

## **CONGESTION CONTROL PADA JARINGAN KOMPUTER BERBASIS MULTI PROTOCOL LABEL SWITCHING (MPLS)**

**Ida Nurhaida**

Fakultas Ilmu Komputer, Program Studi Teknik Informatika  
Universitas Mercubuana  
Email: ida.nurhaida@mercubuana.ac.id

**Ichsan**

Fakultas Ilmu Komputer, Program Studi Teknik Informatika  
Universitas Mercubuana  
Email: 41517110163@student.mercubuana.ac.id

### **ABSTRAK**

Pada saat jaringan *MPLS* mengalami kondisi *network congestion* di lintasannya, maka akan terjadi *packet drop* dan menyebabkan nilai *QoS* dari proses pengiriman data akan menurun. Penelitian ini dilakukan untuk menguji metode *congestion control* dengan penerapan *QoS-Policies* pada jaringan *Multi Protocol Label Switching (MPLS)*. Parameter *QoS (Quality of Services)* yang diuji dalam penelitian ini berupa *delay*, *throughput*, *jitter* dan *packet loss*. Nilai-nilai yang didapatkan dari parameter tersebut kemudian dibandingkan dengan standar dari *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON)* dengan tujuan untuk mengetahui kualitas layanan pengiriman data pada jaringan *MPLS* ketika terjadi *network congestion* di lintasannya. Setelah melakukan proses perancangan, pengujian dan analisa, hasil yang didapat menunjukkan peningkatan nilai-nilai parameter *QoS*. Nilai *QoS* untuk parameter *delay* mengalami penurunan sebesar 48.3%, nilai *throughput* mengalami peningkatan sebesar 87.44%, nilai *jitter* mengalami penurunan nilai sebesar 54.04% dan nilai *packet loss* mengalami penurunan sebesar 99.9%.

**Kata kunci:** *MPLS; congestion; QoS.*

### **ABSTRACT**

When the *MPLS* network experiences *network congestion* on its path, *packet drop* will occur and the *QoS* value of the data transmission process will decrease. This research was conducted to test the *congestion control* method by implementing *QoS-Policies* on the *Multi Protocol Label Switching (MPLS)* network. *QoS (Quality of Services)* parameters tested in this study are *delay*, *throughput*, *jitter* and *packet loss*. The values obtained from these parameters are then compared with the standards of *Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON)* with the aim of knowing the quality of data transmission services in *MPLS* networks when *network congestion* occurs in its location. After carrying out the design, testing and analysis process, the results obtained showed an increase in the values of the *QoS* parameters. The *QoS* value for the *delay* has decreased by 48.3%, the *throughput* value has increased by 87.44%, the *jitter* value has decreased by 54.04% and *packet loss* value has decreased by 99.9%.

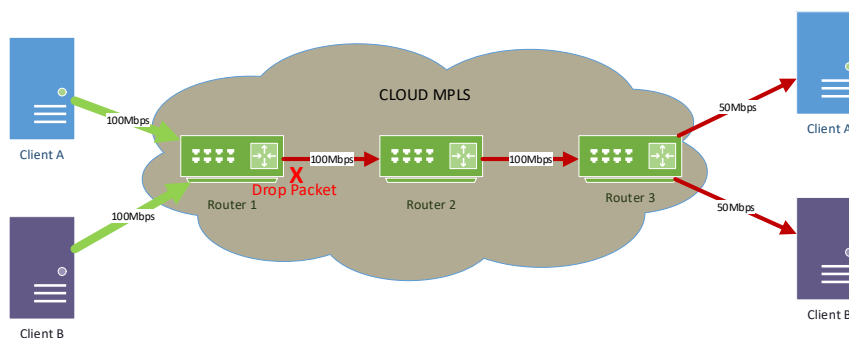
**Keywords:** *MPLS; congestion; QoS.*

### **1. PENDAHULUAN**

Teknologi pengiriman paket data dalam jaringan komputer digunakan untuk kebutuhan yang semakin luas dan juga kompleks secara global. Tidak hanya mengirimkan paket data dari satu komputer ke komputer lainnya, namun saat ini jaringan komputer dituntut untuk dapat melayani banyak *host* dalam satu waktu dengan jarak pengiriman yang semakin beragam.

*MPLS (Multi Protocol Label Switching)* merupakan jawaban dari kebutuhan teknologi pengiriman paket data saat ini, *MPLS* memiliki kemampuan membentuk *tunnel* atau *virtual circuit* yang melintasi *network*. Prinsip kerjanya menggabungkan beberapa keuntungan dari sistem komunikasi *circuit-switched* dan *packet-switched*. kemampuan inilah yang memberikan solusi arsitektur jaringan yang dapat digunakan secara massif dengan skala yang besar. *MPLS* merupakan teknologi terbaru yang digunakan oleh banyak *provider* untuk mengirim paket data pada jaringan *backbone* berkecepatan tinggi tanpa mengubah struktur jaringan yang telah ada sebelumnya. Sekarang ini banyak perusahaan yang jaringannya masih dilakukan secara manual mulai beralih memakai jaringan *MPLS* karena dianggap *low cost* dan *high reliability*(1).

Ketika arsitektur *MPLS* dipakai secara massif dan berkepanjangan sebagai *backbone* untuk pengiriman data, tentu terdapat peningkatan arus trafik yang melewati jaringan *MPLS* tersebut. Pertumbuhan trafik yang tidak diimbangi dengan penambahan kapasitas skala jaringan *MPLS* akan mengakibatkan *network congestion* di beberapa titik di sepanjang jaringan *MPLS*. *Network congestion* adalah kondisi dimana trafik yang lewat melebihi kapasitas dari jaringan tersedia, sebagai contoh terdapat aliran trafik sebesar 200Mbps sedangkan kapasitas jaringan yang dilewati hanya 100Mbps. Dengan kondisi tersebut maka akan terdapat *packet drop* yang menyebabkan *packet loss*. Ilustrasi kondisi *network congestion* ditunjukkan oleh gambar 1 dibawah.



**Gambar 1. Kondisi *Network Congest***

*Congestion Control* terdiri dari dua bidang yaitu *TCP Congestion Control* dan *Active Queue Management*. Keduanya mempunyai mekanisme berbeda. *Active Queue Management* adalah *congestion control* yang diterapkan di jaringan yaitu di *router device*(2). Pada penelitian ini digunakan *congestion control Active Queue* dengan memprioritaskan paket yang keluar dari *router* menggunakan *forwarding class* tertentu.

