Analisis Performansi *High Availability* Jaringan pada *Virtual Private LAN Service* Legasi dan Berbasis *Software Defined Network*

Alaurin Attari [1)], Ridha Muldina Negara [2)], Danu Dwi Sanjoyo [3)]

[1),2),3)]S1 Teknik Telekomunikasi, Universitas Telkom
Jl. Telekomunikasi No. 1, Bandung
Email : alaurinattari@gmail.com

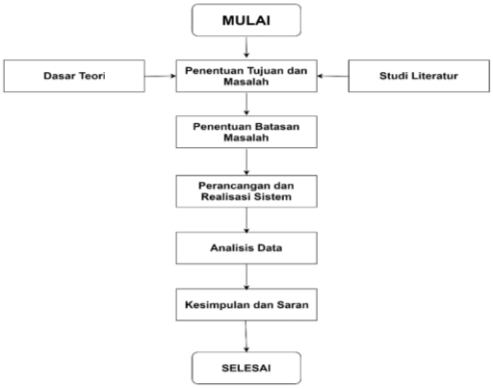
Abstrak. *Virtual Private LAN Service (VPLS) adalah sebuah layanan yang mampu menghubungkan beberapa site dengan letak geografis yang berjauhan namun tetap terasa berada dalam satu jaringan lokal yang sama. VPLS sudah diterapkan pada jaringan legasi dan jaringan berbasis Software Defined Network (SDN). Pada jaringan SDN kontrol VPLS dilakukan secara terpusat oleh controller sedangkan pada jaringan legasi kontrol dilakukan secara tersebar pada masing – masing router. Hal ini menyebabkan failover delay yang tinggi saat terjadi kegagalan pada sistem. Availabilitas jaringan yang baik berhubungan dengan waktu failover delay ketika terjadi kegagalan sistem. Semakin besar waktu failover delay maka high availability (HA) jaringan akan semakin menurun. Pada tugas akhir ini dilakukan pengujian high availability terhadap jaringan VPLS legasi dan VPLS SDN menggunakan ONOS controller untuk melihat pengaruh kontrol terpusat dan kontrol menyebar terhadap high availability jaringan VPLS. Jenis kegagalan sistem yang diujikan adalah dengan melakukan pemutusan link dengan parameter pengujian failover delay. Berdasarkan hasil pengujian didapatkan kesimpulan bahwa failover delay dari penerapan SDN dengan kontrol terpusat 44.7% lebih rendah dibandingkan dengan VPLS legasi tanpa SDN dengan kontrol menyebar. Berdasarkan parameter uji ini, menjadikan VPLS SDN lebih unggul dalam high availability (HA).*

