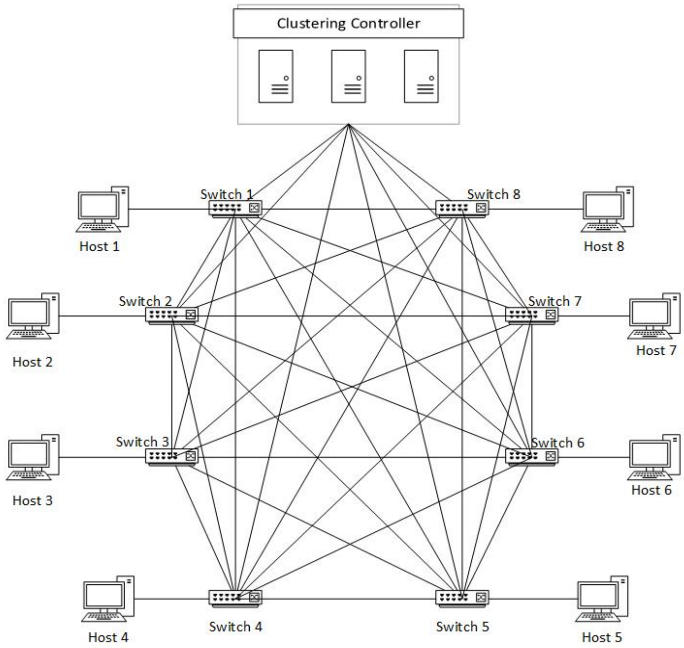


Figure 3.3 Clustering Controller Topology

Figure 3.1 shows the topology on a single controller with 8 hosts and 8 switches, which is connected to each other. Figure 3.2 shows a multi controller topology consisting of 3 ONOS controllers but not in one cluster. Switches 1-3 are connected to ONOS 1, switches 4 and 5 are connected to ONOS 2, while switches 6-8 are connected to ONOS 3. Figure 3.3 shows the topology of multi controllers in one cluster. In one cluster consists of 3 ONOS controllers. All switches are connected to the clustering controller

3.2 Block Diagram

Figure 3.4 illustrates the workflow, starting with a literature study, the author explores the theory, and concepts of the system tools used, journals or other supporting references. Then start the hardware installation consisting of the ONOS controller, Mininet emulator and D-ITG. If the installation is not successful, it will be reinstalled until all hardware and software can be installed. The next step is to design the topology on Mininet and configure the ONOS clustering. After successfully configured, it will proceed with system testing and parameter measurement.

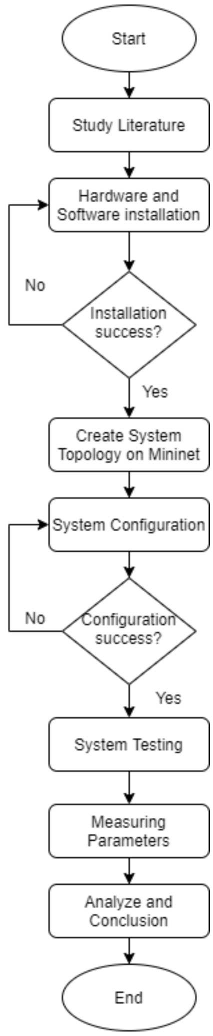


Figure 3.4 Research Flow

3.3 Testing Scenarios

Network performance testing is carried out on three topologies that have been designed, namely the mesh topology. The test was carried out with an end-to-end QoS scenario using D-ITG software to determine the parameters of throughput, delay, and jitter. Parameter testing is done by sending data, voice, and video traffic using the UDP protocol with background traffic of 100Mb, 200Mb, 300Mb, 400Mb, and 500Mb. QoS testing will be tested 30 times each for 20 seconds according to data collection standards. Furthermore, the results will be compared between single controller, multi controller without clustering, and clustering controller. There are three reasons of choosing UDP as the background traffic's protocol : [11]

1. UDP provides more precisely of the actual load on the network estimate

2. Acquired result with UDP represent a top limit to the throughput performance
3. The 802.11 MAC layer performs retransmitting of misplaced or undermined packets so that TCP is not affected by packet loss

IV. RESULT AND ANALYSIS

4.1 Average test scores of single controller, multi controller without clustering and clustering controller in Data Services

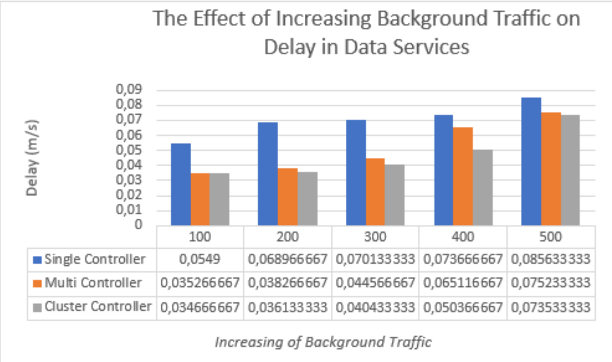


Figure 4.1.1 The Effect of Increased Background Traffic on Data Service Delays.