Kehandalan jaringan merupakan salah satu isu yang sering dibicarakan untuk jaringan *best-effort (IP Based)*. Oleh karena itu teknologi *Multi Protocol Label Switching (MPLS)* digunakan untuk meningkatkan performansi jaringan. *MPLS* bekerja dengan cara menambahkan *header/label* pada paket sebagai identifikasi yang akan digunakan pada proses *switching*. Namun teknologi *MPLS* ini tetap tidak dapat memperhatikan kondisi jaringan. Sehingga apabila terjadi *congest* di jaringan tersebut maka tidak ada mekanisme tertentu untuk mengalihkan trafik ke jalur lain. *Congestion* tidak hanya terjadi pada jaringan *MPLS* saja. Semua teknologi pengiriman data mempunyai masalah yang sama. Seperti di bahas oleh Faishal H, Saputra dkk(3) melakukan survei mekanisme *congestion control* pada jaringan *software defined network (SDN)*.

Semakin beragamnya layanan, mengakibatkan semakin banyak juga jenis paket yang akan dilewatkan di jaringan. *MPLS* mempunyai fitur *QoS policies* dapat menjadi solusi untuk menjaga *QoS* tiap layanan yang bekerja dengan cara mengelompokkan aliran trafik tertentu dalam sejumlah kelas-kelas trafik agar dapat memberikan perlakuan yang berbeda sesuai dengan standar kualitas setiap layanan. *MPLS QoS policies* merupakan metode *congestion control* yang dapat digunakan untuk membedakan perlakuan terhadap paket sesuai dengan jenis layanannya dan dapat meminimalisir efek dari *network congestion*.

Dalam memitigasi *network congestion* terdapat beberapa metode yang umum dilakukan salah satunya adalah *TCP congestion control mechanis*. *TCP* merupakan protokol yang terdapat dalam *OSI Layer* di layer *transport* yang mendukung *transfer mechanism* data yang *reliable*, sehingga aliran data yang dibaca *TCP receiver* berurutan, utuh, tanpa duplisasi dan tidak rusak. Untuk menyediakan *transfer data* yang *reliable*, *TCP* menyediakan layanan *flow control* dan *error checking*. Ketika *TCP sender* mengirim data lebih cepat daripada yang bisa ditangani *TCP receiver*, maka akan terjadi *congestion*(4).

Namun ketika *congestion* terdapat di lintasan *MPLS* maka *TCP congestion control mechanism* tidak dapat digunakan karena *TCP* berada pada layer *transport*. Dalam lintasan *MPLS* sebuah *packet* hanya akan dibuka dan diperiksa sampai *MPLS header* yang letaknya diantara *data link layer* dan *network layer* pada setiap *router* dalam lintasan *MPLS*. Metode yang tepat untuk memitigasi *congestion control* pada jaringan *MPLS* dengan menggunakan *Active Queue Management (AQM)*.

*Active Queue Management (AQM)* merupakan metode untuk mengatur *congestion* dan meningkatkan *QoS*. *AQM* mendukung jaringan *MPLS* dengan memanfaatkan mekanisme pengelompokkan *DiffServ*, memberikan tanda prioritas pada paket yang akan dikirimkan dalam jaringan. Sebelum paket dikirimkan melalui jaringan *MPLS* setiap paket akan diberi *label DiffServ* pada *MPLS header*.

### 1.1 MPLS (Multi Protocol Label Switching)

MPLS adalah teknologi *label-switching* yang menggabungkan kemampuan rekayasa trafik ATM dengan fleksibilitas dan skalabilitas jaringan IP. MPLS memiliki kemampuan membentuk *tunnel* atau *virtual circuit* yang melintasi *network*. Prinsip kerjanya menggabungkan beberapa keuntungan dari sistem komunikasi *circuit-switched* dan *packet-switched*. MPLS memiliki arsitektur *packet switching* dan routing yang sangat baik dan dapat mengirimkan data dengan sangat cepat(5). Dalam jaringan MPLS juga mampu memisahkan setiap trafik *customer* yang berbeda walaupun menggunakan infrastruktur *backbone* secara bersamaan(6).

### 1.2 Quality of Service (QoS)

QoS adalah kemampuan untuk memberikan jaminan *resources* dan diferensiasi layanan dalam jaringan(7). Prinsip kerjanya memberikan batasan-batasan tertentu dalam sebuah layanan dalam jaringan. Terdapat beberapa parameter untuk mengukur kualitas QoS diantaranya *throughput*, *packet loss* dan *jitter*. QoS saat ini menjadi parameter utama untuk mengukur kehandalan dari jaringan komputer, semakin tinggi nilai QoS yang didapat maka semakin handal jaringan komputer tersebut. Pada beberapa tahun sebelumnya teknologi pengiriman data ATM sangat banyak digunakan untuk mengirim data mendapat nilai QoS yang tinggi(8), namun terlalu rumit dan mahal untuk mengimplementasikan teknologi tersebut.

#### 1.2.1 Delay

Delay adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan (9). Tujuan pengukuran *delay* untuk mengetahui berapa lama waktu yang dibutuhkan untuk satu paket sampai dari sumber ke tujuan, pada penelitian ini *delay* yang di ukur merupakan *one-way delay*. Nilai *delay* yang ditampilkan dalam penelitian ini merupakan nilai rata-rata *delay* dari sepuluh kali percobaan di setiap skenario percobaan. Nilai *delay* yang sangat bagus menurut standar TIPHON adalah < 150 ms. Tabel 1 menunjukkan nilai *Delay* menurut standar TIPHON:

Tabel 1. Nilai *Delay* Standar TIPHON(10)

Category	Delay
Best	< 150 ms
Good	150 ms - 300 ms
Medium	300 ms - 450 ms
Poor	> 450 ms

#### 1.2.2 Throughput

Throughput dapat diartikan sebagai jumlah data per satuan waktu yang dikirim di dalam sebuah jaringan, dari suatu titik jaringan ke titik jaringan yang lain(11). Tujuan pengukuran *throughput* adalah untuk mengetahui kehandalan jaringan dalam meneruskan paket yang datang hingga sampai di tujuan. Pengukuran dilakukan dengan melakukan pengiriman paket *end to end* melewati jaringan MPLS yang sudah terdapat *network congestion*, dengan menggunakan *background* trafik 10 Mbps, 50 Mbps dan 100 Mbps selama satu menit sebanyak 10 kali percobaan di setiap skenarionya. Nilai *throughput* yang ditampilkan pada hasil merupakan nilai rata-rata dari setiap percobaan. Tabel 2 menunjukkan nilai *Throughput* menurut standar TIPHON:

Tabel 2. Nilai *Throughput* Standar TIPHON(10)

Category	Throughput
Best	> 2101 Kbps
Better	1201-2100 Kbps
Good	701-1200 Kbps
Medium	339-700 Kbps
Poor	0-338 Kbps

#### 1.2.3 Jitter

*Jitter* didefinisikan sebagai variasi *delay* yang diakibatkan oleh panjang antrean dalam suatu pengolahan data dan *reassemble* paket-paket data di akhir pengiriman akibat kegagalan sebelumnya(11).