***Kata kunci**: VPLS, SDN, ONOS Controller, High Availability.*

1. Pendahuluan[1],[2],[3],[4],[5],[6],[7]

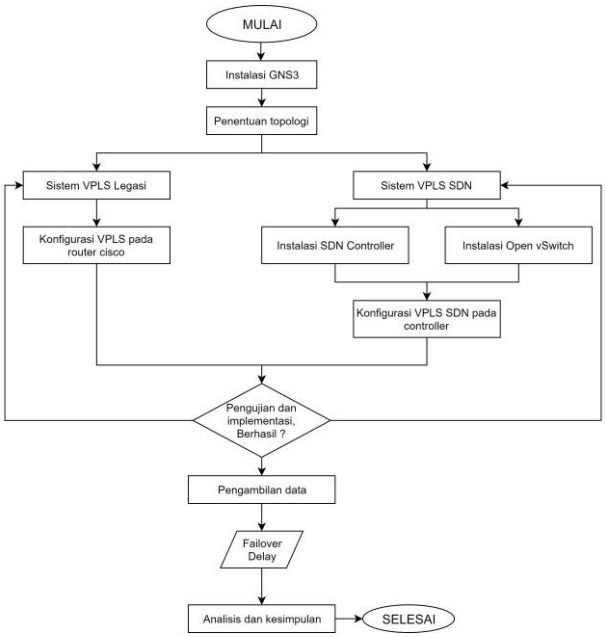
Pada era globalisasi ini pertumbuhan dalam dunia bisnis berkembang dengan sangat pesat. Banyak perusahaan berskala internasional yang memiliki banyak kantor cabang di beberapa negara merasa perlu untuk mengelola akses sumber dayanya, khususnya dalam pengawasan pengendalian dan pertukaran data perusahaan [4]. Biasanya suatu perusahaan besar akan menggunakan *Virtual Private Network* (VPN) untuk mengakses data rahasia, sehingga data dapat diakses dan dikontrol dengan aman melalui jaringan publik seperti internet. Untuk menerapkan teknologi VPN pada perusahaan berskala internasional dibutuhkan sebuah teknologi yang dinamakan *Virtual Private LAN Service* (VPLS). VPLS kini menjadi salah satu teknologi yang memiliki popularitas tinggi diantara perusahaan – perusahaan *enterprise* sebagai solusi jaringan ideal untuk mengelola data perusahaan dan melakukan kontrol perangkat pada sebuah jaringan bersama. IETF menstandarisasi dua jenis VPLS dengan menggunakan BGP dan LDP [5] [6]. VPLS ini adalah sebuah teknologi pada layer dua yang dapat menghubungkan beberapa *sites* menggunakan sebuah jembatan *(bridge)* dengan menggunakan jaringan *Internet Protokol* (IP) yang diatur oleh penyedia layanan dengan *Multi Protokol Label Switching* (MPLS) sebagai pendukungnya [3]. VPLS memperluas domain *broadcast* Ethernet menjadi beberapa *sites* yang terpisah secara geografis baik dalam satu wilayah negara maupun diseluruh dunia [1]. Namun seperti yang diketahui jaringan VPLS yang sudah ada sekarang ini atau bisa disebut dengan VPLS legasi memiliki arsitektur yang kompleks dan tidak fleksibel. Karena jaringannya yang kompleks dan penerapannya yang menyebar secara geografis sehingga menimbulkan jarak yang besar antar pengguna menyebabkan adanya *delay* yang sangat tinggi [1]. Oleh karena itu diperlukan suatu konsep teknologi baru untuk meningkatkan efisiensi jaringan VPLS legasi khususnya dalam *high availability* (HA) jaringannya yang memiliki kaitan erat terhadap *delay*. Maka dari itu akan dilakukan penerapan konsep *Software Defined Network* (SDN) pada jaringan VPLS dengan melakukan kontrol terpusat. Salah satu masalah yang tidak dapat dihindari dari penerapan VPLS ini sendiri adalah *delay* yang sangat tinggi dikarenakan jarak geografis antar site [2]. Untuk pembangunan jalur komunikasi sendiri VPLS legasi bisa memakan waktu paling sedikit 2000 ms antar *site* VPLS dengan transmisi delay sebesar 500 ms [7]. Hal ini tentu sangat mempengaruhi HA VPLS legasi. Maka pada tugas akhir ini akan melakukan pengujian *high availability* pada VPLS legasi dan VPLS yang menerapkan konsep *Software Defined Networking* dan dibandingkan untuk kemudian menghasilkan kesimpulan apakah

dengan adanya penerapan SDN dengan kontrol terpusat pada VPLS memiliki pengaruh yang signifikan terhadap avaibilitas jaringan VPLS. Adapun rumusan masalah pada penelitian ini yaitu mengetahui perbandingan *failover delay* antara jaringan VPLS SDN dan VPLS legasi serta menganalisa pengaruh penerapan konsep SDN pada VPLS dalam kasus *failover*. Metode penelitian yang digunakan adalah melakukan penentuan tujuan dan masalah dengan mencari dasar teori dan studi literatur, penentuan batasan masalah, perancangan dan realisasi sistem, analisis data, dan membuat kesimpulan. Berikut adalah diagram alir metode penelitian yang dilakukan :



