

Implementasi Dan Analisis Load Balancing Per Connection Classifier Untuk Radius Server Pada Jaringan Software Defined Network Menggunakan Onos Controller

Implementation And Analysis Of Load Balancing Per Connection Classifier For Radius Server On Software Defined Network Using Onos Controller

1st Mohamad Nur Firdaus
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
nfirdaus@student.telkomuniversity.ac.id

2nd Rohmat Tulloh
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
rohmatth@telkomuniversity.ac.id
indrarini

3rd Indrarini Dyah Irawai
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia
taftazani@telkomuniversity.ac.id

Abstrak—Jaringan telekomunikasi merupakan hal yang paling dibutuhkan oleh manusia. Penggunaan jaringan telekomunikasi bisa dimanfaatkan untuk berbagai keperluan. Pada tahun 2019 hingga 2020 penggunaan jaringan telekomunikasi di Indonesia meningkat sekitar 8,9 %. Hal ini menyebabkan penggunaan akses jaringan telekomunikasi semakin berat. Salah satu cara untuk mengoptimalkan performansi jaringan dengan mengimplementasikan sistem *Load Balance* dan *Radius Server Software Defined Network*. Pada Proyek Akhir ini dilakukan sebuah implementasi sistem *Load Balancing* dan *Radius Server* pada jaringan konvensional serta pada jaringan SDN. Metode sistem *Load Balancing* menggunakan metode PCC dengan menggabungkan dua *traffic* dari ISP Telkomsel dan StarNet. *Radius Server* diimplementasikan pada fitur *userman*, sistem ini berfungsi untuk otentikasi *login* dan manajemen *bandwidth user* di jaringan. Pengukuran QoS pada setiap jenis jaringan dilakukan dengan menggunakan aplikasi Wireshark. Hasil pengujian menunjukan QoS pada Proyek Akhir ini memiliki hasil yang baik berdasarkan pada standard *Telecommunication and Internet Protocol Harmonization Over Network (TIPHON)* dengan pengukuran jumlah *capture packet* dimulai dari 2000, 4000, 6000, 8000 dan 10000. Protokol yang diukur dalam penelitian ini adalah protokol UDP dan TCP. Nilai *delay* terkecil ditunjukan oleh sistem *Load Balancing* dan *Radius Server* dengan rata-rata *delay* 0.112876218 ms. Nilai rata-rata *jitter* terkecil dengan sistem tersebut adalah 0.000046 ms.

Nilai *packet loss* sebesar 0.0135% dan nilai *throughput* pada sistem yang sama yaitu sebesar 56.2402 Mbps pada pengujian protokol TCP.

Kata kunci—*redudansi, performansi, Load Balance, traffic, Radius Server, Software Defined Network, QoS*

Abstract—*Telecommunications network is the thing that is most needed by humans. The use of telecommunications networks can be used for various purposes. From 2019 to 2020 the use of telecommunication networks in Indonesia increased by around 8.9%. This causes the use of telecommunication network access is getting heavier. One way to optimize network performance is to implement Load Balance and Radius Server Software Defined Network systems. In this final project, an implementation of Load Balancing and Radius Server systems is carried out on conventional networks and on SDN networks. The Load Balancing system method uses the PCC method by combining two traffic from ISP Telkomsel and StarNet. Radius Server is implemented on the userman feature, this system functions for login authentication and user bandwidth management on the network. QoS measurements on each type of network are carried out using the Wireshark application. The test results show that the QoS in this final project has good results based on the Telecommunication and Internet Protocol Harmonization Over Network (TIPHON) standard with measurement of the number of packet capture starting*

from 2000, 4000, 6000, 8000 and 10000. The protocols measured in this study are protocols. UDP and TCP. The smallest delay value is shown by the Load Balancing and Radius Server system with an average delay of 0.112876218 ms. The average value of the smallest jitter with the system is 0.000046 ms. The packet loss value is 0.0135% and the throughput value on the same system is 56.2402 Mbps in the TCP protocol test.

Keywords—*redundancy, performance, Load Balance, traffic, Radius Server, Software Defined Network, QoS.*

I. PENDAHULUAN

A. Latar Belakang

Infrastruktur jaringan telekomunikasi berkembang sangat cepat dan kompleks dengan berbagai vendor penyedia perangkat jaringan. Sebuah jaringan yang kompleks akan menimbulkan data traffic yang besar serta akan membuat jaringan menjadi *overload*. Semua itu menyebabkan penurunan performansi pada layanan jaringan. *Software Defined Network (SDN)* adalah konsep baru dalam mengontrol, mengimplementasi serta mengelola suatu jaringan yang mendukung kebutuhan dan inovasi di bidang telekomunikasi yang semakin lama semakin berkembang dan kompleks. Konsep dasar dari SDN yaitu dengan memisahkan antara *control plane* dan *data plane*. Beberapa penelitian mengenai SDN sudah dilakukan seperti implementasi SDN pada perangkat raspberry pi 3[1] dan penelitian protokol routing BGP (*Border Gateway Protocol*) yang di implementasikan pada jaringan SDN[2]. Penelitian tersebut dilakukan untuk mengukur performansi jaringan berdasarkan nilai QoS.