*Jitter* merupakan masalah yang khas pada *connectionless* atau *packet switch network*. Tujuan pengukuran *jitter* adalah untuk mengetahui kestabilan *forwarding* data dalam suatu jaringan. Kestabilan bisa dilihat dari banyaknya variasi *delay* yang terjadi selama waktu komunikasi di jaringan. Berdasarkan standar *TIPHON* nilai *jitter* yang sangat bagus adalah 0 s/d 75 ms. Tabel 3 menunjukkan nilai *Jitter* menurut standar *TIPHON*:

**Tabel 3. Nilai *Jitter* Standar *TIPHON* (12)**

<i>Category</i>	<i>Jitter</i>
<i>Perfect</i>	0 ms
<i>Good</i>	75 ms
<i>Medium</i>	125 ms
<i>Poor</i>	225 ms

#### 1.2.4 Packet Loss

*Packet Loss* adalah parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah paket data yang hilang karena *collusion* atau *congest* didalam jaringan(13). Satuan yang dipakai adalah persen (%). Tujuan pengukuran *packet loss* dilakukan untuk melihat kehandalan metode yang digunakan dalam pengiriman paket pada saat terjadi *network-congestion*. Berdasarkan standar *TIPHON* nilai *packet loss* yang sangat bagus adalah 0%. Tabel 4 menunjukkan nilai *Packet Loss* menurut standar *TIPHON*:

**Tabel 4. Nilai *Packet Loss* Standar *TIPHON* (12)**

<i>Category</i>	<i>Packet Loss</i>
<i>Perfect</i>	0 %
<i>Good</i>	3 %
<i>Medium</i>	15 %
<i>Poor</i>	25 %

#### 1.3 Forwarding-Class (FC)

*Router* memiliki kemampuan membagi beberapa *forwarding-class*, cara kerjanya mengatur dan menangani bagaimana mekanisme paket di antrekan, antrean dibuat menurut prioritas masing-masing *forwarding-class* sebelum paket diteruskan ke dalam *switch-fabric*. *Router* memiliki delapan kelas *forwarding-class*, seperti yang ditunjukkan pada tabel 5 dibawah:

**Tabel 5. Tipe *Forwarding Classes* (14)**

FC-ID	FC Name	FC Designation	DiffServ Names
7	<i>Network Control</i>	NC	NC2
6	<i>High-1</i>	H1	NC1
5	<i>Expedited</i>	EF	EF
4	<i>High-2</i>	H2	AF4
3	<i>Low-1</i>	L1	AF2
2	<i>Assured</i>	AF	AF1
1	<i>Low-2</i>	L2	CS1
0	<i>Best Effort</i>	BE	BE

Secara umum *forwarding-class* terbagi menjadi tiga kategori utama yaitu, *High-Priority*, *Assured* dan *Best-Effort*. Dalam kategori *high-priority* terdapat empat kelas yaitu, *network-controller* (nc), *expedited* (ef), *high 1* (h1), *high 2* (h2). Kategori *high-priority* selalu menjadi prioritas teratas untuk dapat dilewatkan paket ketika terjadi *network congestion*. Pada kategori *assured* terdapat dua kelas yaitu, *assured* (af) dan *low 1* (l1). Kategori *assured* hanya menjamin paket sampai ketujuan jika *bandwith* dalam jaringan masih memungkinkan dan tidak terdapat kelas yang lebih tinggi di atasnya. Pada kategori *best-effort* terdapat dua kelas yaitu, *low 2* (l2) dan *best-effort* (be). Kelas *best-effort* tidak memiliki jaminan pengiriman, semua paket dalam kelas ini dikirim sesuai kemampuan *bandwith*. Kelas *best-effort* juga menjadi kelas *default* bagi pengiriman paket-paket *MPLS*.

## 1.4 Router

Router bertugas meneruskan dan menentukan rute paket ke seluruh jaringan dari sumber ke tujuan akhirnya, proses tersebut dikenal dengan istilah *routing*(15). Proses *routing* terjadi pada lapisan 3 dari struktur *OSI* dengan cara melihat alamat network dari setiap paket yang diproses, apabila sebuah paket mempunyai tujuan ke alamat *host network* yang berbeda, maka paket akan diteruskan ke alamat *network* tersebut, dan jika sebuah paket mempunyai tujuan ke alamat *network* yang sama maka *router* akan menghalangi paket tersebut keluar. *Router* menggunakan *IP address* tujuan untuk mengirimkan paket, dan agar *router* mengetahui rute mana yang harus digunakan untuk meneruskan paket ke alamat tujuan, *router* harus belajar atau bertukar informasi sesama *router* yang saling terhubung untuk mengetahui jalur atau rute yang terbaik(16).

## 2. METODE PENELITIAN

Metode desain jaringan yang digunakan dalam penelitian ini adalah *PPDIOO* (*Prepare, Plan, Design, Implement, Operate and Optimize*) yang juga dikenal dengan *network lifecycle* merupakan sebuah metode penelitian yang dikembangkan oleh *Cisco System.Inc* (17).

### 2.1 Prepare

*Prepare* merupakan tahapan awal proses penelitian untuk menyusun rencana kerja agar penelitian dapat berjalan dengan baik. Pada tahapan ini rencana terkait hal-hal yang berhubungan dengan analisa pokok pembahasan seperti permasalahan yang muncul, analisa kebutuhan penelitian dalam *MPLS* dan *congestion control*, baik itu dari segi *hardware* ataupun *software* dan analisa topologi jaringan yang akan dibangun.

