



## **IMPLEMENTASI DAN ANALISIS MPLS-TE PADA JARINGAN BERBASIS MIKROTIK**



UNIVERSITAS  
**Dinamika**

**Oleh :**

**Rachmad Riadi Hari Purnomo**

**10.41020.0079**

---

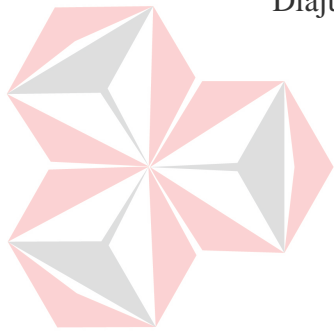
---

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA  
2015**

**IMPLEMENTASI DAN ANALISIS MPLS-TE PADA JARINGAN  
BERBASIS MIKROTIK**

**TUGAS AKHIR**

Diajukan sebagai salah satu syarat untuk menyelesaikan



Program Sarjana Komputer

UNIVERSITAS  
**Dinamika**

Oleh:

Nama : Rachmad Riadi Hari Purnomo

Nim : 10.41020.0079

Jurusan : Sistem Komputer

Program : S1 (Strata Satu)

**FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA  
2015**



UNIVERSITAS  
**Dinamika**

*“The key for a happiness is when you thankful for the grace that God  
has given... ”*

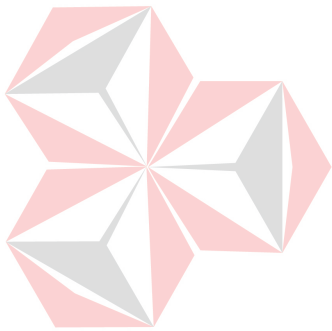
*Saya persembahkan kepada:*

*Allah SWT,*

*Keluargaku yang tercinta,*

*Orang-orang yang tersayang, serta*

*Teman – temanku yang terbaik.*



UNIVERSITAS  
**Dinamika**

**Tugas Akhir**  
**IMPLEMENTASI DAN ANALISIS MPLS-TE PADA JARINGAN**  
**BERBASIS MIKROTIK**

dipersiapkan dan disusun oleh  
**Rachmad Riadi Hari Purnomo**  
**NIM : 10.41020.0079**

Telah diperiksa, diuji dan disetujui oleh Dewan Penguji  
Pada :

**Susunan Dewan Penguji**

	Pembimbing	
	<b>I. Dr. Jusak</b>	_____
	<b>II. Johan Pamungkas, S.T</b>	_____
	Penguji	
	<b>I. Anjik Sukmaaji, S.Kom, M.Eng.</b>	_____
	<b>II. Yosefine Triwidyastuti, M.T</b>	_____

Tugas Akhir ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Sarjana

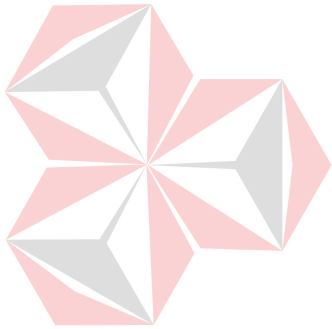
**Dr. Jusak.**  
**Dekan Fakultas Teknologi dan Informatika**

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

## PERNYATAAN

Dengan ini saya menyatakan dengan benar, bahwa Tugas Akhir ini adalah asli karya saya, bukan plagiat baik sebagian maupun apalagi keseluruhan. Karya atau pendapat orang lain yang ada dalam Tugas Akhir ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya.

Apabila dikemudian hari ditemukan adanya tindakan plagiat pada karya Tugas Akhir ini, maka saya bersedia untuk dilakukan pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.