Penggunaan jaringan telekomunikasi menurut Badan Pusat Statistik (BPS) dari populasi Indonesia tahun 2019 berjumlah 266.911.900 juta, sehingga pengguna internet Indonesia diperkirakan sebanyak 196,7 juta pengguna. Jumlah tersebut naik dari 171 juta di tahun 2019 dengan penetrasi 73,7 persen atau naik sekitar 8,9 persen atau sekitar 25,5 juta pengguna[3]. Kebutuhan *bandwidth* akan naik seiring bertambahnya pengguna jaringan telekomunikasi sesuai dengan data statistik tersebut. Performansi jaringan akan cenderung menurun jika dipakai secara masif. Dampak dari penurunan performa jaringan dapat membuat kualitas akses informasi menjadi lambat. Untuk mengoptimalkan jaringan diperlukan sistem yang dapat mengatur *data traffic*. Manajemen data traffic pada jaringan adalah suatu solusi untuk mengatasi jaringan *overload*.

Penelitian yang akan dilakukan untuk membuat penggunaan *traffic* pada jaringan lebih optimal dengan mengintegrasikan antara *Load Balancing*, *Radius Server* dan jaringan SDN. Simulasi awal akan dibuat sebuah sistem penggabungan dua ISP dengan metode *Load Balancing Per-Connection Classifier* pada jaringan konvensional dan SDN. Perancangan selanjutnya membuat radius server

yang berfungsi untuk membuat limitasi penggunaan jaringan. Tahap analisa dilakukan dengan membandingkan hasil simulasi perhitungan QoS pada jaringan konvensional dan jaringan yang sudah mengimplementasikan SDN. Target dari Proyek Akhir ini diharapkan dapat menganalisis QoS (*throughput, delay, jitter, packet loss*) berdasarkan standard *Telecommunications and Internet Protocol Harmonization Over Network (TIPHON)*.

B. Tujuan dan Manfaat

Adapun tujuan dari penulisan Proyek Akhir ini, sebagai berikut.

1. Mengetahui perbedaan topologi jaringan konvensional dengan jaringan *Software Defined Network (SDN)*.
2. Mensimulasikan skenario *Load Balancing* dengan metode *Per-Connection Classifier* dan *Radius Server* pada jaringan konvensional dan jaringan SDN.
3. Menganalisis skenario perbandingan QoS antara jaringan konvensional dan jaringan yang sudah mengimplementasikan SDN.
4. Menganalisis hasil perhitungan QoS berdasarkan standar nilai dari *Telecommunications and Internet Protocol Harmonization Over Network (TIPHON)*.

Manfaat dari penulisan Proyek Akhir ini, sebagai berikut.

1. Dapat merekomendasikan solusi permasalahan pada jaringan yang mempunyai *traffic* yang cukup tinggi.
2. Dapat membandingkan prinsip topologi pada jaringan konvensional dengan jaringan yang sudah menggunakan teknologi SDN.
3. Dapat mengimplementasikan metode *Load Balancing Per-Connection Classifier* serta limitasi menggunakan *Radius Server*.
4. Dapat menganalisis hasil simulasi perancangan dengan memperhatikan nilai QoS berdasarkan standar dari TIPHON.

C. Rumusan Masalah

Adapun rumusan masalah dari Proyek Akhir ini, sebagai berikut.

1. Bagaimanakah performansi *Load Balancing* dan *Radius Server* pada jaringan konvensional?
2. Bagaimana mengimplementasikan jaringan *Software Defined Network (SDN)* dengan menggunakan *ONOS Controller*?
3. Bagaimana menerapkan *Load Balancing* dengan metode *Per-Connection Classifier* dan manajemen *bandwidth* menggunakan *Radius Server* pada jaringan SDN?
4. Bagaimanakah hasil analisa QoS yang dihasilkan dari perbandingan jaringan konvensional dan jaringan SDN?

5. Apakah hasil analisa QoS perancangan dan simulasi masih berada pada nilai standar dari TPIHON?

D. Batasan Masalah

Adapun batasan masalah dari Proyek Akhir ini, sebagai berikut.

