

PERANCANGAN DAN IMPLEMENTASI PROTOKOL ROUTING EBGP PADA SOFTWARE DEFINED NETWORK MENGGUNAKAN ONOS CONTROLLER

Design and Implementation of EBGP Routing Protocol for Software Defined Network With Onos Controller

Muhammad Nurul Yaqin¹, Rohmat Tulloh, S.T., M.T.², Dr. Indarini Dyah Irawati, S.T., M.T.³

¹ Prodi D3 Teknologi Telekomunikasi, ²Fakultas Ilmu Terapan, ³Universitas Telkom

Jln. Telekomunikasi No.1 Terusan Buah Batu Bandung 40257 Indonesia

mny1702@gmail.com, rohmatth@telkomuniversity.ac.id indrarini@telkomuniversity.ac.id

ABSTRAK

Pada jaman sekarang kebutuhan atas layanan jaringan semakin meningkat, maka secara tidak langsung meningkat juga kebutuhan penyedia layanan mulai segi bandwidth / penambahan perangkat dan itu akan membuat segi kompleksitas jaringan meningkat dan segi konfigurasi semakin memakan waktu yang lama, tidak hanya itu dengan adanya teknologi Software Defined Network (SDN) yang bergantung pada controller terpusat dan apabila controller tersebut traffic jaringannya sudah penuh maka akan melumpuhkan semua jaringan yang terkoneksi ke jaringan SDN tersebut.

Dalam Proyek Akhir kali ini akan coba membahas tentang teknologi jaringan SDN menggunakan Open Network Operating System (ONOS) yang mampu memisahkan antara control plane dan data plane yang biasanya terdapat dalam satu perangkat jaringan, kini control plane bisa dipindah ke bagian core network. ONOS juga support untuk virtualisasi perangkat seperti router dan switch menggunakan quagga, mininet, docker atau openstack sehingga mampu mengurangi penggunaan perangkat. Untuk merealisasikan jaringan SDN pada jaringan WAN, ONOS mempunyai fitur ONOS Cluster yang memungkinkan untuk mendistribusikan beban traffic ke controller lain dengan mudah. Untuk memastikan semua perangkat pada jaringan SDN sudah terhubung dengan baik, ONOS mempunyai fitur ONOS Intents yang memudahkan teknisi jaringan untuk mengatur jaringan dari high level.

Pada proyek akhir ini jaringan SDN-IP terdiri dari 4 switch OpenFlow dan 4 router yang semuanya terhubung ke ONOS controller, ada 2 cara pengujian pada Proyek Akhir ini, yaitu dengan menggunakan layanan FTP dan VoIP. Untuk FTP QoS yang didapatkan adalah 6,367 MB/s (throughput) 0,152 ms (Delay) dan 0,01 ms (Jitter). Untuk VoIP yaitu 32 KB/s (throughput) 6,66 ms (Delay) dan 1,14 ms (Jitter)

Kata Kunci : Software Defined Network, WAN, Onos Controller, ONOS Cluster, ONOS intent, OpenFlow

ABSTRACT

In this day and age the need for network services is increasing, then indirectly increasing also the need for service providers starting in terms of bandwidth / additional devices and that will make the network complexity increase and the configuration aspect increasingly takes a long time, not only that with the existence of Software Defined Network (SDN) that relies on a centralized controller and if the controller is full network traffic it will cripple all networks connected to the SDN network.

In this Final Project, we will try to discuss SDN network technology using the Open Network Operating System (ONOS) which is able to separate the control plane from the data plane that is usually contained in a network device, now the control plane can be moved to the core network. ONOS also supports virtualization of devices such as routers and switches using Quagga, Mininet, Docker or OpenStack so as to reduce device usage. To realize the SDN network on a WAN network, ONOS has an ONOS Cluster feature that makes it possible to easily distribute traffic loads to other controllers. To ensure all devices on the SDN network are well connected, ONOS has an ONOS Intents feature that makes it easy for network technicians to manage networks from high level.