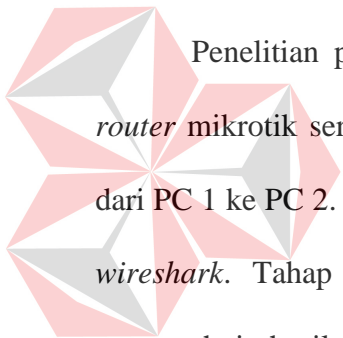
Surabaya, Februari 2015

Materai  
Rp. 6000,-

**Rachmad Riadi Hari Purnomo**  
NIM : 10.41020.0079

## ABSTRAK

Saat ini, permintaan layanan internet meningkat. Dengan demikian, untuk mengatasi efek kenaikan lalu lintas pada internet digunakan teknologi *Multiprotocol Label Switching* (MPLS). Dalam perkembangan teknologi ini MPLS-Traffic Engineering diperkenalkan untuk melengkapi teknologi MPLS sebelumnya. Traffic Engineering dalam MPLS yaitu proses pemilihan saluran data trafik untuk menyeimbangkan beban trafik di berbagai jalur dan titik *network* serta manajemen trafik pada jaringan yang lebih baik dengan menggunakan jalur alternative.



Penelitian pada MPLS-TE tahap pertama adalah pemberian alamat pada 4 *router* mikrotik serta konfigurasi MPLS-TE. Tahap kedua adalah pengambilan data dari PC 1 ke PC 2. Sedangkan tahap ketiga adalah pengambilan data dengan *software wireshark*. Tahap yang terakhir adalah pengujian dan perhitungan data untuk mengetahui hasil kualitas data yang didapat untuk menjadi acuan menarik kesimpulan.

Berdasarkan hasil penelitian implementasi dan analisis MPLS-TE pada jaringan berbasis mikrotik, kinerja MPLS tidak berbeda secara signifikan dengan MPLS-TE. Analisis yang dilakukan dengan menggunakan parameter QoS seperti *latency (delay)*, *jitter*, *throughput* dan *packet loss*.

**Kata kunci :** *MPLS-TE, Quality of Service, Mikrotik*

## KATA PENGANTAR

Puji syukur kehadiran Allah SWT, karena atas berkat rahmat dan ridho-Nya, sehingga dapat menyelesaikan tugas seminar sebagai bahan proposal Tugas Akhir dengan judul **“IMPLEMENTASI DAN ANALISIS MPLS-TE PADA JARINGAN BERBASIS MIKROTIK”**, tepat pada waktunya. Penulis menyadari bahwa Tugas Akhir ini tidak akan dapat terselesaikan dengan baik tanpa keterlibatan dan bantuan dari berbagai pihak. Pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Pimpinan STIKOM Surabaya yang telah banyak memberikan motivasi serta teladan yang dapat membantu penulis selama menempuh pembelajaran hingga saat ini.
2. Bapak Dr. Jusak dan Bapak Johan Pamungkas, S.T selaku dosen pembimbing yang telah meluangkan waktu untuk selalu memberikan bimbingan, masukan, dan koreksi selama proses penyusunan Tugas Akhir.
3. Yosefine Triwidyastuti, M.T dan Anjik Sukmaaji, S.Kom, M.Eng selaku dosen penguji Tugas Akhir yang telah memberikan masukan dan kritikan yang membangun.
4. Bapak Harianto, S.Kom., M.Eng. selaku dosen wali dan para dosen lainnya yang telah membantu penulis jika mengalami berbagai macam kesulitan sehingga penulis dapat termotivasi untuk terus berusaha hingga Tugas Akhir ini terlaksana sesuai dengan harapan.
5. Bapak dan Ibu tercinta yang telah memberikan dukungan dan doa dalam kelangsungan dan kelancaran penyelesaian Tugas Akhir.



6. Vinza Firqinia Fristia yang telah membantu segala hal dari awal penyusunan Tugas Akhir. Terima kasih sudah meluangkan banyak waktu. Terima kasih atas motivatsi dan semangatnya.
7. Teman-teman penulis, Adam Whiter Utha Bramantya, Imam Fauzi, Faysal Afdhor Rinzani, Fransiscus Daso dan Ahmad Dani P yang telah mendampingi dan memberi tempat saat penulis membutuhkan yang juga membantu dalam pelaksanaan Tugas Akhir ini.
8. Seluruh mahasiswa S1 Sistem Komputer angkatan 2010 yang ikut membantu dalam penyelesaian Tugas Akhir ini.
9. Serta pihak-pihak lain yang tidak bisa disebutkan satu persatu atas semua bantuan dalam penyusunan Tugas Akhir ini.