1. Menggunakan ONOS controller sebagai *control plane*.
2. Menggunakan mesin virtual sebagai *controller* dengan sistem operasi Linux Ubuntu 16.04.
3. Menggunakan *Load Balancing* dengan metode *Per Connection Classifier* dari dua ISP.
4. Menggunakan *Radius Server* untuk sistem autentikasi dan manajemen *bandwidth*.
5. Perangkat yang digunakan untuk implementasi jaringan ini adalah empat buah laptop, tiga *routerboard* mikrotik, satu *converter USB to Ethernet*, satu buah switch dan satu buah *controller (Virtual Machine)*.
6. Melakukan perbandingan nilai QoS pada jaringan konvensional dan jaringan *Software Defined Network (SDN)*.
7. Parameter QoS yang diukur adalah *Throughput, Jitter, Delay* dan *Packet Loss*.

E. Metodologi

Adapun metodologi pada penelitian Proyek Akhir ini, sebagai berikut.

1. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan literatur-literatur dan kajian-kajian yang berkaitan dengan permasalahan yang ada pada penelitian Proyek Akhir ini, baik berupa buku referensi, artikel, maupun *e-journal* yang berhubungan dengan perancangan sistem jaringan yang dibuat.

2. Perancangan Sistem

Perancangan dilakukan dengan membuat skema topologi jaringan konvensional dan jaringan yang sudah menggunakan *Software Defined Network (SDN)*.

3. Implementasi Perancangan

Implementasi sistem *Load Balancing* dan *Radius Server* pada jaringan konvensional dan jaringan *Software Defined Network (SDN)*.

4. Analisis Sistem

Analisis dilakukan dengan cara menganalisa perbandingan hasil perhitungan QoS pada skenario simulasi sistem *Load Balancing* dan *Radius Server* pada jaringan konvensional dan jaringan *Software Defined Network (SDN)*.

5. Penarikan Kesimpulan

Menyimpulkan hasil dari seluruh tahapan yang telah dilakukan dalam proses pengerjaan Proyek Akhir dengan berbagai masukan dan saran dari dosen pembimbing maka dapat diambil kesimpulan dari hasil yang telah dilakukan.

II. DASAR TEORI

A. Software Defined Network

Software Defined Network (SDN) adalah konsep baru dalam mengontrol, mengimplementasi serta mengelola suatu jaringan yang mendukung kebutuhan dan inovasi di bidang telekomunikasi yg semakin lama semakin berkembang dan kompleks. Konsep dasar dari SDN yaitu dengan memisahkan antara *control planed* dan *data plane* [4]. SDN mampu membuat jaringan baik berskala kecil maupun berskala besar mampu terkontrol menggunakan 1 *controller* yang terpusat. Beberapa *controller* yang ada pada SDN diantaranya adalah POX, RYU, OpenDaylight, dan ONOS

B. Open Network Operating System

ONOS adalah sistem operasi jaringan SDN *open source* terkemuka untuk membangun solusi SDN generasi selanjutnya yang ditargetkan secara khusus di *Service Provider* dan *Mission Critical Network*. ONOS dibangun untuk memberikan ketersediaan tinggi (*High Availability*), *scale-out*, dan performansi kinerja jaringan yang dibutuhkan. *Controller* ini berbasis java [7].

C. OpenvSwitch

OpenvSwitch adalah *software* yang merepresentasikan multilayer virtual switch dibawah lisensi Apache 2.0. OpenvSwitch di desain untuk mendukung perubahan arsitektur jaringan dengan teknologi automasi jaringan menuju *programmatic extension*. OpenvSwitch juga mendukung *standard management interface* dan protokol seperti NetFlow, sFlow, IPFIX, RSPAN, CLI, LACP, 802.1ag. Selain itu, OpenvSwitch juga dapat digunakan sebagai *distribution* antar platform server yang berbeda [9].

D. Linux

Linux. Sistem operasi adalah perangkat lunak yang manajemen semua sumber daya perangkat keras pada sebuah perangkat komputasi seperti komputer dan ponsel pintar. Sederhananya, sistem operasi mengelola komunikasi antara perangkat lunak dan perangkat keras komputer. Tanpa sistem operasi, perangkat lunak tidak akan berfungsi[10].

E. Load Balancing

Load balancing adalah teknik untuk mendistribusikan beban trafik pada dua atau lebih jalur koneksi secara seimbang, agar trafik dapat berjalan optimal, memaksimalkan throughput,

memperkecil waktu tanggap dan menghindari *overload* pada salah satu jalur koneksi.