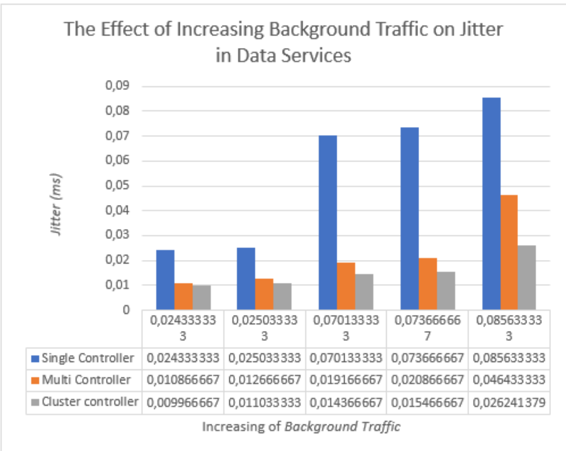


Figure 4.1.2 The Effect of Increased Background Traffic Data Services Jitters

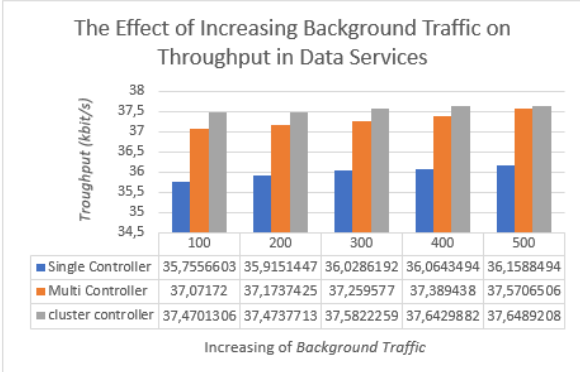


Figure 4.1.3 The Effect of Increased Background Traffic Data Services Throughput

Table 4.1.1, 4.1.2, and 4.1.3, are the average result of QoS testing on data services. The more background traffic increases, the greater the delay, jitter, and throughput generated. The clustering controller has the lowest delay and jitter values, while the throughput has the highest value, which means that the QoS value for clustering controllers can be said to be superior to single controllers and multi controllers without clustering. This is because the control center on the clustering controller is handled by 3 controllers, so the value obtained is not much different from the multi controller without clustering.

4.2 Average test scores of single controller, multi controller without clustering and clustering controller in Voice Services

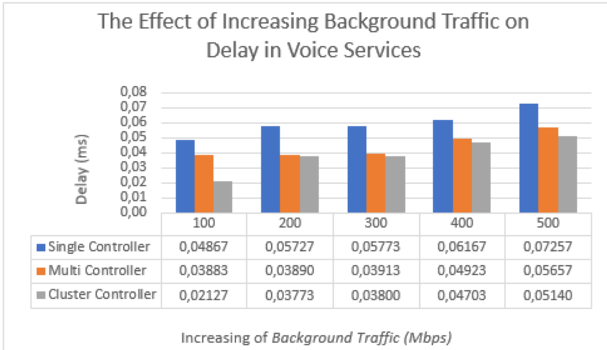
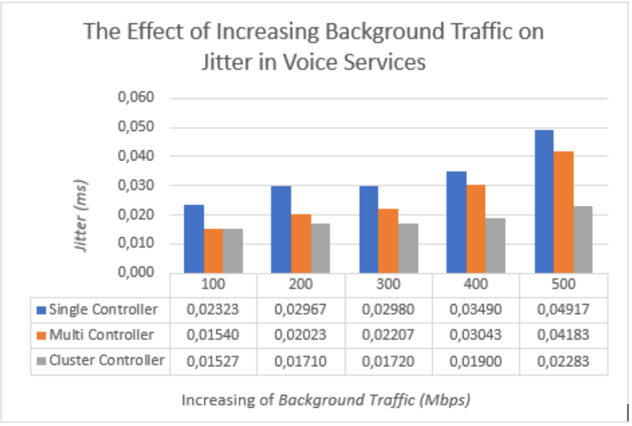
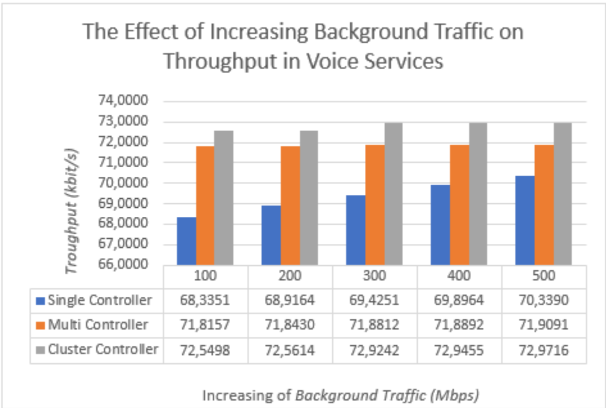