Banyak hal dalam laporan Tugas Akhir ini yang masih perlu diperbaiki lagi. Oleh karena itu penulis mengharapkan saran dan kritik yang dapat membangun dari semua pihak agar dapat menyempurnakan penulisan ini ke depannya. Penulis juga memohon maaf yang sebesar-besarnya jika terdapat kata-kata yang salah serta menyinggung perasaan pembaca. Akhir kata penulis ucapkan banyak-banyak terimakasih yang sebesar-besarnya kepada para pembaca, semoga tulisan ini dapat bermanfaat bagi para pembaca.

Surabaya, Februari 2015

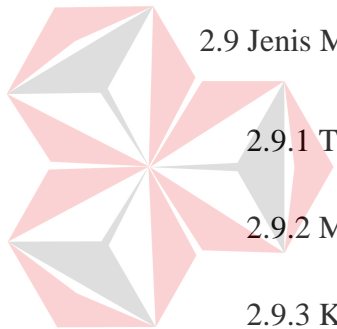
Penulis

## DAFTAR ISI

Halaman

ABSTRAK .....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI .....	x
DAFTAR GAMBAR .....	xv
DAFTAR TABEL .....	xvii
DAFTAR LAMPIRAN .....	xix
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Perumusan Masalah .....	3
1.3 Batasan Masalah .....	3
1.4 Tujuan .....	3
1.5 Sistematika Penulisan .....	4
BAB II LANDASAN TEORI .....	6
2.1 Quality of Service .....	6
2.2 Parameter-parameter Quality of Service .....	7
2.2.1 Jitter.....	7
2.2.2 Latency (Delay).....	7
2.2.3 Packet Loss .....	7
2.2.4 Throughput.....	7

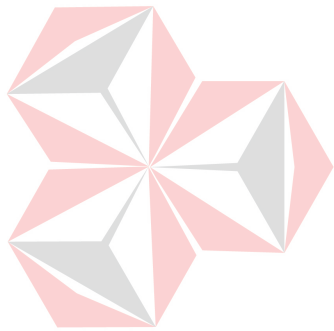
2.3 Definisi Multi Protocol Label Switching .....	8
2.4 Manfaat MPLS .....	9
2.5 Arsitektur MPLS .....	9
2.6 MPLS Label .....	11
2.7 Komponen MPLS .....	12
2.8 Struktur MPLS .....	14
2.8.1 <i>Edge Label Switching Routers (ELSR)</i> .....	14
2.8.2 <i>Label Distribution Protocol (LDP)</i> .....	14
2.9 Jenis MPLS .....	15
2.9.1 Traffic Engineering .....	15
2.9.2 Manfaat MPLS pada Traffic Engineering.....	17
2.9.3 Keterbatasan MPLS-TE .....	18
2.10 Routing .....	19
2.10.1 OSPF .....	19
2.10.2 Karakteristik OSPF .....	20
2.11 Definisi Internet Protocol Version 4 .....	21
2.11.1 Representasi Alamat.....	21
2.11.2 Jenis Alamat .....	22
2.11.3 Kelas IPv4 .....	22
2.12 TCP/IP (Transmission Control Protocol) .....	23



2.13 Lapisan Network .....	21
2.14 Lapisan Transport .....	22
2.15 Prinsip Kerja TCP/IP .....	25
2.16 Mikrotik .....	27
2.16.1 Mikrotik RB750 .....	28
2.16.2 Fitur Mikrotik.....	28
2.17 Layanan .....	31
2.17.1 FTP Server .....	32
2.17.2 FTP Client .....	32
2.18 Network Monitoring .....	32
2.18.1 Wireshark .....	32
2.18.2 Tujuan dan Manfaat Wireshark.....	33
<b>BAB III METODE PENELITIAN .....</b>	<b>34</b>
3.1 Metode Penelitian .....	34
3.1.1 Studi Kepustakaan.....	34
3.1.2 Percobaan .....	35
3.1.3 Analisis.....	35
3.2 Prosedur Penelitian .....	35
3.2.1 Pengumpulan Data dan Parameter Penelitian .....	36
3.2.2 Desain dan Topologi .....	37