F. Per Connection Classifier

PCC merupakan metode yang menspesifikasikan suatu paket menuju *gateway* koneksi tertentu. PCC mengelompokkan trafik koneksi yang akan melalui atau keluar masuk router menjadi beberapa kelompok. Mikrotik akan mengingat-ingat *gateway* yang telah dilewati di awal trafik koneksi, sehingga pada paket-paket data selanjutnya yang masih berkaitan dengan paket data sebelumnya akan dilewatkan pada jalur *gateway* yang sama[12].

G. Radius Server

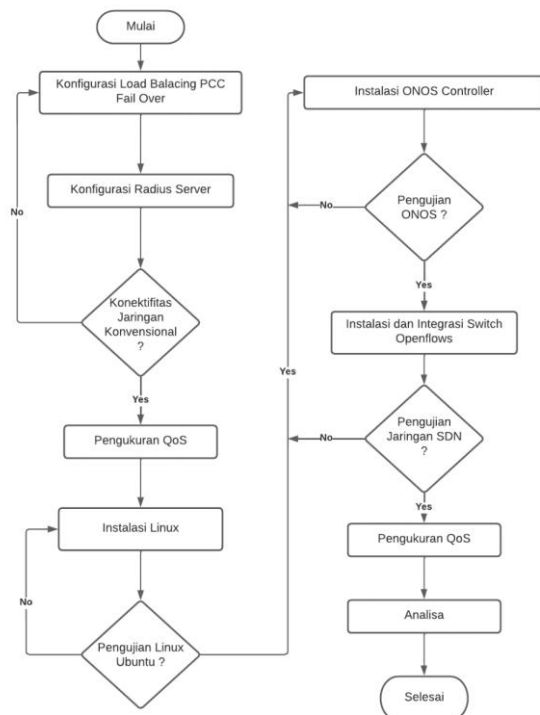
Radius adalah protokol autentifikasi akses server dan akutansi yang telah memperoleh dukungan luas. Server autentifikasi radius menjaga autentifikasi pengguna dan informasi akses jaringan. Klien akan berjalan ke akses server dan mengirim permintaan autentifikasi untuk *radius server*.

H. Quality of Services

Quality of Service (QoS) atau Kualitas layanan adalah metode pengukuran yang digunakan untuk menentukan kemampuan sebuah jaringan seperti; aplikasi jaringan, host atau router dengan tujuan memberikan network service yang lebih baik dan terencana sehingga dapat memenuhi kebutuhan suatu layanan[14].

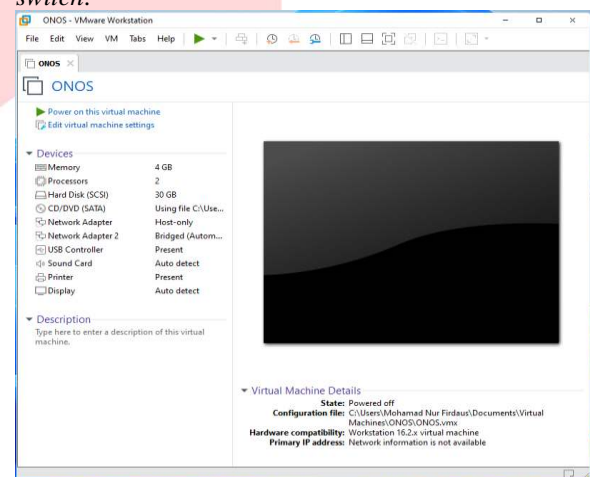
III. PERANCANGAN SISTEM

A. Metode Penelitian



Gambar 3. 1 Flowchart Metode Penelitian

Pada Gambar 3.1 diatas menunjukan flowchart untuk perancangan sistem pada Proyek Akhir, dimulai dari merancang skema jaringan konvensional. Konfigurasi *Load Balance* dengan menggabungkan dua ISP yang berbeda serta masing masing ISP mempunyai *bandwidth* yang berbeda. Selanjutnya mengkonfigurasi *Fail Over* yang berfungsi sebagai pengaturan redundansi jika salah satu ISP bermasalah. Langkah selanjutnya melakukan konfigurasi *Radius Server* yang berfungsi untuk limitasi *bandwidth* dan otentikasi login. Ketika skema jaringan konvensional sudah berjalan maka akan dilakukan pengukuran QoS. Tahap berikutnya melakukan instalasi Linux Ubuntu dan ONOS sebagai *controller* yang digunakan pada mesin virtual. Langkah berikutnya adalah melakukan instalasi *OpenFlow* pada setiap *routerboard* yang digunakan pada jaringan SDN kemudian dihubungkan dengan *controller* melalui *switch*.