Figure 4.2.1 The Effect of Increased Background Traffic on Voice Services Delays



	100	200	300	400	500
Single Controller	0,02323	0,02967	0,02980	0,03490	0,04917
Multi Controller	0,01540	0,02023	0,02207	0,03043	0,04183
Cluster Controller	0,01527	0,01710	0,01720	0,01900	0,02283

Figure 4.2.2 The Effect of Increased Background Traffic on Voice Services Jitters

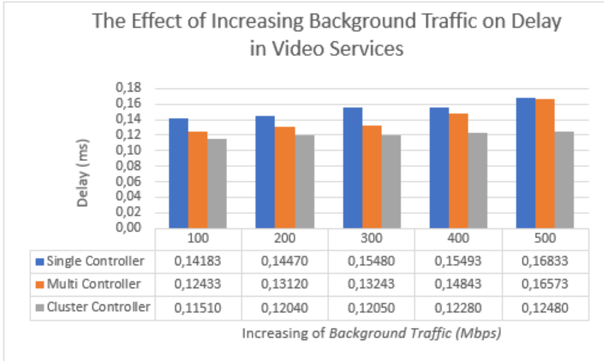


	100	200	300	400	500
Single Controller	68,3351	68,9164	69,4251	69,8964	70,3390
Multi Controller	71,8157	71,8430	71,8812	71,8892	71,9091
Cluster Controller	72,5498	72,5614	72,9242	72,9455	72,9716

Figure 4.2.3 The Effect of Increased Background Traffic on Voice Services Throughputs

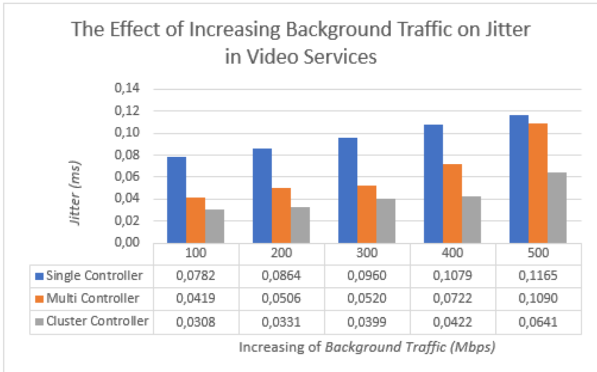
Figure 4.2.1, 4.2.2, and 4.2.3 are the average result of QoS testing on voice services. Just like data service analysis, the more background traffic, the greater the delay, jitter, and throughput generated. The clustering controller has the lowest delay and jitter values, while the throughput has the highest value, which means that the QoS value for clustering controllers can be said to be superior to single controllers and multi controllers without clustering. This is because the control center on the clustering controller is handled by 3 controllers, so the value obtained is not much different from the multi controller without clustering.

4.3 Average test scores of single controller, multi controller without clustering and clustering controller in Video Services



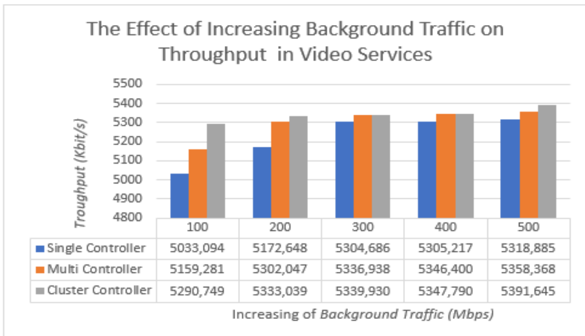
	100	200	300	400	500
Single Controller	0,14183	0,14470	0,15480	0,15493	0,16833
Multi Controller	0,12433	0,13120	0,13243	0,14843	0,16573
Cluster Controller	0,11510	0,12040	0,12050	0,12280	0,12480