#### 2.1.1 Perangkat Keras

Dalam penelitian ini perangkat keras yang digunakan adalah sebagai berikut:

- Router*: *Router* berfungsi sebagai pengatur lalu lintas jaringan komputer, pada penelitian ini digunakan empat unit *router* merk *NOKIA* tipe *7210 SAS-M* yang dapat mengaplikasikan konfigurasi jaringan *MPLS* sesuai dengan standar *RFC-3031*
- SFP* (*small form-factor pluggable*): *SFP* berfungsi sebagai media *interface* di *router* untuk digunakan sebagai *interface* ke arah akses maupun ke arah *network*. Pada penelitian ini digunakan total dua belas unit *SFP* dengan pembagian empat unit *SFP* tipe *TX* dan delapan unit *SFP* tipe *LX* dengan konektor tipe *LC*.
- PC Laptop*: *Laptop* pada penelitian ini digunakan sebagai *node client* dalam jaringan *MPLS*, *laptop* juga berfungsi untuk menjalankan *software* untuk menguji kualitas layanan. *Laptop* yang digunakan dalam penelitian ini berjumlah 2 unit.
- Kabel *Patchcord*: *Patchcord* pada penelitian ini digunakan sebagai media penghubung *interface network* antar *router*, dibutuhkan sebanyak empat unit *patchcord single-mode* dengan tipe konektor *LC-LC*.
- Kabel *UTP* (*Unshielded twisted-pair*): Pada penelitian ini kabel *UTP* digunakan sebagai media penghubung *interface* akses dari *router* menuju *laptop*. Dibutuhkan empat unit kabel *UTP* dengan tipe *CAT-6*.

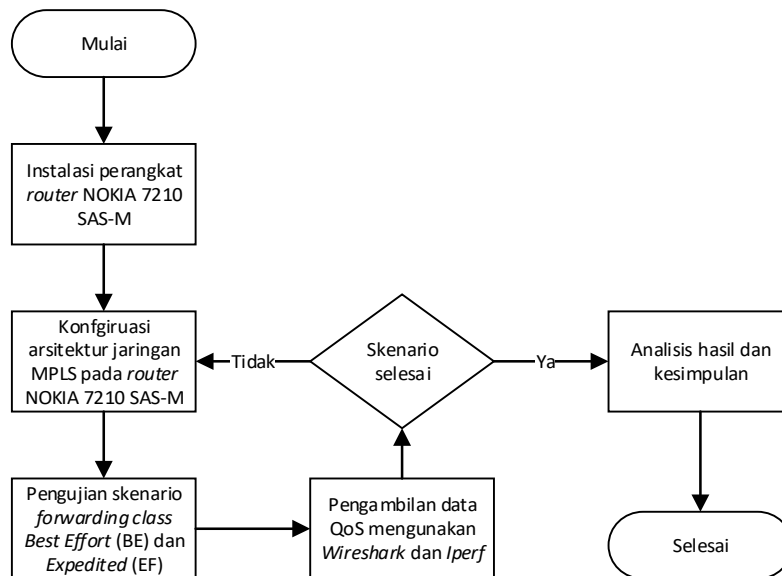
#### 2.1.2 Perangkat Lunak

Dalam penelitian ini perangkat lunak yang digunakan adalah sebagai berikut:

- Sistem Operasi *Router*: *Router nokia* memakai sistem operasi *Timetra Operating-System* (*TiMOS*), pada penelitian ini digunakan versi *TiMOS-B-7.0.R5*.
- Sistem operasi *PC Laptop*: *Laptop* yang digunakan dalam penelitian ini menggunakan sistem operasi *windows 8*.
- iperf V2*: Berfungsi untuk melakukan *inject* trafik *end to end* antar *laptop* pada jaringan *MPLS* dengan skema *background* trafik *10 Mbps*, *50 Mbps* dan *100 Mbps*. *Iperf* juga berfungsi untuk mengambil nilai *throughput*, *jitter* dan *packet loss* dalam setiap skenario pengujian.
- Wireshark V3.0.0*: Berfungsi melakukan pengamatan pengukuran nilai *delay*.

## 2.2 Plan

Tahapan *plan* merupakan perencanaan jaringan yang dibuat baik segi *software* atau segi *hardware* yang dibutuhkan beserta rencana skenario pengujian yang dilakukan pada penelitian. Berikut *flowchart diagram* yang menjelaskan tahapan perancangan penelitian:

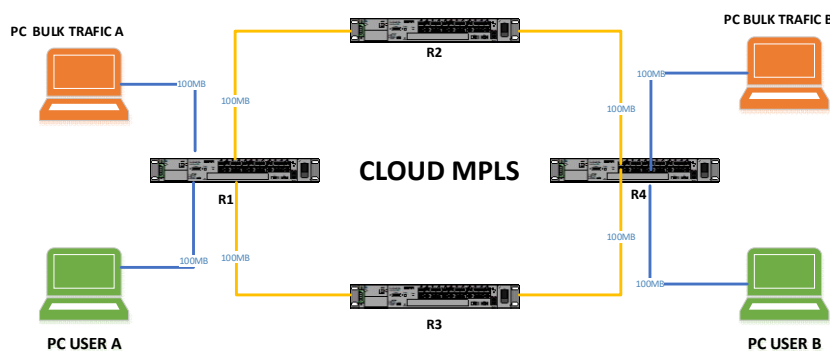


Gambar 2. Flowchart Perancangan Penelitian

Gambar 2 menunjukkan alur perancangan penelitian dimulai dengan instalasi perangkat *router nokia*, konfigurasi arsitektur *MPLS* sehingga semua *router* terhubung satu sama lain. Pengujian skenario *congestion control* menggunakan *QoS policies best effort* dan *expedited*, pengambilan data parameter *QoS* menggunakan *Wireshark* dan *Iperf*. Setelah data diambil maka proses selanjutnya menganalisa hasil dan menarik kesimpulan.

## 2.3 Design

Tahapan *design* merupakan tahapan membuat topologi jaringan yang akan diimplementasikan pada penelitian. Gambar 3 menunjukkan topologi yang dibangun pada penelitian ini :



Gambar 3. Topologi Jaringan

Topologi jaringan *MPLS* yang dibangun menggunakan empat unit *router nokia 7210 SAS-M* dengan sistem operasi *TiMOS* versi *TiMOS-B-9.0.R9*, tabel 6 menunjukkan pengalamatan perangkat *backbone MPLS*.

**Tabel 6. Tabel pengalaman *device backbone MPLS***

Perangkat	IP Address
R1	10.10.10.1/32
R2	10.10.10.2/32
R3	10.10.10.3/32
R4	10.10.10.4/32

*Router* dibentuk dengan topologi *ring* dimana masing-masing *router* memiliki dua *interface* ke arah *router* lainnya dengan kapasitas masing-masing *interface* sebesar 100 Mbps, tabel 7 menunjukkan pengalaman *interface*.