Gambar 1. Diagram Alir Metode Penelitian

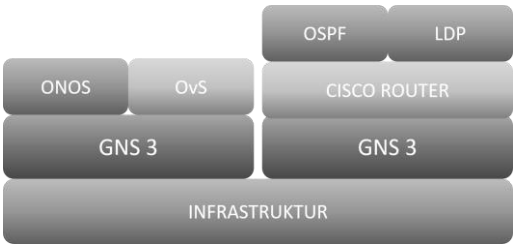
Berdasarkan diagram alir pada Gambar 1 di atas, secara garis besar proses perancangan dan realisasi sistem dilakukan dengan cara sebagai berikut :



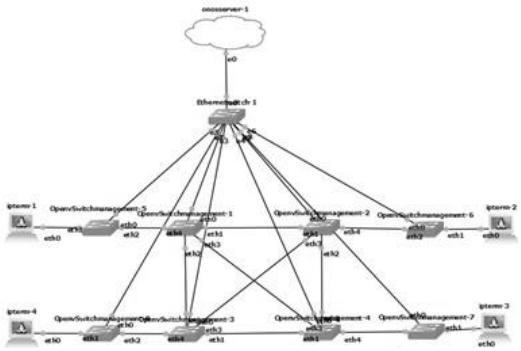
Gambar 2. Diagram Alir Perancangan dan Realisasi Sistem

Pada Gambar 2 di atas menunjukan alur proses perbandingan analisis performansi *high availability* antara VPLS legasi dan VPLS berbasis SDN untuk membantu dalam memahami proses perancangan yang dibuat. Pada VPLS legasi menggunakan cisco *router* yang akan dikonfigurasi dengan protokol LDP dan protokol OSPF untuk membangun jaringan VPLS sedangkan pada VPLS berbasis SDN, SDN *controller* yang digunakan adalah ONOS *peacock* versi 1.15.0, ONOS menyediakan *control plane* untuk SDN dan manajemen komponen – komponen jaringan seperti *switch* dan *link.* Perangkat jaringan yang digunakan adalah Open v*Switch* yaitu sebuah *programmable switch* untuk memungkinkan automasi jaringan masif dengan protokol Openflow di dalamnya. Seluruh *switch* yang ada dalam sistem ini terhubung melalui jaringan Ethernet. Setiap *switch* yang digunakan memiliki

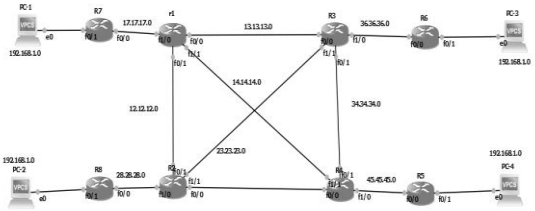
alamat IP yang telah dikonfigurasi dan dihubungkan dengan *controller.* Alamat IP yang digunakan oleh *switch* dan *controller* berada dalam satu *network* dan subnet yang sama dalam jaringan layer 2 *broadcast multi-access* Ethernet. Pada masing – masing Open v*Switch* VPLS SDN dibuat *virtual interface bridge* br0 dan br1. Hal ini dilakukan untuk melakukan pemisahan antara manajemen *switch* openflow dengan port akses antar host, *bridge* br0 akan terhubung langsung ke SDN *controller* sedangkan *bridge* br1 untuk komunikasi antar open v*switch* dan *client.* Arsitektur sistem yang dibangun ditunjukan pada Gambar 3 dan topologi sistem yang digunakan pada masing – masing VPLS ditunjukan pada Gambar 4 dan Gambar 5.