Gambar 3. 2 Tampilan Mesin Virtual

Pada Gambar 3.2 menunjukan mesin virtual yang di instalasi Linux Ubuntu beserta *ONOS Controller*. Mesin virtual ini memiliki dua NIC, dimana setiap NIC mempunyai masing-masing fungsi yang berbeda. NIC yang pertama ini berfungsi untuk konektivitas antara komputer lokal dan Linux Ubuntu. Selain menjadi jembatan antara komputer lokal dan Linux NIC yang pertama digunakan sebagai *gateway* internet untuk melakukan instalasi *ONOS Controller*. NIC yang kedua bertipe *bridge* atau NIC yang terhubung secara langsung ke *interface ethernet* komputer. NIC ini berfungsi untuk menghubungkan *switch OpenFlow* ke *ONOS Controller*. Dua switch openflow terhubung dengan satu switch kemudian dihubungkan ke *ONOS Controller*. Proyek Akhir ini dilakukan terdapat 2 aplikasi yang diaktifkan pada *controller ONOS* yaitu *OpenFlow* dan *Forwarding*.

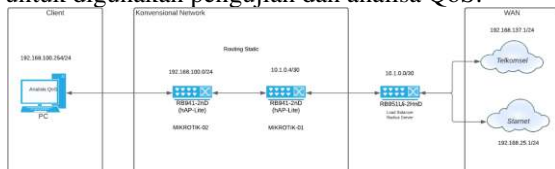
Tahap pengujian dimulai dari melihat hasil *throughput speedtest* setelah dilakukan *Load Balancing* dari kedua ISP. Pengujian yang selanjutnya adalah mencoba sistem *Radius Server* dengan mencoba login dengan dua user yang telah

dibuat. User pertama tidak diberikan limitasi *bandwidth* dan user yang kedua diberikan limitasi *bandwidth*. Tahap pengujian berikutnya mengumpulkan data dari pengukuran QoS pada jaringan konvensional dan jaringan SDN. Pengujian ini dilakukan dengan mengukur berdasarkan jenis protokol UDP dan TCP. Untuk pengukuran protokol UDP dilakukan dengan cara menayang *live streaming* video melalui youtube. Protokol TCP diukur dengan cara mendownload *file* dari internet. Pengukuran tersebut dilakukan dengan cara mengcapture paket dengan jumlah yang berbeda menggunakan aplikasi Wireshark. Tahap terakhir dari Proyek Akhir ini adalah menganalisa pengukuran QoS dari setiap skema jaringan yang dibuat dan dilakukan penarikan kesimpulan.

B. Topologi Jaringan

1. Konvensional

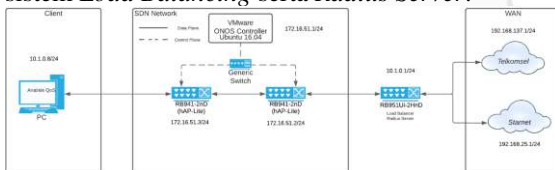
Perangkat yang digunakan yaitu tiga mikrotik routerboard, satu PC dengan satu buah smartphone untuk digunakan sebagai internet gateway untuk ISP kesatu, satu buah router untuk ISP kedua dan satu PC untuk digunakan pengujian dan analisa QoS.



Gambar 3. 3 Topologi Logic Konvensional

2. Software Defined Network

Perangkat yang digunakan pada perancangan topologi ini yaitu tiga buah *routerboard*. Dua *routerboard* difungsikan sebagai switch *OpenFlow* dan satu *routerboard* digunakan untuk menjalankan sistem *Load Balancing* serta *Radius Server*.



Gambar 3. 4 Topologi Logic SDN

IV. HASIL PENGUJIAN

A. Pengujian Load Balancing

Pengujian *Load Balancing* pada Proyek Akhir ini dilakukan dengan menguji *throughput bandwidth* download dan *upload* pada masing-masing ISP. ISP pertama memiliki *throughput* sebesar 22.3 Mbps untuk download dan 7.45 Mbps untuk *upload*. ISP kedua memiliki *throughput* sebesar 28.60 Mbps untuk *download* dan 29.17 Mbps untuk *upload*.

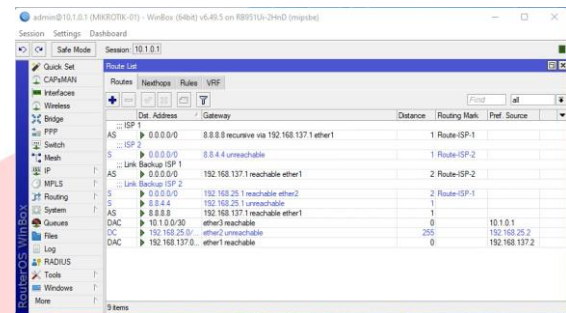
Tabel 4. 1 Perbandingan Throughput semua ISP

ISP	Throughput	
	Download	Upload
Telkomsel	22.7	7.45
StarNet	28.60	29.17

Load Balancing PCC	53.08	39.82
--------------------	-------	-------

B. Pengujian Fail Over

Pengujian pada tahap ini dilakukan dengan mencoba memutus salah satu *link* koneksi ISP. Pengujian dilakukan dengan skenario memutus *link* ISP kedua (StarNet). Pada tabel *routing* pengecekan koneksi pada ISP dinyatakan *unreachable* sehingga *link backup* akan diarahkan ke ISP pertama.