In this final project the SDN-IP network consists of 4 OpenFlow switches and 4 routers which are all connected to the ONOS controller, there are 2 ways of testing in this Final Project, namely by using FTP and

VoIP services. For FTP QoS obtained is 6.367 MB / s (throughput) 0.152 ms (Delay) and 0.01 ms (Jitter). For VoIP, which is 32 KB / s (throughput) 6.66 ms (Delay) and 1.14 ms (Jitter)

Keyword : Software Defined Network, WAN, Onos Controler, ONOS Cluster, ONOS intent, OpenFlow

1. Pendahuluan

Beberapa penyedia layanan internet berencana untuk berinovasi infrastruktur mereka melalui proses pemerangkat lunakan jaringan yang bisa di program. Paradigma SDN bertujuan untuk meningkatkan kelincahan desain, konfigurasi, pemeliharaan, dan penyediaan layanan jaringan melalui control plane yang terpusat yang bertugas mengelola seluruh sistem. Ini mudah dicapai untuk jaringan lokal, dimana manfaat memiliki akses yang dapat diprogram ke seluruh jairngan tidak dibatasi oleh latency. Namun, di Wide Area Networks, control plane terpusat membatasi kecepatan respons terhadap peristiwa jaringan yang dibatasi waktu karena latency yang tidak dapat dihindari yang disebabkan oleh jarak fisik [1] .

Untuk mendukung jaringan produksi skala besar, Network Operating System harus memenuhi persyaratan yang menuntut Scalability, Performance, dan Availability. Untuk mengatasi tantangan tersebut, ONF memperkenalkan ONOS [2]. ONOS mengadopsi aritektur terdistribusi untuk Availability dan skala tinggi. Dalam arsitektur SDN terdistribusi, sebuah jaringan di kontrol melalui beberapa controller [3]. Pada controller ONOS terdapat fitur ONOS cluster yang memungkinkan controller bisa membagikan beban traffic. Selain ONOS cluster terdapat juga fitur ONOS intents yang memudahkan engineer jaringan untuk mengatur jaringan dari high level tanpa takut detail perangkat yang berada dibawahnya [4]. ONOS adalah controller berbasis Java dan memanfaatkan Open Service Gateway initiatives (OSGi) untuk mempertahankan aplikasi dengan menginstal, memulai, menghentikan, memperbarui, dan mencopot pemasangan dari jarak jauh tanpa perlu reboot. ONOS memberikan keamanan yang baik, dengan melacak dan memblokir akses tidak sah ke sumber daya pada saat run-time. Dengan mengaktifkan Transport Layer Security (TLS) dan Hypertext Transfer Protocol Secure (HTTPS) di Southbound Interface (SBI) dan Northbound Interface (NBI) masing-masing, pengungkapan informasi dan ancaman gangguan ditangani. Antarmuka Pengguna Grafis (GUI) ONOS juga memberikan kinerja terbaik dibandingkan dengan Odl, Ryu, dan POX untuk bandwidth yang melebihi 9 Gbps [5].

Sebelumnya telah dilakukan penelitian tentang perbandingan topologi serta jumlah perangkat jaringan SDN menggunakan ONOS yang menggunakan switch berjumlah 16 dan 32 switch serta menggunakan topology linear, custom, dan torus tanpa adanya protocol routing. RTT minimum untuk topologi torus yang diperoleh untuk konfigurasi 16 dan 32 switch adalah 0,054 ms hingga 0,063 ms. Sebaliknya, topologi linear menunjukkan RTT 0,076 ms dalam 16 kasus switch dan 0,163 ms dalam kasus 32 switch. Dalam topologi kustom yang dibuat, RTT minimum antara switch 16 dan 32 kasus switch masing-masing adalah 0,069 ms dan 0,094 ms [5] . Berbeda dengan Proyek Akhir kali ini, menggunakan protocol routing BGP serta dilakukan implementasi langsung pada perangkat. Pada proyek akhir ini jaringan SDN-IP terdiri dari 4 switch OpenFlow dan 4 router yang semuanya terhubung ke ONOS controller, ada 2 cara pengujian pada Proyek Akhir ini, yaitu dengan menggunakan layanan FTP dan VoIP. Untuk FTP QoS yang didapatkan adalah 6,367 MB/s (throughput) 0,152 ms (Delay) dan 0,01 ms (Jitter). Untuk VoIP yaitu 32 KB/s (throughput) 6,66 ms (Delay) dan 1,14 ms (Jitter)