**Tabel 7. Tabel pengalaman *interface***

Perangkat	Port	Type	IP Address	Nama Interface	Protokol
R1	1/1/1	network	192.168.1.1/30	to-r2	OSPFv2, MPLS, RSVP, LDP
R1	1/1/2	network	192.168.1.9/30	to-r4	OSPFv2, MPLS, RSVP, LDP
R2	1/1/1	network	192.168.1.2/30	to-r1	OSPFv2, MPLS, RSVP, LDP
R2	1/1/3	network	192.168.1.13/30	to-r3	OSPFv2, MPLS, RSVP, LDP
R3	1/1/3	network	192.168.1.14/30	to-r2	OSPFv2, MPLS, RSVP, LDP
R3	1/1/1	network	192.168.1.21/30	to-r4	OSPFv2, MPLS, RSVP, LDP
R4	1/1/2	network	192.168.1.10/30	to-r1	OSPFv2, MPLS, RSVP, LDP
R4	1/1/1	network	192.168.1.22/30	to-r3	OSPFv2, MPLS, RSVP, LDP

Untuk melakukan *inject* trafik dan melakukan percobaan *congestion control* digunakan empat unit laptop *PC* dengan sistem operasi *Windows* versi 8. Dua unit *PC* mempunyai koneksi ke *router* R1 dan dua unit *PC* mempunyai koneksi ke R4, koneksi antara *PC* - *router* menggunakan kabel *UTP* tipe *CAT 6*.

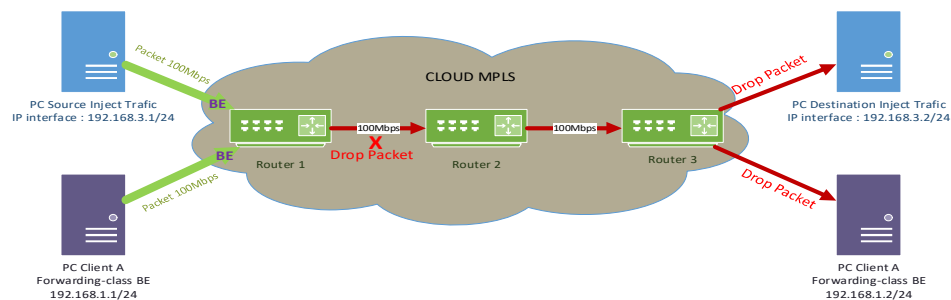
#### 2.4 Implement

*Implement* adalah tahap yang dilakukan sesuai pada topologi tahapan *design*. Mulai dari mempersiapkan arsitektur jaringan, melakukan instalasi *software* pada komputer laptop dan konfigurasi empat unit *router* nokia sebagai *backbone MPLS*. Melakukan *inject* trafik menggunakan aplikasi *iperf*, menganalisa hasil menggunakan aplikasi *wireshark* dan melakukan konfigurasi *QoS policies* pada *router*.

#### 2.5 Operate

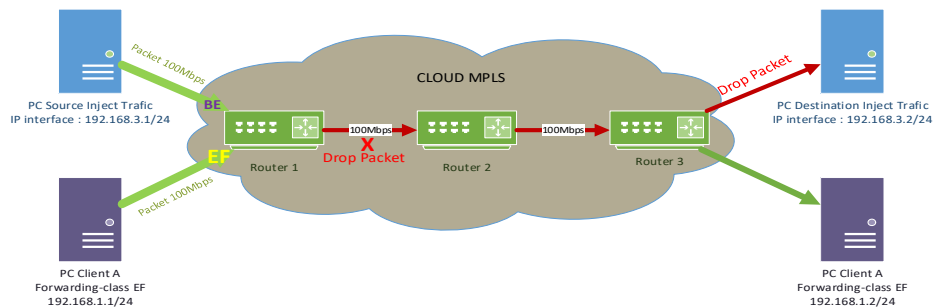
*Operate* merupakan tahap melakukan percobaan skenario yang telah disiapkan. Percobaan yang dilakukan dengan mengalirkan *traffic bulk* dari *source* ke *destination* menggunakan *software iperf* sehingga membuat lintasan *MPLS* terdapat *network congestion*. Pada *host* laptop *PC* User A dan B melakukan *capture wireshark* untuk menguji *QoS* dari skenario yang sudah ditetapkan dan pemecahan masalah yang timbul selama proses pengambilan data.

Dalam menguji kehandalan jaringan *MPLS* ketika terjadi *network congestion* di beberapa titik di sepanjang lintasan, maka dalam pengambilan data ada empat parameter *QoS* yang umum dilakukan pengukuran yaitu *delay*, *throughput*, *jitter* dan *packet loss*(18).



**Gambar 4. Topologi Dengan Skenario *Forwarding Class BE (Best-Effort)***

Pada dasarnya proses pengiriman dalam jaringan *MPLS* secara *default* menggunakan *forwarding class best effort*, seperti terlihat pada gambar 4, masing-masing *PC* akan mengirimkan paket sebesar 100Mbps, namun dengan kapasitas *interface* antar *router* yang hanya mampu melewati paket sebesar 100Mbps maka *router* akan melakukan *drop* paket dari kedua *PC* dikarenakan *forwarding class* yang digunakan padat paket tersebut merupakan *best effort class*.



**Gambar 5. Topologi Dengan Skenario Forwarding Class EF (Expedited)**

Pada paket dengan *forwarding class expedited* (EF) yang ditunjukkan pada gambar 5, proses antrean paket selama proses pengiriman mendapatkan prioritas selama paket yang dikirim tidak melebihi dari kapasitas *interface* antar *router* dan paket-paket lain memiliki *forwarding class* lebih rendah.

Poin utama yang diuji dalam penelitian ini yaitu menganalisa paket yang dilewati dalam jaringan *MPLS* yang telah terdapat *network congestion* di sepanjang lintasan menggunakan metode *congestion control* dengan *forwarding-class Best Effort* (BE) dan *QoS-Policies* menggunakan *forwarding-class Expedited* (EF)

Setiap skenario pengujian pengiriman paket akan diberikan beban *background* trafik sebesar 10 Mbps, 50 Mbps dan 100 Mbps. Beban trafik yang digunakan mereplikasikan trafik *full load*, *medium load* dan *low load* trafik pada jaringan *MPLS*.