3.2.3 Proses Monitoring dan Pengambilan Data .....	39
3.2.4 Konfigurasi Sistem.....	40
3.2.5 Uji Koneksi .....	41
3.2.6 Pengolahan Data.....	44
3.2.7 Pengujian Sistem dan Plotting .....	45
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>48</b>
4.1 Kebutuhan Sistem .....	48
4.2 Hasil Penelitian .....	49
4.2.1 Analisis Latency bandwidth 128 Kbps .....	49
4.2.2 Analisis Jitter bandwidth 128 Kbps .....	52
4.2.3 Analisis Throughput bandwidth 128 Kbps .....	55
4.2.4 Analisis Packet Loss bandwidth 128 Kbps .....	58
4.2.5 Analisis Latency bandwidth 512 Kbps .....	59
4.2.6 Analisis Jitter bandwidth 512 Kbps .....	62
4.2.7 Analisis Throughput bandwidth 512 Kbps .....	65
4.2.8 Analisis Packet Loss bandwidth 512 Kbps .....	68
4.2.9 Analisis Latency bandwidth 1 Mbps.....	69
4.2.10 Analisis Jitter bandwidth 1 Mbps.....	72
4.2.11 Analisis Throughput bandwidth 1 Mbps.....	75
4.2.12 Analisis Packet Loss bandwidth 1 Mbps .....	78
4.2.13 Hasil analisis dari Qos dengan bandwidth yang berbeda.....	79

BAB V PENUTUP .....	86
5.1 Kesimpulan .....	86
5.2 Saran .....	87
DAFTAR PUSTAKA .....	88



UNIVERSITAS  
**Dinamika**

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Perancangan Arsitektur MPLS.....	10
Gambar 2.2 Header MPLS.....	11
Gambar 2.3 Struktur komponen MPLS .....	12
Gambar 2.4 Mikrotik RB750 .....	28
Gambar 2.5 Wireshark .....	32
Gambar 3.1 Topologi MPLS-TE .....	38
Gambar 3.2 Flowchart proses pengambilan data .....	39
Gambar 3.3 Ping Client ke Router 2 .....	41
Gambar 3.4 Ping Client ke Router 3 .....	41
Gambar 3.5 Ping Client ke Router 4.....	42
Gambar 3.6 Ping Client ke Router 3 .....	42
Gambar 3.7 Ping Client ke Server .....	43
Gambar 4.1 Hasil grafik perbandingan delay audio .....	50
Gambar 4.2 Hasil grafik perbandingan delay video .....	51
Gambar 4.3 Hasil grafik perbandingan jitter audio.....	53
Gambar 4.4 Hasil grafik perbandingan jitter video.....	54
Gambar 4.5 Hasil grafik perbandingan throughput audio.....	56
Gambar 4.6 Hasil grafik perbandingan throughput video.....	57
Gambar 4.7 Hasil grafik perbandingan delay audio .....	60
Gambar 4.8 Hasil grafik perbandingan delay video .....	61
Gambar 4.9 Hasil grafik perbandingan jitter audio.....	63
Gambar 4.10 Hasil grafik perbandingan jitter video.....	64