2. Dasar Teori

2.1. Software Defined Network

SDN adalah sebuah konsep pendekatan jaringan komputer dimana sistem pengontrol (*control plane*) dari arus data dipisahkan dari perangkat kerasnya (*data plane*)[6]. Pada teknologi jaringan konvensional, sistem pembuat keputusan kemana arus data dikirimkan dibuat menyatu dengan perangkat kerasnya. Namun di dalam teknologi SDN memiliki dua karakteristik. Pertama, SDN memisah antara *control plane* dari *data plane*. Kedua, SDN menggabungkan *control plane* setiap perangkat menjadi sebuah kontroler yang berbasis *programmable software*. Sehingga sebuah kontroler tersebut dapat mengontrol banyak perangkat dalam sebuah *data plane*. Di dalam SDN sebuah jaringan tersentralisasi dalam sebuah kontroler yang berbasis *software* yang dapat memelihara jaringan secara keseluruhan, Sehingga dapat mempermudah dalam mendesain dan mengoperasikan jaringan karena hanya melalui sebuah *logical point*.

2.2. Open Network Operating System (ONOS)

Open Network Operating System atau yang sering disingkat ONOS merupakan Operasi Sistem *Open Source* SDN pertama yang yang ditargetkan untuk penyedia layanan jaringan (*service provider*) [2]. Sebelumnya tim gabungan dari universitas Stanford, Berkeley, dan Nicira Networks membuat beberapa *controller open source* seperti NOX, POX dan Beacon. Namun, perlu diketahui bahwa *controller* tersebut dirancang hanya untuk mengeksplorasi dan demonstrasi potensi SDN. Perlu diketahui juga *controller* NOX, POX dan Beacon merupakan *controller* SDN yang tidak dirancang untuk kebutuhan komersial yang membutuhkan aspek *Scalability, High Availability, dan Performance* [2].

Kernel ONOS dan *core services*, ditulis menggunakan Bahasa Java dan dimuat ke karaf *Open Service Gateway initiatives (OSGi) container* yang memungkinkan untuk memasang (*install*), mencopot (*remove*), memperbarui (*update*) tanpa harus melakukan reboot controller / operasi system.

2.3. Border Gateway Protocol

Border Gateway Protocol adalah protokol routing standar yang bertujuan untuk memilih jalur – jalur interdomain. Fungsi utama dari BGP untuk mempertukarkan *network reachability information* antar BGP router [12]. BGP dirancang untuk menyediakan link bebas loop antar penyedia layanan. BGP menggunakan protokol TCP (*port* 179). Dalam protokol BGP terdapat sebuah *Autonomous Systems (AS)*, AS merupakan sebuah group router yang saling berbagi aturan kebijakan routing yang sama dan beroperasi dalam satu *domain* administrative, satu penyedia layanan biasanya hanya mempunyai satu AS . Nomor AS yang tersedia antara 1 sampai 65.535.