## 2.6 Optimize

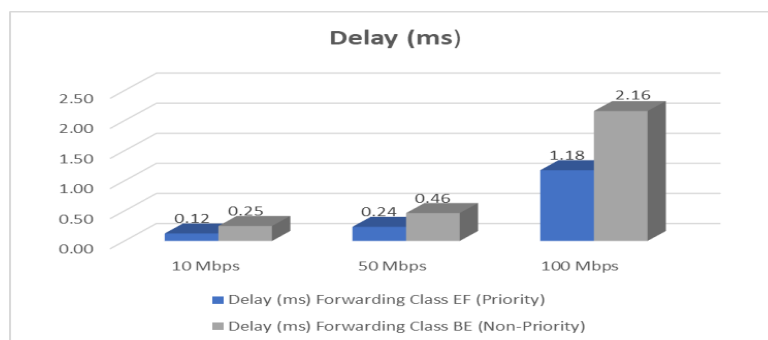
*Optimize* merupakan tahapan optimalisasi setelah melakukan analisa pada tahapan *operate*, tahap ini dapat menyebabkan perlunya desain ulang jaringan apabila terlalu banyak masalah jaringan yang timbul atau kinerja tidak sesuai dengan yang diharapkan(19)

Setelah semua percobaan pada penelitian ini dilakukan dan didapatkan data percobaan, tahapan *optimize* tidak dilakukan dikarenakan tidak adanya masalah yang memerlukan desain ulang terhadap arsitektur jaringan *MPLS* yang telah dirancang.

## 3. HASIL DAN PEMBAHASAN

Pada penelitian ini setelah melalui tahapan perancangan jaringan, konfigurasi jaringan *MPLS*, uji coba penerapan metode *congestion control* dengan menggunakan *QoS Polices* maka didapatkan nilai *QoS* dari masing-masing skenario pengujian, nilai *QoS* yang di uji pada penelitian ini berupa *delay*, *throughput*, *jitter* dan *packet loss*.

### 3.1 Delay



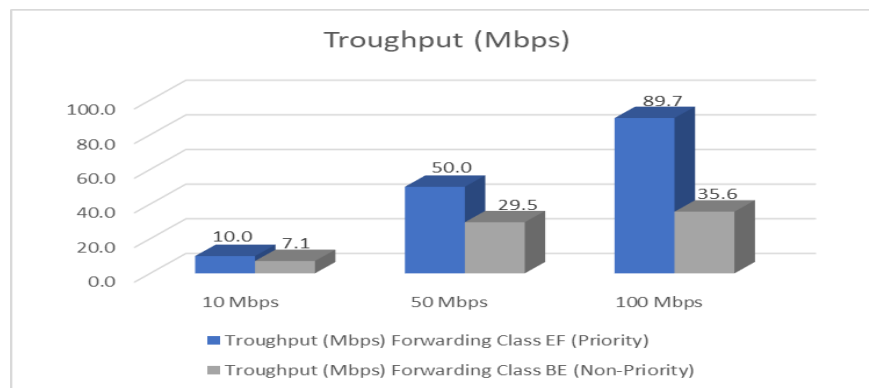
**Gambar 6. Grafik Pengujian Delay**



Pada hasil pengujian *delay* menggunakan *forwarding-class* EF (*priority*) dan *forwarding-class* BE (*non-priority*) rata-rata dari sepuluh kali percobaan nilai *delay* dapat mencapai kategori *best* menurut standar *TIPHON*. Seperti yang ditunjukkan pada gambar 6 pengujian dengan *background* trafik 10 Mbps parameter *delay* menunjukkan nilai 0.17 ms untuk *forwarding-class* EF dan 0.25 ms untuk *forwarding-class* BE, pada skenario pengujian dengan *background trafik* 50 Mbps *delay* yang dihasilkan 0.24 ms untuk *forwarding-class* EF dan 0.46 ms untuk *forwarding-class* BE, pada pengujian dengan *background trafik* 100 Mbps *delay* menunjukkan nilai 1.18 ms untuk *forwarding-class* EF dan 2.16 ms untuk *forwarding-class* BE.

Dari hasil pengujian ini menunjukkan bahwa paket-paket yang diberikan label *forwarding class* EF (*priority*) mengalami peningkatan serta mendapatkan nilai *delay* yang lebih kecil dibandingkan dengan paket-paket dengan *forwarding class* BE (*non-priority*). Selain karena *forwarding class* yang diberikan pada paket *MPLS*, *delay* dapat dipengaruhi oleh jarak, media fisik, kondisi *congest* atau juga waktu proses yang lama. Titik-titik ini dapat berupa perangkat komputer, atau perangkat jaringan lainnya seperti *router*, *modem* dan sebagainya yang dilewati oleh paket(20).

### 3.2 Troughput



Gambar 7. Grafik Pengujian Troughput

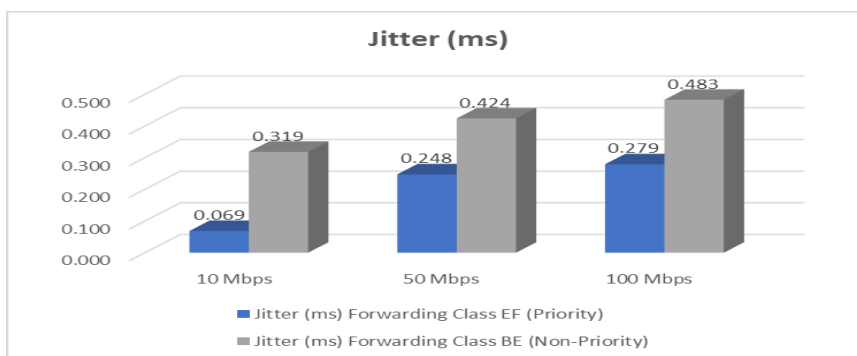
Seperti yang ditunjukkan pada gambar 7, hasil pengujian *troughput* dengan aliran trafik yang menggunakan *forwarding-class* EF (*priority*) rata-rata dapat mencapai nilai kategori *best* menurut standar *TIPHON*, pada *background trafik* 10 Mbps *troughput* mencapai nilai maksimal 10 Mbps, pada skenario pengujian 50 Mbps *troughput* yang dihasilkan mencapai nilai maksimal 50 Mbps sedangkan pada *background trafik* 100 Mbps *troughput* tidak mencapai angka maksimal 100 Mbps hanya mencapai angka 89.7 Mbps, hal tersebut disebabkan karena keterbatasan *resource* dari kapasitas interface antar router sebesar 100 Mbps.