Gambar 4.11 Hasil grafik perbandingan throughput audio.....	66
Gambar 4.12 Hasil grafik perbandingan throughput video.....	67
Gambar 4.13 Hasil grafik perbandingan delay audio .....	70
Gambar 4.14 Hasil grafik perbandingan delay video .....	71
Gambar 4.15 Hasil grafik perbandingan jitter audio.....	73
Gambar 4.16 Hasil grafik perbandingan jitter video.....	74
Gambar 4.17 Hasil grafik perbandingan Throughput audio .....	76
Gambar 4.18 Hasil grafik perbandingan Throughput video .....	77
Gambar 4.19 Grafik Hasil dari keseluruhan bandwidth delay pada audio .....	79
Gambar 4.20 Grafik Hasil dari keseluruhan bandwidth delay pada video .....	80
Gambar 4.21 Grafik Hasil dari keseluruhan bandwidth jitter pada audio .....	81
Gambar 4.22 Grafik Hasil dari keseluruhan bandwidth jitter pada video .....	82
Gambar 4.23 Grafik Hasil dari keseluruhan bandwidth Throughput pada audio ..	83
Gambar 4.24 Grafik Hasil dari keseluruhan bandwidth Throughput pada video ..	84



## DAFTAR TABEL

	Halaman
Tabel 3.1 Prosedur penelitian .....	36
Tabel 3.2 IP Address pada <i>Router</i> .....	38
Tabel 3.3 Data audio .....	43
Tabel 3.4 Data video .....	44
Tabel 4.1 Kebutuhan <i>Hardware</i> .....	48
Tabel 4.2 Kebutuhan <i>Software</i> .....	48
Tabel 4.3 Hasil perbandingan delay audio .....	49
Tabel 4.4 Hasil perbandingan delay video .....	51
Tabel 4.5 Hasil perbandingan jitter audio .....	52
Tabel 4.6 Hasil perbandingan jitter video .....	54
Tabel 4.7 Hasil perbandingan throughput audio .....	56
Tabel 4.8 Hasil perbandingan throughput video .....	57
Tabel 4.9 Hasil perbandingan packet loss audio .....	58
Tabel 4.10 Hasil perbandingan packet loss video .....	58
Tabel 4.11 Hasil perbandingan delay audio .....	59
Tabel 4.12 Hasil perbandingan delay video .....	61
Tabel 4.13 Hasil perbandingan jitter audio .....	62
Tabel 4.14 Hasil perbandingan jitter video .....	64
Tabel 4.15 Hasil perbandingan throughput audio .....	66

Tabel 4.16 Hasil perbandingan throughput video .....	67
Tabel 4.17 Hasil perbandingan packet loss audio .....	68
Tabel 4.18 Hasil perbandingan packet loss video .....	68
Tabel 4.19 Hasil perbandingan delay audio .....	69
Tabel 4.20 Hasil perbandingan delay video .....	71
Tabel 4.21 Hasil perbandingan jitter audio .....	72
Tabel 4.22 Hasil perbandingan jitter video .....	74
Tabel 4.23 Hasil perbandingan throughput audio .....	76
Tabel 4.24 Hasil perbandingan throughput video .....	77
Tabel 4.25 Hasil perbandingan packet loss audio .....	78
Tabel 4.26 Hasil perbandingan packet loss video .....	78
Tabel 4.27 Hasil dari keseluruhan bandwidth delay pada audio .....	79
Tabel 4.28 Hasil dari keseluruhan bandwidth delay pada video .....	80
Tabel 4.29 Hasil dari keseluruhan bandwidth jitter pada audio .....	81
Tabel 4.30 Hasil dari keseluruhan bandwidth jitter pada video .....	82
Tabel 4.31 Hasil dari keseluruhan bandwidth Throughput pada audio .....	83
Tabel 4.32 Hasil dari keseluruhan bandwidth Throughput pada video .....	84
Tabel 4.33 Hasil dari keseluruhan bandwidth Packet loss pada audio .....	85
Tabel 4.34 Hasil dari keseluruhan bandwidth Packet loss pada video .....	85