2.4. SDN-IP

SDN-IP adalah aplikasi ONOS yang memungkinkan sebuah jaringan SDN untuk terhubung ke jaringan lain di internet menggunakan Border Gateway Protocol (BGP). Dari sudut pandang BGP, sebuah jaringan SDN dikenali sebagai single Autonomous System (AS) .[8] Cara kerja dari SDN-IP adalah jalur routing yang dikenalkan oleh *border router* ke jaringan luar diterima oleh BGP *speaker* kemudian diproses oleh SDN-IP dan akhirnya dikenalkan kembali ke jaringan luar. Routing diproses dengan kebijakan BGP normal. Demikian pula, routing tersebut dikenalkan ke *controller* sebagai iBGP normal. Jalur terbaik untuk setiap tujuan diproses oleh SDN-IP sesuai aturan iBGP dan terjemahkan kedalam sebuah aplikasi ONOS yaitu *intents request*. ONOS menerjemahkan *intents request* kedalam *forwarding rules* di *data plane*. Aturan tersebut digunakan untuk meneruskan traffic antara jaringan IP yang terhubung. SDN-IP membuat ONOS *intents* yang bertujuan untuk memungkinkan router BGP eksternal untuk melihat ke BGP *speaker* menggunakan perangkat *data plane*. Dengan demikian, BGP dapat dengan mudah mencari kesalahan pada perangkat *data plane* yang mempengaruhi lalu lintas data yang tidak diharapkan.

2.5. Quagga

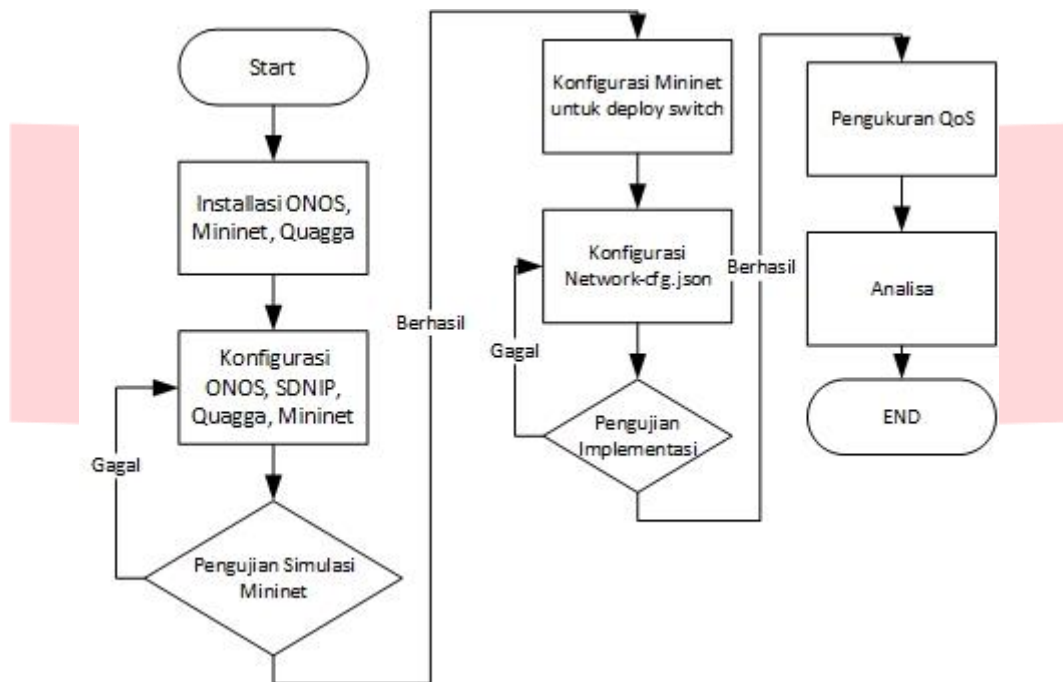
Quagga adalah suatu *routing engine* yang dapat menjalankan protokol *routing* konvensional seperti RIP, OSPF, BGP, dan lain-lain. Quagga yang tujuannya secara umum implementasi *routing* bersifat *open source* yang cocok untuk digunakan di SDN *environment*. Penerapan quagga dapat digunakan untuk Unix *platform* khususnya FreeBSD, Linux, Solaris, dan NetBSD. Arsitektur Quagga terdiri dari *core daemon* , zebra, yang bertindak sebagai layer abstraksi untuk Unix kernel yang mendasari dan menyajikan Zserv API diatas unix atau TCP *stream* untuk Quagga *client*. Pada Zserv *client* ini biasanya menerapkan protokol *routing* dan menyampaikan *update routing* ke zebra daemon[15].