Gambar 4. 1 Fail Over ISP 2

C. Pengujian Radius Server

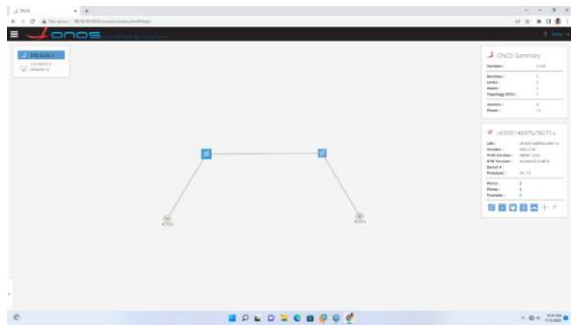
Pengujian *Radius Server* dilakukan dengan mencoba otentikasi login pada halaman web. Pada Proyek Akhir ini dilakukan otentikasi *login* dua *user*. Untuk *user* kedua hanya diberikan limitasi *bandwidth* sebesar 10 mbps. Halaman status *login* ditunjukkan oleh Gambar 4.2 untuk *session* user 10 mbps.



Gambar 4. 2 Topologi Logic SDN

D. Pengujian ONOS Controller

Pengujian *ONOS Controller* dilakukan dengan melihat koneksi atau *topology* dari setiap *switch* *OpenFlow*. Menu ini bisa diakses pada halaman *topology* yang tersedia pada tampilan *ONOS Controller*. Pada Gambar 4.3 dijelaskan bahwa pada Proyek Akhir ini menggunakan dua buah *switch* *OpenFlow* dan dua buah *host*.

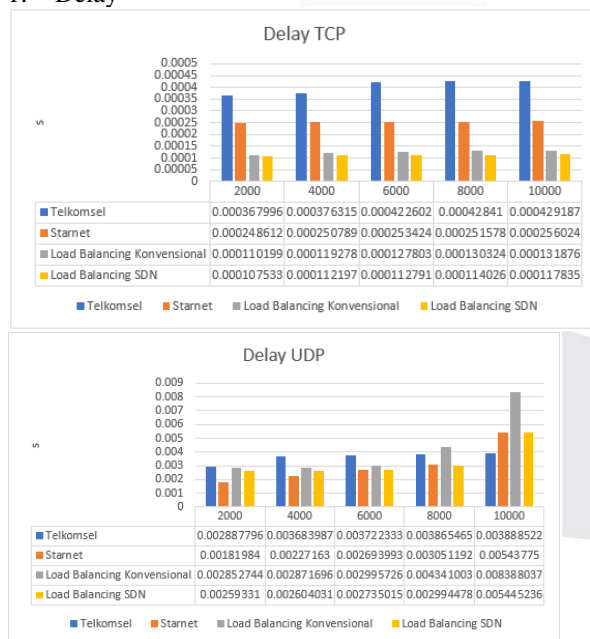


Gambar 4. 3 Topologi Logic SDN

E. Pengukuran Performansi QoS

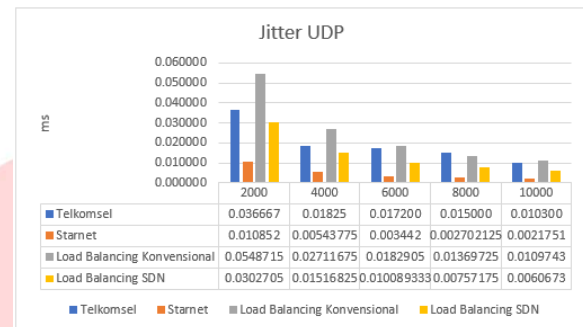
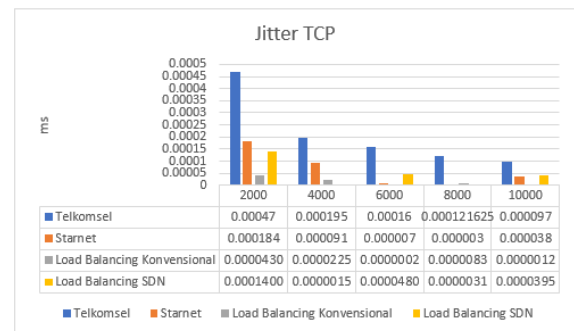
Pada pengukuran performansi jaringan Proyek Akhir ini menerapkan layanan *Load Balancing* dan *Radius Server* pada jaringan *konvensional* dan jaringan *SDN*. Pengukuran dilakukan dengan mengukur dua jenis protokol yaitu protokol *UDP* dan *TCP*. Pengukuran *UDP* dilakukan dengan melakukan *live streaming video* pada *youtube*. Pengukuran *TCP* dilakukan dengan melakukan proses *download* suatu file yang ada di internet. Pengambilan data dilakukan dengan cara mengambil jumlah *capture* paket dengan aplikasi *Wireshark*. Jumlah *capture* pada setiap pengukuran adalah 2000, 4000, 6000, 8000 dan 10000 *capture* serta diakhir akan diambil hasil rata-rata dari setiap pengukuran.