Gambar 3. Arsitektur Sistem



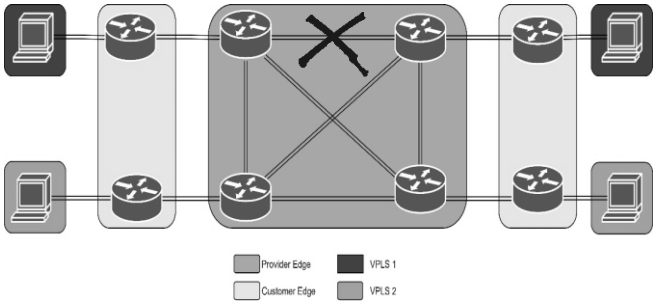
Gambar 4. Topologi VPLS SDN



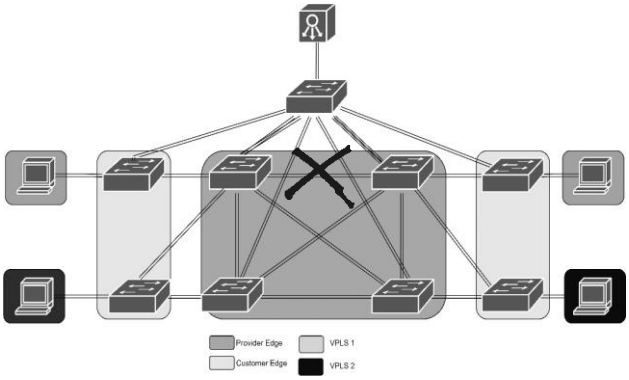
Gambar 5. Topologi VPLS Legasi

2. Pembahasan

Pengujian untuk mengukur *failover delay* akan dilakukan dengan metodologi kegagalan *link.* Pengujian *failover delay* ini bertujuan untuk mengetahui waktu yang dibutuhkan *controller* pada VPLS SDN dan *router* pada VPLS legasi untuk melakukan pengalihan ke rute alternatif. Dalam metodologi pengujian ini salah satu *host* dipilih secara acak untuk kemudian mengirimkan paket ICMP ke sisi *client* dalam satu jaringan VPLS yang sama. Paket ICMP akan dikirimkan secara kontinu dengan interval waktu 1 detik. Metodologi kegagalan *link* yang dilakukan dapat dilihat pada Gambar 6 dan Gambar 7. *Link* yang diputus berada pada sisi *Provider Edge* (PE).

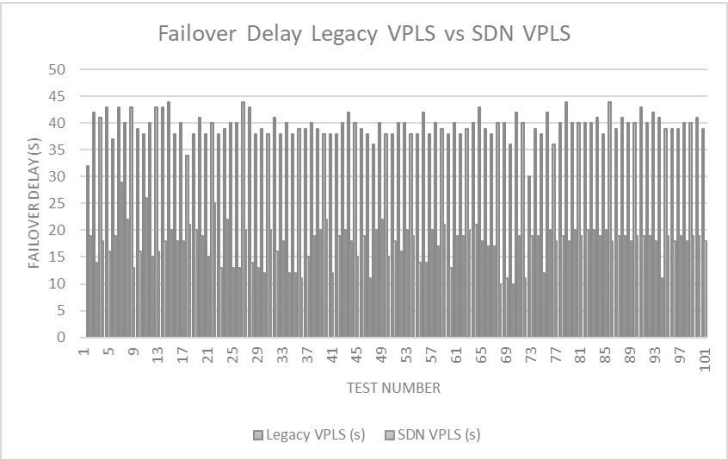


Gambar 6. Skenario Kegagalan *Link* VPLS Legasi



Gambar 7. Skenario Kegagalan *Link* VPLS SDN

Sementara komputer *host* terus melakukan pengiriman paket ICMP, pada detik ke sepuluh *link* pengiriman utama akan diputus. Hal ini menyebabkan pengiriman paket ICMP terganggu, selama kegagalan *link* ini terjadi paket ICMP akan mencari jalur alternatif yang dapat dilewati. Sementara paket ICMP sampai ditujuan dan mencari jalur alternatif, status paket akan terus tercatat pada jendela *host* pengirim. Ketika sedang terjadi kegagalan link maka akan muncul *Request Time Out* (RTO). Berdasarkan standarisasi IETF RTO yang direkomendasikan adalah minimal 80 ms atau 0.08 detik dan RTO maksimum sebesar 60000 ms atau 60 detik. Untuk menghitung *failover delay* dilakukan dengan menghitung jumlah RTO (*Request Time Out*) yang muncul dikalikan dengan interval waktu pengiriman paket ICMP. Setelah melakukan pengujian sebanyak 100 kali pada masing – masing VPLS didapatkan hasil seperti yang terlihat pada grafik di bawah ini.