2.6. OpenFlow

OpenFlow adalah *protocol* komunikasi layer 2 yang memberikan akses ke perangkat *data plane* seperti *switch* atau *router* melalui sebuah jaringan, *OpenFlow* merupakan sub-bagian dari sebuah jaringan *Software Defined Network* [9]. *OpenFlow* biasanya diimplementasikan untuk menghubungkan / komunikasi antara *controller* dan perangkat *data plane*. *OpenFlow* mendefinisikan infrastruktur *flow-based forwarding* dan *Application Programmatic interface (API)* standar yang memungkinkan *controller* untuk mengarahkan fungsi dari *switch* melalui saluran yang aman (*secure chanel*)[9]. *Controller* yang telah diprogram sesuai dengan konfigurasi jaringan yang diinginkan dapat terhubung dengan *forward plane*. Dimana *secure channel* ini merupakan perantara bagi *controller* untuk mengontrol jaringan pada *forward plane*.

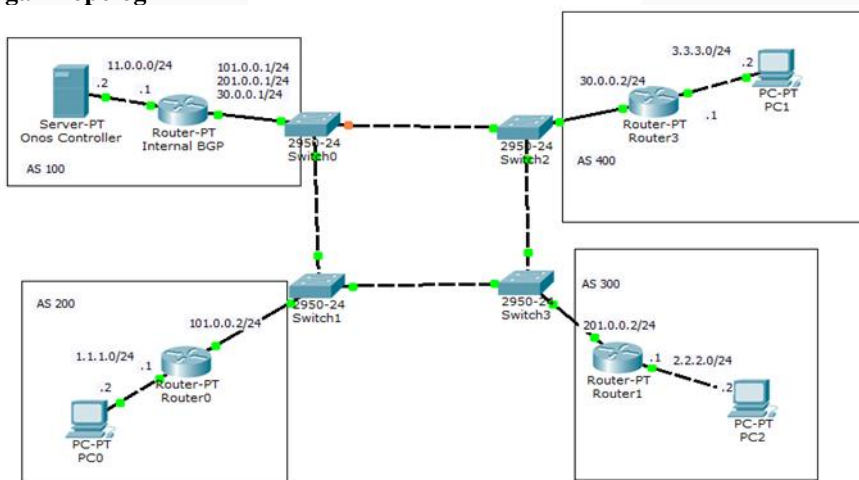
3. Perancangan dan Implementasi Sistem

3.1. Perancangan Sistem



Gambar 3.1 Diagram Alir Pengerjaan

3.2. Perancangan Topologi



Gambar 3.2 Topologi Jaringan

Pada tahap ini penulis melakukan perancangan jaringan yaitu berupa membuat topologi yang akan diimplementasikan. Komponen yang dibutuhkan yaitu 4 buah *switch openflow* sebagai *data plane*, 4 buah router, dan 4 laptop (1 sebagai *controller* dan 3 buah sebagai *client*)

3.3. Perancangan Operasi System

Pada proses implementasi. Desain software yang digunakan adalah dengan menggunakan Ubuntu 16.04 LTS yang telah terinstall ONOS yang digunakan sebagai Controller Jaringan SDN. 5 virtual server menggunakan operasi system Ubuntu server 16.04 yang masing – masing digunakan untuk :

1. Internal BGP = terinstall quagga
2. Virtual router 1 = terinstall quagga
3. Virtual router 2 = terinstall quagga

4. Virtual router 3 = terinstall quagga
5. Server mininet = terinstall mininet

4. Hasil Pengujian dan Analisis

Berikut adalah hasil pengujian dan analisis setiap parameter yang diuji.