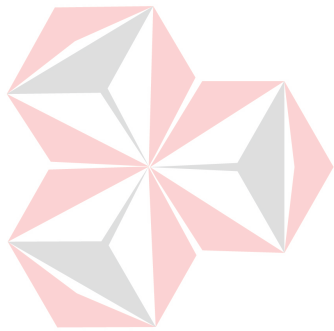
## **DAFTAR LAMPIRAN**

Lampiran 1 Data Penulis

Lampiran 2 Konfigurasi IP Address pada masing – masing router

Lampiran 3 Konfigurasi MPLS

Lampiran 4 Konfigurasi MPLS-Traffic Engineering



UNIVERSITAS  
**Dinamika**

# BAB 1

## PENDAHULUAN

### 1.1. Latar Belakang Masalah

Semakin tinggi pemakaian layanan internet akan semakin tinggi trafik internet yang digunakan, sehingga kecepatan pengiriman data menjadi masalah yang sering dialami dalam jaringan komputer. Untuk itu perlu proses yang cepat untuk mengatasi pengiriman dan pengambilan data untuk mengurangi waktu atau efisiensi waktu yang digunakan oleh *user* agar tidak perlu membuang waktu. Ada beberapa protocol jaringan yang dapat digunakan, salah satunya protokol *Multiprotocol Label Switching* (MPLS). *Multiprotocol Label Switching* (MPLS) adalah sebuah teknik yang menggabungkan kemampuan manajemen *switching* yang ada pada teknologi ATM atau disebut juga *Asynchronous Transfer Mode* dengan fleksibilitas *network layer* yang dimiliki pada teknologi IP. MPLS menggabungkan teknologi *switching layer 2* dan teknologi *routing layer 3* sehingga menjadi solusi jaringan terbaik dalam masalah kecepatan, *scalability*, *Quality of Service* (QoS) dan *traffic engineering*. Ada berbagai jenis MPLS tetapi penulis menggunakan salah satu jenis tersebut, yaitu *traffic engineering*.

*Traffic Engineering* dalam MPLS yaitu suatu proses pemilihan saluran data trafik untuk menyeimbangkan beban trafik pada berbagai jalur dan titik dalam network (Rijayana, 2005). Tujuan dari *traffic engineering* ini yaitu untuk memungkinkan operasional network yang handal dan efisien,

sekaligus mengoptimalkan penggunaan sumberdaya dan performansi trafik. (Wastuwibowo, 2003)

Beberapa *paper* telah membahas implementasi MPLS-TE, misal seperti *paper* berjudul “Analisis dan Implementasi Jaringan Backbone MPLS-TE pada Layanan VoIP”, mengatakan bahwa unjuk kerja dari parameter QoS untuk delay, throughput dan packet loss pada MPLS-TE selalu lebih unggul dikarenakan ada mekanisme *fast reroute* (Simatupang, 2011). *Paper* lain berjudul “Analisis pemodelan sistem jaringan MPLS-TE pada layanan multimedia berbasis IP Multimedia Subsystem (IMS)”, mengatakan bahwa performansi IMS server yang menggunakan teknologi MPLS-TE lebih baik jika dibandingkan dengan yang tanpa menggunakan teknologi MPLS-TE. (Aloriadi, 2011)

Pada tugas akhir ini akan dilakukan penelitian unjuk kerja jaringan menggunakan teknologi MPLS-TE. Hal ini dilatar belakangi oleh kurangnya implementasi teknologi MPLS pada perusahaan – perusahaan. Padahal MPLS-TE dapat digunakan untuk manajemen trafik pada jaringan yang lebih baik dengan menggunakan jalur alternative.

Berdasarkan uraian diatas dalam tugas akhir ini akan dikaji tentang bagaimana Implementasi dan Analisis MPLS-TE pada jaringan berbasis mikrotik. Tujuan dari penulis ini dapat menganalisis seberapa bagus kinerja pada protokol MPLS-TE ini yang digunakan untuk pengiriman data pada protokol tersebut.