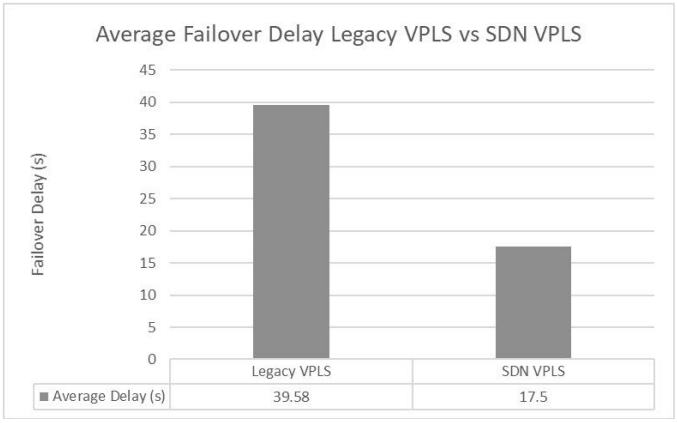


Gambar 8. Grafik Pengukuran *Failover Delay*

Pada grafik Gambar 8 didapatkan waktu *failover delay* tertinggi pada VPLS legasi adalah 44 detik dan waktu tersingkat adalah 30 detik, sedangkan pada VPLS SDN waktu *failover delay* tertinggi adalah 30 detik dengan waktu tersingkat adalah 8 detik. Berdasarkan rata – rata *failover delay* dari kedua VPLS terjadi penurunan *failover delay* hingga 44.7% dengan adanya penerapan SDN pada VPLS.

Hal ini terjadi dikarenakan pada VPLS SDN *control plane* dipisahkan dengan *data plane* dan fungsi *control plane* terletak pada *controller* yang tersentralisasi secara logika sehingga dapat melakukan kontrol *data plane* dengan menggunakan protokol OpenFlow. Dengan adanya kontrol terpusat ini memungkinkan adanya prosedur pengambilan keputusan tersentralisasi yang lebih efisien jika dibandingkan dengan prosedur yang terdistribusi pada *control plane* dan *data plane* yang digabung pada VPLS legasi. Dengan adanya protokol Openflow yang dapat mengatur perilaku jaringan pada *switch* memungkinkan untuk membuat jalur baru antar *end point* untuk layanan VPLS data dibutuhkan.

Pada saat terjadi kegagalan link atau jalur utama yang dilewati paket diputus pada VPLS SDN maka akan terjadi komunikasi langsung antara *switch* dan *controller*. Ketika ada link yang terputus *switch* akan mengirimkan laporan kepada *controller* dengan mengirimkan paket *port status* yang mengidentifikasikan bahwa ada kondisi dan konfigurasi *port* yang berubah, kemudian *controller* akan memberikan respon dengan mengirimkan paket *flow mod* kepada *switch* yang berisikan modifikasi *flow* untuk *reroute* jalur baru yang akan dilewati. Kemudian *controller* akan mengirimkan paket lagi berupa paket *barrier request* untuk memastikan *flow* yang dikirimkan oleh paket *flow mod* sudah sesuai dan tidak ada kemungkinan *collison* dengan *flow* lainnya pada jalur yang baru. Jika tidak ada respon dari *switch* maka *controller* akan mengirimkan paket *flow mod* baru dan paket *barrier request* hingga *switch* mengirimkan respon kepada *controller* berupa paket *barrier reply* yang menandakan bahwa *flow* yang baru sudah sesuai dan bisa dilewati. Kemudian dari *controller* akan mengirimkan paket *flow removed* kepada *switch* untuk menghapus *flow* awal saat terjadi pemutusan jalur. Sedangkan pada VPLS legasi dengan distribusi yang menyebar waktu untuk *failover* menjadi lebih lama, hal ini disebabkan ketika terjadi kegagalan *link router* akan melakukan komunikasi dengan semua *router* dan mencari jalur yang sesuai sebagai jalur alternatif dengan melakukan estimasi jalur sesuai algoritma *routing* yang digunakan.