4.1 Analisa Pengujian FTP

Berdasarkan hasil pengujian yang telah dilakukan pada Proyek Akhir ini, langkah terakhir dari bab IV adalah membandingkan nilai hasil pengujian dengan standarisasi QoS ITU-T G 1010 End-user multimedia QoS Categories. Berikut standar yang digunakan:

Tabel 4.1 Hasil Pengukuran FTP

NO	Parameter	Hasil Pengukuran Implementasi	ITU-T G. 1010
1	<i>Throughput</i>	6,367 MB/sec	NA
2	<i>Delay</i>	0,15 ms	<i>Preffered < 15 s acceptable < 60 s</i>
3	<i>Jitter</i>	0,01 ms	NA
4	<i>Packet Loss</i>	0 %	0 %

Tabel 4.2 Hasil Pengukuran VoIP

NO	Parameter	Hasil Pengukuran Implementasi	ITU-T G. 1010
1	<i>Throughput</i>	32 KB/sec	NA
2	<i>Delay</i>	6,66 ms	<i>preferred <150 ms limit <400 ms</i>
3	<i>Jitter</i>	1,14 ms	1 ms
4	<i>Packet Loss</i>	0 %	3 %

5. Kesimpulan

Berdasarkan hasil penelitian selama pengerjaan proyek akhir ini dapat ditarik kesimpulan bahwa:

1. Implementasi jaringan SDN yang memisahkan fungsi kontrol dengan fungsi forwarding dapat dilakukan dengan menggunakan ONOS
2. Hasil pengujian performansi penerapan routing eBGP pada Software Defined Network menunjukkan bahwa nilai ketiga dari empat parameter QoS (throughput, jitter, delay dan packet loss) masih berada pada nilai yang menjadi standar ITU-T G.1010.
3. Implementasi jaringan software defined network dengan menggunakan Controller ONOS telah selesai dibuat dan di uji performansinya menggunakan parameter convergence time, throughput, delay, jitter dan packet loss dengan hasil sebagai berikut:
 - a. Rata-rata throughput implementasi dari hasil Proyek Akhir ini adalah 6,367 MB/s untuk FTP dan 32 KB/s untuk VoIP
 - b. Rata-rata delay implementasi dari hasil Proyek Akhir ini adalah 0,15 ms untuk FTP dan 6,66 ms untuk VoIP
 - c. Rata-rata jitter implementasi dari hasil Proyek Akhir ini adalah 0,01 ms untuk FTP dan 1,14 untuk VoIP
 - d. Rata-rata packet loss implementasi dari hasil Proyek Akhir ini adalah 0 %