Figure 4.3.1 The Effect of Increased Background Traffic on Video Services Delays



	100	200	300	400	500
Single Controller	0,0782	0,0864	0,0960	0,1079	0,1165
Multi Controller	0,0419	0,0506	0,0520	0,0722	0,1090
Cluster Controller	0,0308	0,0331	0,0399	0,0422	0,0641

Figure 4.3 2 The Effect of Increased Background Traffic on Video Services Jitters



	100	200	300	400	500
Single Controller	5033,094	5172,648	5304,686	5305,217	5318,885
Multi Controller	5159,281	5302,047	5336,938	5346,400	5358,368
Cluster Controller	5290,749	5333,039	5339,930	5347,790	5391,645

Figure 4.3.3 The Effect of Increasing Background Traffic on Video Services Throughputs

Figure 4.3.1, 4.3.2, and 4.3.3 are the average result of QoS testing on data services. Not much different from data and voice services, the more background traffic increases, the greater the delay, jitter, and throughput generated. The clustering controller has the lowest delay and jitter values, while the throughput has the highest value, which means that the QoS value for clustering controllers can be said to be superior to single controllers and multi controllers without clustering. This is because the control center on the clustering controller is handled by 3 controllers, so the value obtained is not much different from the multi controller without clustering.

V. CONCLUSION

The performance of the clustering controller architecture was better than single controllers and multi controllers without clustering. Clustering controllers were better able to guaranteed the delivery of data, voice, and video services on a large scale. This was because in a multi controller architecture without clustering, each controller was not connected each other and there was no process of exchanging data, so data requests from the data plane were only handled by each controller, while in the clustering controller architecture the control center on the clustering controller was handled. by 3 controllers, so the value obtained was not much different from multi controller without clustering. In the delay parameter, the clustering controller had the lowest value for data, voice, and video services, which means that the lower the delay value, the better the network. The jitter parameter also shows that it had the lowest value on the clustering controller for data, voice, and video services, which means that the lower the jitter value, the better the network could be. In the throughput parameter, the results of the clustering controller test had the highest throughput values for data, voice, and video services, which means that the higher the throughput performance value, the better the network.

VI. REFERENCES

[1] T. Hu, Z. Guo, T. Baker dan J. Lan, “Multi-controller Based Software-Defined Networking: A Survey,” *Digital Object Identifier IEEE Access,* 2017.

[2] E. Hernandez, “Dipartimento di Elettronica , Informazione e Bioingegneria Implementation and Performance of a SDN Cluster- Controller Based on the OpenDayLight Framework,” 2016.

[3] P. Peter, M. Bilšák dan . J. Rouseková, “Impact of fragmentation threshold tuning on performance of voice service and background traffic in IEEE 802.11b WLANs,” *Proceedings of 20th International Conference Radioelektronika 2010,* pp. 75-78, 2010

[4] M. Gerola, . M. Santuari, F. Lucrezia, E. Salvadori, P. L. Ventre, S. Salsano and M. Campanella, "ICONA: A peer-to-peer approach for software defined wide area networks using ONOS," *Proceedings - European Workshop on Software-Defined Networks, EWSDN,* Vols. 2016-October, pp. 27-42, 2017.

[5] A. S. Muqaddas, . P. Giaccone, A. Bianco dan G. Maier, “Inter-controller traffic to support consistency in ONOS clusters,” *IEEE Transactions on Network and Service Management,* vol. 14, no. 4, pp. 1018-1031, 2017.

[6] A. Abdelaziz, A. T. Fong, A. Gani, U. Garba, S. Khan, A. Akhunzada, H. Talebian dan K.-K. R. choo, “Distributed Controller Clustering in Software Defined Networks,” *Plos One,* vol. 12 no.4, no. 2017, pp. 1-19, 2017.

[7] S. Cho and S. Lee, "Architecture for SDN-Independent Gateway," pp. 76-81, 2019.

[8] Q. Ou, H. Chen, and B. Zou, “Controller Cluster Based Interconnecting for Multi-domain SDN Networks,” *Proc. - 2017 Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov. CyberC 2017*, vol. 2018-Janua, pp. 482–485, 2017, doi: 10.1109/CyberC.2017.41.

[9] S. Avallone, S. Guadagno, A. Pescape, G. Vente dan D. Emma, “D-ITG Distributed Internet Traffic Generator,” *Conference Paper,* 2004.

[10] G. S. C. T. Center, SDNCTC, “ONOS Controller Performance Test Report,” SDNCTC, 2016.

[11] P. Peter, M. Bilšák dan . J. Rouseková, “Impact of fragmentation threshold tuning on performance of voice service and background traffic in IEEE 802.11b WLANs,” *Proceedings of 20th International Conference Radioelektronika 2010,* pp. 75-78, 2010.