1. Delay



Gambar 4. 4 Grafik Delay

2. Jitter



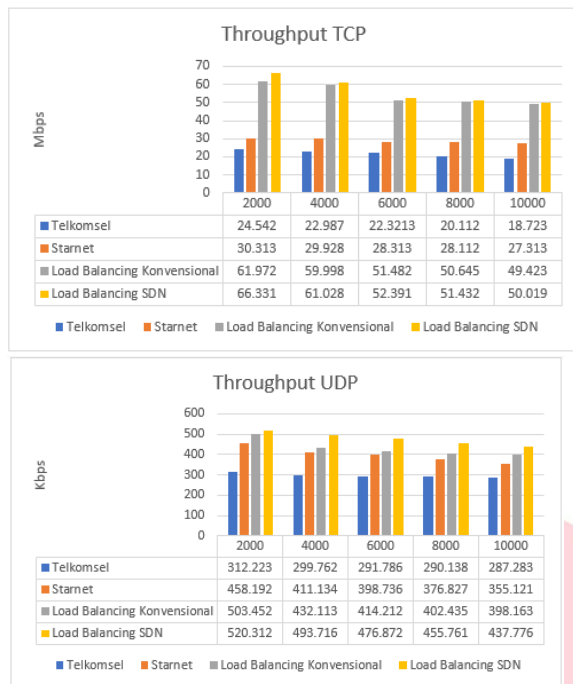
Gambar 4. 5 Grafik Jitter

3. Packet Loss



Gambar 4. 6 Grafik Packet Loss

4. Throughput



Gambar 4. 7 Grafik Throughput

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil Proyek Akhir yang telah dilakukan, maka kesimpulannya adalah sebagai berikut :

1. Implementasi *Load Balancing* dan *Radius Server* bisa dilakukan pada jenis jaringan konvensional dan jaringan SDN yang menggunakan *ONOS Controller* yang diinstal pada Linux Ubuntu 16.04 Server.
2. Implementasi *ONOS Controller* bisa dilakukan menggunakan mesin *virtual VMWare* yang sudah terinstall sistem operasi Linux Ubuntu 16.04 Server.
3. Hasil pengujian *throughput* dari kedua ISP menunjukkan hasil optimal. Dengan menggabungkan dua jenis koneksi yang berbeda serta manajemen penggunaan *traffic* memungkinkan bisa menambah *throughput* dan mengurangi *overload* pada jaringan.
4. Pada Proyek Akhir ini dilakukan perbandingan hasil QoS berdasarkan protokol UDP dan TCP dari setiap jenis jaringan. Pengambilan jumlah *capture* pada setiap pengukuran berjumlah 2000, 4000, 6000, 8000 dan 10000. Hasil dari setiap pengukuran masih menunjukkan kualitas QoS yang baik sesuai dengan standar TIPHONE.

B. Saran

Berdasarkan hasil pembangunan Proyek Akhir ini, dapat disampaikan beberapa saran untuk pengembangan selanjutnya yaitu:

1. Mengembangkan *Load Balancing* dan

Radius Server pada *controller* lain.

2. Mengimplementasikan *controller* pada *real devices*.
3. Mengembangkan jumlah *switch OpenFlow* yang digunakan pada jaringan SDN.
4. Mengimplementasikan *ONOS Controller* dengan versi terbaru.