DAFTAR PUSTAKA

- [1] M. Gerola *et al.*, "ICONA: A peer-to-peer approach for software defined wide area networks using ONOS," *Proc. - Eur. Work. Software-Defined Networks, EWSDN*, vol. 16, pp. 37–42, 2017.
- [2] ON.LAB, "Introducing ONOS - a SDN network operating system for Service Providers," *White Pap.*, vol. 1, p. 14, 2014.
- [3] P. Berde *et al.*, "ONOS: towards an open, distributed SDN OS," *Proc. third Work. Hot Top. Softw. Defin. Netw.*, vol. 1, pp. 1–6, 2014.
- [4] R. A. Addad, D. L. C. Dutra, M. Bagaa, T. Taleb, H. Flinck, and M. Namane, "Benchmarking the ONOS Intent Interfaces to Ease 5G Service Management," *2018 IEEE Glob. Commun. Conf. GLOBECOM 2018 - Proc.*, pp. 1–6, 2018.
- [5] A. Rajaratnam, R. Kadikar, S. Prince, and M. Valarmathi, "Software defined networks: Comparative analysis of topologies with ONOS," *Proc. 2017 Int. Conf. Wirel. Commun. Signal Process. Networking, WiSPNET 2017*, vol. 2018-Janua, pp. 1377–1381, 2018.
- [6] R. M. Negara and R. Tulloh, "Analisis Simulasi Penerapan Algoritma OSPF Menggunakan RouteFlow pada Jaringan Software Defined Network (SDN)," *J. Infotel*, vol. 9, no. 1, p. 75, 2017.
- [7] Violet R. Syrotiuk, "An Introduction to Software Defined Networking and OpenFlow," *Glob. Environ. Netw. Innov.*, vol. 1, 2017.
- [8] H. A. Friwansya, I. D. Irawati, Y. S. Hariyani, F. I. Terapan, and U. Telkom, "IMPLEMENTASI PROTOKOL ROUTING EBGp PADA SOFTWARE DEFINED," *E-Proceeding Applied Sci.*, vol. 4, no. 3, pp. 2453–2462, 2018.
- [9] M. Bjorn, "Software Defined Networks (SDN) and its relevance in the DC," *E-Proceeding Cisco Connect*, vol. 1, pp. 1–43, 2013.
- [10] C. Hao, Chang, and Y.-D. Lin, "OpenFlow Version Roadmap," *E-Proceeding Natl. Chiao Tung Univ.*, vol. 1, pp. 1–15, 2015.
- [11] Dimitra Sakellaropoulou, "A Qualitative Study of SDN Controllers," *E-Proceeding MSc. Thesis Athens Univ. Econ. Bus.*, vol. 1, 2017.
- [12] T. Ernawati and J. Endrawan, "Peningkatan Kinerja Jaringan Komputer dengan Border Gateway Protocol (BGP) dan Dynamic Routing," *J. Ilmu Komput. dan Inform.*, vol. 4, no. 1, p. 35, 2018.
- [13] A. Cvjetic and A. Smiljanic, "Improving BGP protocol to advertise multiple routes for the same destination prefix," *IEEE Commun. Lett.*, vol. 18, no. 1, pp. 106–109, 2014.
- [14] L. Todd, *CCNA Routing and Switching Study Guide: Exams 100-101, 200-101, and 200-120*. 2013.
- [15] A. Kodar, "Analisa Dan Uji Kinerja PC Router Yang Menjalankan Protokol Routing Border Gateway Protocol (BGP) Menggunakan Zebra / Quagga," *Semin. Nas. Pengaplikasian Telemat. SINAPTIKA 2010*, vol. 1, pp. 162–167, 2010.
- [16] F. Adnanya, S. N. Hertiana, L. Vidya, and Y. St, "Simulasi Dan Analisis Performansi Protokol Ruting Ebgp Pada Sdn (Software Defined Network) Simulation and Perfomance Analysis of Ebgp Routing Protocol on Sdn (Software Defined Network)," *e-Proceeding Eng.*, vol. 2, no. 2, pp. 2346–2353, 2015.
- [17] Yanto, "Analisis Qos (Quality of Service) Pada Jaringan Internet (Studi Kasus: Fakultas Teknik Universitas Tanjungpura)," *J. Mhs. Fak. Sains dan Teknol.*, vol. 0, no. 0, 2013.
- [18] ITU-T Rec-G.1010, "End-user multimedia QoS Categories," *Telecommun. Stand. Sect. Int. Telecommun. Union Recomm. G. 1000.*, vol. 1010, 2001.
- [19] A. Irmawati, I. D. Irawati, and Y. S. Hariyani, "IMPLEMENTASI PROTOKOL ROUTING OSPF PADA SOFTWARE DEFINED NETWORK BERBASIS ROUTEFLOW," *E-Proceeding Applied Sci.*, vol. 2, no. 3, pp. 1053–1061, 2016.