Pada hasil pengujian *troughput* menggunakan *forwarding-class* BE (*non-priority*) di *background trafik* 10 Mbps *troughput* yang dihasilkan berada angka 7.1 Mbps, di *background trafik* 50 Mbps nilai *troughput* yang dihasilkan berada di angka 29.5 Mbps dan di *background trafik* 100 Mbps nilai *troughput* yang dihasilkan berada di angka 35.6 Mbps

Dari hasil pengujian ini menunjukkan bahwa paket-paket yang diberikan label *forwarding class* EF (*priority*) dapat melewati *network congestion* dan menjamin paket diterima dengan nilai *troughput* maksimal jika tidak terdapat keterbatasan *resource* pada jaringan. Sedangkan paket-paket yang diberikan label *forwarding class* BE (*non-priority*) jika melewati *network congestion* maka nilai *troughput* yang dihasilkan mengalami degradasi dan menjadi tidak maksimal.

Peningkatan sangat terlihat pada paket dengan *forwarding-class* EF (*priority*) dari ketiga skenario *background trafik* yang dilakukan pengujian, dua di antaranya dapat dicapai dengan hasil yang sangat baik, hal tersebut di sebabkan oleh proses *queue* yang diberikan oleh *forwarding-class* EF (*priority*) yang memberikan prioritas tinggi pada paket (21).

### 3.3 Jitter

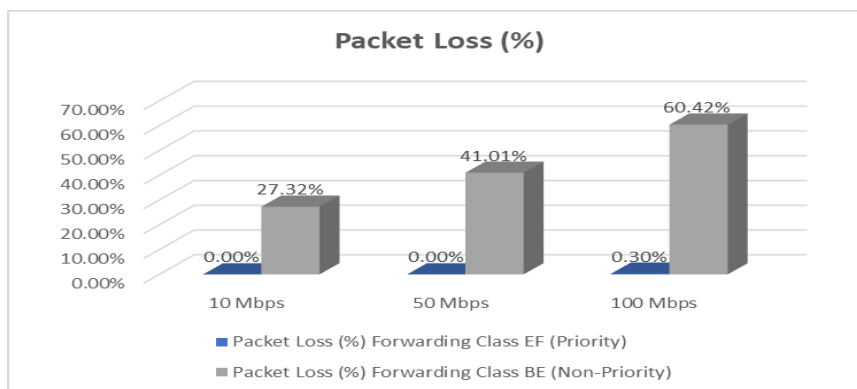


Gambar 8. Grafik Pengujian Jitter

Seperti yang ditunjukkan pada gambar 8, hasil pengujian *jitter* yang didapatkan menunjukkan nilai < 1 ms pada setiap skenario *background* trafik. Pada skenario pengujian dengan *background* trafik 10 Mbps didapatkan angka 0.069 ms untuk *forwarding-class* EF (*priority*) dan 0.319 ms untuk *forwarding-class* BE (*non-priority*), pada skenario dengan *background* trafik 50 Mbps didapatkan angka 0.248 ms untuk *forwarding-class* EF (*priority*) dan 0.424 ms untuk *forwarding-class* BE (*non-priority*), pada skenario *background* trafik 100 Mbps didapatkan angka 0.279 ms untuk *forwarding-class* EF (*priority*) dan 0.483 ms untuk *forwarding-class* BE (*non-priority*).

Dari hasil pengujian ini menunjukkan semua skenario pengujian masuk kategori sempurna menurut standar *TIPHON* untuk nilai *jitter*. Kondisi *jitter* dipengaruhi oleh berbagai macam aspek, salah satunya adalah jarak hop yang dilewati paket dari sumber ke tujuan, semakin banyak hop yang dilewati maka akan semakin besar variasi *jitter* yang dihasilkan(22). Pada pengujian ini semua skenario melewati tiga hop dari sumber hingga sampai ke tujuan hal tersebut yang menyebabkan hasil pengujian setiap skenario menunjukkan hasil sempurna.

### 3.4 Packet Loss



Gambar 9. Grafik Pengujian Packet Loss

Seperti yang ditunjukkan pada gambar 9, hasil pengujian *packet loss* yang menunjukkan nilai yang beragam pada setiap skenario *background* trafik. Pada skenario pengujian dengan *background* trafik 10 Mbps didapatkan angka *packet loss* sebesar 0.00% untuk *forwarding-class* EF (*priority*) dan 27.32% untuk *forwarding-class* BE (*non-priority*), pada skenario dengan *background* trafik 50 Mbps didapatkan angka 0.00% untuk *forwarding-class* EF (*priority*) dan 41.01% untuk *forwarding-class* BE (*non-priority*), pada skenario *background* trafik 100 Mbps didapatkan angka 0.30% untuk *forwarding-class* EF (*priority*) dan 60.42% untuk *forwarding-class* BE (*non-priority*).

Dari hasil pengujian menunjukkan *packet loss* untuk *forwarding-class* EF (*priority*) telah memenuhi standar *TIPHON*, dimana semua hasil pengujian dengan tiga skenario *background* trafik mempunyai nilai kurang dari 3% *packet loss*. Sedangkan untuk *forwarding-class* BE (*non-priority*) dimana semua hasil pengujian dengan tiga skenario *background* trafik rata-rata nilai *packet loss* yang dihasilkan sebesar 42.92% yang artinya sangat buruk menurut standar *TIPHON*.

*Packet loss* bisa terjadi karena faktor bertambah banyaknya pengguna pada suatu jaringan yang saling tarik menarik *bandwidth*(23). Dengan menggunakan *QoS policies* dapat ditentukan *user* atau *host*

mana saja yang mendapatkan prioritas disaat terjadinya *network congestion*, sehingga dapat meminimalisir terjadinya *packet loss*.

#### 4. KESIMPULAN

Berdasarkan hasil penelitian *congestion control* pada jaringan *Multi Protocol Label Switching (MPLS)* selama proses perancangan infrastruktur, pengambilan data dan analisa hasil pada jaringan *MPLS* ketika terdapat *congestion* di sepanjang lintasannya. Maka dari penelitian ini dapat memberikan kesimpulan.