Gambar 9. Rata – Rata *Failover Delay*

Pada Gambar 9 ditampilkan *failover delay* rata – rata dari masing – masing durasi terjadinya kegagalan *link*. Waktu *failover* VPLS legasi yang terbilang lama jika dibandingkan dengan waktu *failover* VPLS SDN juga disebabkan karena adanya proses enkapsulasi dan dekapsulasi data ketika mencari jalur alternative baru.

Pada VPLS legasi terdapat beberapa komponen yang berhubungan dengan proses enkapsulasi dan dekapsulasi data yaitu *pseudowire* (PW), *forwarding database* (FDB), dan *service destination point* (SDP). Ketika terjadi pemutusan *link,* FDB akan mengecek kembali MAC *address* dari paket yang akan dikirimkan ke sisi *client*, lalu memasukan MAC *address* yang sudah dicek tadi ke dalam MAC *table*. Kemudian FDB akan mencari MAC *address* tujuan dari paket yang jalurnya mengalami pemutusan *link* tadi. Pada tahap mencari MAC *address* tujuan paket, ketika MAC *address* tujuan tidak ditemukan pada *router* PE terdekat dari *host* paket yang akan dikirimkan maka *pseudowire* akan

melakukan enkapsulasi data dan kemudian membroadcast paket ke PE lainnya. Pada setiap *router* PE yang menerima *broadcast*, FDB pada PE tersebut akan mengecek MAC *address* paket yang sampai dan memasukannya ke dalam MAC *table*. Proses ini terjadi pada setiap PE yang menerima *broadcast* paket tadi. Setelah menemukan PE yang sesuai, paket akan diteruskan ke *router customer edge* (CE) tujuan. Pada *router* CE terjadi proses dekapsulasi data sehingga format paket yang dikirimkan sama seperti format paket awal ketika dikirimkan dari sisi *host*. Jika *router* CE bukan tujuan dari paket tadi maka paket akan otomatis di *drop*, namun apabila paket sampai di *router* CE dan saat pengecekan MAC *address* pada MAC *table router* CE memiliki *destination* MAC *address* yang sama dengan MAC *address* tujuan paket maka *router* CE tersebut akan menerima paket dan meneruskannya ke sisi *client* yang merupakan *service destination point* (SDP).

3. Kesimpulan

1. Penerapan *Software Defined Networking* (SDN) pada jaringan VPLS mampu meningkatkan *high availability* jaringan jika dibandingkan dengan VPLS legasi.
2. VPLS berbasis SDN memiliki waktu *failover delay* tertinggi 30 detik dengan waktu tersingkat 8 detik sedangkan pada VPLS legasi membutuhkan waktu *failover* yang lebih lama yaitu waktu tertinggi 44 detik dan tersingkat 30 detik.
3. Terjadi penurunan *failover delay* sebesar 44.7 % dengan adanya penerapan kontrol terpusat pada VPLS menggunakan ONOS sebagai SDN *controller*.

Daftar Pustaka

[1]. C. Fancy and T. Mishal, “An Evaluation of Alternative Protokols-Based Virtual Private LAN Service (VPLS)”, *IEEE*, 2017.
[2]. L. Madhusanka, Y. Mika, G. Andrei, “Fast Transmission Mechanism for Secure VPLS Architectures”, *IEEE*, 2017.
[3]. H. Ye, Z. Song, Q. Sun, “Device performance analysis of cloud computing data center two-layer interconnection model based on MPLS/IP core backbone network”, *IEEE*, 2014.
[4]. R. Smeliansky V. Pashkov, A. Shalimov. *Controller* failover for sdn enterprise networks. *IEEE*, 2014.
[5]. Open Network Foundation.(2012). Software-Defined Networking: The New Norm for Networks
[6]. SDN Architecture Overview, Open Networking Foundation ONF, December, 2013.
[7]. L. Madhusanka, Y. Mika, G. Andrei, “Software Defined VPLS Architecture: Opportunities and Challenges”, *IEEE*, 2017.