DAFTAR PUSTAKA

- [1] M. N. Yaqin, R. Tulloh dan I. D. Irawati, "PERANCANGAN DAN IMPLEMENTASI PROTOKOL ROUTING EBGP PADA SOFTWARE DEFINED NETWORK MENGGUNAKAN ONOS CONTROLLER," *e-Proceeding of Applied Science*, vol. 6, no. 1, pp. 574 - 579, June 2020.
- [2] R. C. Kurniawan S, R. Tulloh dan I. D. Irawati, "IMPLEMENTASI VPLS PADA JARINGAN SOFTWARE DEFINED NETWORK (SDN) DENGAN MENGGUNAKAN ONOS CONTROLLER BERBASIS RASPBERRY-PI 3," *e-Proceeding of Applied Science*, vol. 6, no. 2, pp. 3796-3805, 2020.
- [3] A. M. Komaruddin, D. M. Sipitorini dan P. Rispian, "LOAD BALANCING DENGAN METODE ROUND ROBIN UNTUK PEMBAGIAN BEBAN KERJA WEB SERVER," *Jurnal Siliwangi*, vol. 5, no. 2, pp. 48 - 50, March 2019.
- [4] I. Santoso, . M. . H. H. Ichsan dan W. Yahya, "Implementasi Switch Openflow Berbasis Software Dengan Memanfaatkan Raspberry Pi Untuk Infrastruktur SDN," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 1, pp. 981-988, Desember 2019.
- [5] E. Hendarto, "SISTEM MANAJEMEN USER LOGIN HOTSPOT MIKROTIK DENGAN RADIUS SERVER MENGGUNAKAN RASPBERRY PI," *Jurnal Mahasiswa Teknik Informatika*, vol. 2, no. 2, pp. 72 - 81, 2018.
- [6] S. S. Hanadwiputra dan K. M. Prabowo, "Optimalisasi Radius Server Sebagai Sistem Otentikasi dan Otorisasi untuk Proses Login Multi Aplikasi Mikhmon Menggunakan User Manager di Mikrotik," *Jurnal PowerPlant*, vol. 6, no. 2, pp. 86 - 91, 2018.
- [7] F. Ardianto, B. Alfaresi dan A. Darmadi, "RANCANG BANGUN LOAD BALANCING DUA INTERNET SERVICE PROVIDER (ISP) BERBASIS MIKROTIK," *Jurnal Surya Energy*, vol. 3, no. 1, pp. 198 -202, 2018.

- [8] "Load Balance dengan Menggunakan Metode PCC (Simple)," CITRAWEB SOLUSI TEKNOLOGI, PT, 18 Desember 2020. [Online]. Available: https://citraweb.com/artikel_lihat.php?id=417. [Diakses 12 Januari 2022].
- [9] M. Riadi, "Pengertian, Layanan dan Parameter Quality of Service (QoS)," Kajian Pustaka, 26 Mei 2019. [Online]. Available: <https://www.kajianpustaka.com/2019/05/pengertian-layanan-dan-parameter-quality-of-service-qos.html>. [Diakses 13 Januari 2022].
- [10] I. "Dirjen PPI: Survei Penetrasi Pengguna Internet di Indonesia Bagian Penting dari Transformasi Digital," Kementerian Komunikasi dan Informatika Republik Indonesia, 09 November 2020. [Online]. Available: https://www.kominfo.go.id/content/detail/30653/dirjen-ppi-survei-penetrasi-pengguna-internet-di-indonesia-bagian-penting-dari-transformasi-digital/0/berita_satker. [Diakses 12 Januari 2022].
- [11] "Production Quality, Multilayer Open Virtual Switch," Linux Foundation, 2016. [Online]. Available: <https://www.openvswitch.org/>. [Diakses 13 Januari 2022].
- [12] P. R. Utami, "ANALISIS PERBANDINGAN QUALITY OF SERVICE JARINGAN INTERNET BERBASIS WIRELESS PADA LAYANAN INTERNET SERVICE PROVIDER (ISP) INDIHOME DAN FIRST MEDIA," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 20, no. 2, pp. 125 - 137, 2020.
- [13] S. R. Sari, R. Munadi dan D. D. Sanjoyo, "ANALISIS PERFORMANSI SEGMENT ROUTING PADA SOFTWARE DEFINED NETWORK," *e-Proceeding of Engineering*, vol. 6, no. 2, pp. 3469-3476, 2019.
- [14] Linux Foundation, "Production Quality, Multilayer Open Virtual Switch," 2016. [Online]. Available: <https://www.openvswitch.org/>. [Diakses 4 Juni 2022].
- [15] R. P. Sari dan F. S. Redha, "Sistem Pendukung Keputusan Pemilihan Distro Linux Menggunakan," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 2, no. 3, pp. 348-355, 2021.
- [16] V. Monita, I. D. Irawati dan R. Tulloh, "Comparison of routing protocol performance on multimedia services on software defined network," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 4, pp. 1612-1619, 2020.