Pengambilan nilai-nilai parameter *QoS* dalam proses penelitian ini dibagi dalam tiga skenario *background* trafik yang mereplikasi kondisi *high* trafik dengan beban trafik 100 Mbps, *medium* trafik dengan beban 50 Mbps dan *low* trafik dengan beban 10 Mbps. Masing-masing skenario di uji dengan dua *forwarding-class* berbeda ketika terjadi *network congestion*. *Forwarding class* pertama menggunakan *forwarding class* BE (*best-effort*) yang kedua dilakukan optimalisasi menggunakan *forwarding-class* EF (*expedited*).

Setelah dilakukan optimalisasi dengan menggunakan *forwarding-class* EF, semua nilai parameter *QoS* yang di uji yaitu *delay*, *throughput*, *jitter* dan *packet loss* mengalami perubahan yang signifikan ketika paket *MPLS* melewati kondisi *network congestion*. Nilai *QoS* untuk parameter *delay* mengalami penurunan sebesar 48.3%, nilai *throughput* mengalami peningkatan sebesar 87.44%, nilai *jitter* mengalami penurunan nilai sebesar 54.04% dan nilai *packet loss* mengalami penurunan sebesar 99.9%.

*Congestion control* dengan menggunakan *forwarding class Expedited* (EF) merupakan hasil terbaik untuk mendapatkan nilai *QoS* yang optimal dan memenuhi standar *TIPHON* ketika terjadi *network congestion* di sepanjang lintasan *MPLS*.

Mekanisme antrean paket pengiriman pada *forwarding class Expedited* (EF) merupakan faktor yang paling mempengaruhi nilai-nilai parameter *QoS* ketika terjadi *network congestion* di sepanjang lintasan *MPLS*.

#### DAFTAR PUSTAKA

- [1] Tjiptadi R. Sistem Jaringan Berbasis Vpn Dengan Metode Remote-Access. J Ilm Fifo. 2014;Vi(2):119–24.
- [2] Fajri M. Simulasi Antrian Paket Data Jaringan Dengan Mekanisme Drop Tail. J Ilm Fifo. 2016;Viii(2):151–60.
- [3] Saputra Fh. Survei Mekanisme Congestion Kontrol Pada Transmission Control Protocol Di Software Defined Network. Juti J Ilm Teknol Inf. :1–9.
- [4] Harfad At, Akbar Sr, Bhawiyuga A. Analisa Kinerja Algoritma Tcp Congestion Control Cubic , Reno , Vegas Dan Westwood+. J Pengemb Teknol Dan Ilmu Komput. 2018;2(3):1099–108.
- [5] Soewito B, Gunawan Fe, Afdhal S, Antonyova A. Analysis Of Quality Network Using Mpls And Non Mpls. 2017 Int Semin Intell Technol Its Appl Strength Link Between Univ Res Ind To Support Asean Energy Sect Isitia 2017 - Proceeding. 2017;2017–Janua:1–4.
- [6] Hundley K. Alcatel-Lucent Scalable Ip Network Nrs1. Wiley Publishing, Inc.; 2009. 430-431 P.
- [7] Othman Az, Rahman Ra, Md Zan Mm, Yusof Mi. The Effect Of Qos Implementation In Mpls Network. Ieee Symp Wirel Technol Appl Iswta. 2012;321–6.
- [8] Han L, Wang J, Wang C. A Variable Forwarding Equivalence Class For Mpls Networks. 2009 Int Conf Multimed Inf Netw Secur. 2009;0–3.
- [9] Sitohang S. Implementasi Jaringan Fiber To The Home ( Ftth ) Dengan Teknologi Gigabit Passive Optical Network ( Gpon ). J Simetris. 2018;9(2):879–88.
- [10] Susandi H, Pinem M. Analisis Kualitas Layanan Data Pada Jaringan Telekomunikasi Berbasis Cdma Evdo Rev . A. Singuda Ensikom. 2014;6(2):93–8.
- [11] Wedda App, Mayasari R, Elektro Ft, Telkom U, Play Lt. Implementasi Dan Analisis Soft Qos ( Diffserv ) Pada Jaringan Mpls – Te Untuk Layanan Triple Play Implementation And Analysis Of Soft Qos ( Diffserv ) On Mpls-Te. E-Proceeding Eng. 2015;2(2):2916–24.
- [12] Tiphon. General Aspects Of Quality Of Service (Qos). Vol. 1. 1999.
- [13] Nurhaida I, Ngadiyono. Quality Of Service For Traffic Monitoring System Based On Static Routing Using Eoip Tunnel Over Isec. Proc 2019 Asia Pacific Inf Technol Conf. 2019;(1):1–9.
- [14] Alcatel-Lucent. Qos Policies. 2015.
- [15] Hundley K. Alcatel-Lucent Scalable Ip Network Self-Study. Wiley Publishing, Inc.; 2009.
- [16] Kusniyati H, Yusuf R, Wiraka Bc. Analisis Kinerja Routing Protokol Ripng Dengan Ospfv3 Pada Jaringan Ipv6 Tunneling. J Pengkaj Dan Penerapan Tek Inform. 2017;10(2):56–63.
- [17] Wilkins. S. Cisco's Ppdioo Network Cycle. 2011.
- [18] Wulandari R. Analisis Qos ( Quality Of Service ) Pada Jaringan Internet ( Studi Kasus : Upt Loka

- Uji Teknik Penambangan Jampang Kulon – Lipi ). *J Tek Inform Dan Sist Inf.* 2016;2:162–72.
- [19] Dian L, Saputra D, Sulisty W. Analisis Qos Differentiated Service Pada Jaringan Mpls. *J Teknol Inf Dan Ilmu Komput.* 2017;4(4):227–36.
- [20] Naraswari N, Imansyah F, W Ftp. Analisis Uji Kuat Sinyal Terhadap Jarak Jangkauan Maksimal Sistem Penerimaan Sinyal Internet Berbasis Edimax Hp-5101ack. *J Fak Tek Univ Tanjungpura.* 2017;
- [21] Barry Ma, Tamgno Jk, Lishou C, Cissé Mb. Qos Impact On Multimedia Traffic Load ( Iptv , Roip , Voip ) In Best Effort Mode. *Int Conf Adv Commun Technol.* 2018;694–700.
- [22] Kandar S, Bhunia, C T. A New Protocol For Minimizing Jitter For Guaranteed Qos In Network Multimedia Communication. *2010 Int Conf Adv Comput Eng.* 2010;(1):15–8.
- [23] Nasir J, Andrianto E. Implementasi Quality Of Service , Limit Bandwidth Dan Load Balancing Dengan Menggunakan Firmware Dd-Wrt Pada Router Buffalo Whr-Hp-G300n. *J Simetris.* 2018;9(1):403–12.