## 1.2. Perumusan Masalah

Berdasarkan latar belakang diatas maka dapat dirumuskan permasalahan :

1. Bagaimana mengimplementasikan MPLS-TE berbasis IPv4 pada jaringan dengan menggunakan perangkat mikrotik ?
2. Bagaimana melakukan pengujian dan analisis kualitas layanan pada protokol MPLS-TE pada jaringan IPv4 dengan menggunakan parameter uji latency (delay), jitter, packet loss, dan throughput ?

## 1.3. Batasan Masalah

Untuk menghindari pembahasan yang lebih luas terkait dengan implementasi dan analisis MPLS-TE pada mikrotik. Terdapat beberapa batasan masalah, maka penelitian ini hanya ditentukan pada rung lingkup tertentu antara lain :

1. Penelitian ini menggunakan 4 router mikrotik RB750.
2. Media penghubung antar router dan pc/laptop menggunakan kabel UTP
3. Pengujian dilakukan dengan menggunakan software wireshark.

## 1.4. Tujuan

Berdasarkan rumusan masalah yang diuraikan diatas, maka tujuan penelitian ini adalah sebagai berikut :

1. Mengimplementasikan MPLS-TE berbasis IPv4 pada jaringan dengan menggunakan perangkat mikrotik.
2. Melakukan pengujian dan analisis kualitas layanan pada protokol MPLS-TE pada jaringan IPv4 dengan menggunakan parameter uji latency (delay), jitter, packet loss, dan throughput.

### 1.5. Sistematika Penulisan

Untuk memudahkan didalam memahami persoalan dan pembahasannya, maka penulisan laporan Tugas Akhir ini dibuat dengan sistematika sebagai berikut:

#### **BAB I        PENDAHULUAN**

Pada bab ini dikemukakan hal-hal yang menjadi latar belakang, perumusan masalah, batasan masalah, tujuan yang ingin dicapai serta sistematika penulisan laporan tugas akhir ini.

#### **BAB II        LANDASAN TEORI**

Pada bab ini membahas mengenai teori - teori dari penelitian yang dilakukan, antara lain *Quality of Service, Multi Protocol Label Switching, MPLS-Traffic Engineering, Internet Protocol version 4, Routing OSPF, Mikrotik, dan Monitoring system*

#### **BAB III        METODE PENELITIAN**

Bab ini menjelaskan mengenai perancangan dan pembuatan topologi jaringan yang digunakan, kebutuhan system, kebutuhan data, pengalamatan IP, dan parameter uji QoS.

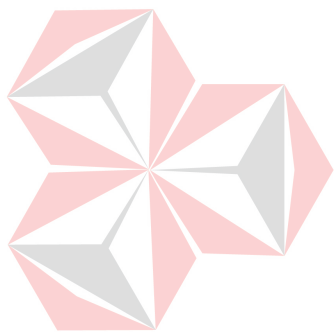
#### **BAB IV        HASIL DAN PEMBAHASAN**

Pada bab ini membahas penelitian yang dilakukan, antara lain : membahas mengenai uji koneksi antara client dan server, membahas hasil dari penelitian, membahas tabel yang digunakan pada penelitian, membahas analisis kinerja pada teknologi MPLS-TE dan MPLS dengan parameter uji QoS, antara lain *latency*

*(delay), jitter, throughput dan packet loss.*

## **BAB V      PENUTUP**

Pada bab ini akan membahas tentang kesimpulan dari hasil yang didapat pada saat perhitungan data dan analisis data.



UNIVERSITAS  
**Dinamika**



## BAB II

### LANDASAN TEORI

#### 2.1. Quality of Service

*Quality of Service* (QoS) dapat didefinisikan sebagai kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan penyediaan bandwidth dalam jumlah yang tepat serta mengatasi *delay* dan *jitter*. Definisi QoS dari segi *networking* mengacu pada kemampuan QoS untuk memberikan layanan kepada *traffic* jaringan dengan kelas yang berbeda, sesuai dengan tujuan akhir dari QoS yakni memberikan *network service* yang lebih baik dan terencana dengan *dedicated bandwidth*, *jitter*, dan *latency* yang terkontrol dan meningkatkan karakteristik *loss*. QoS dibuat dengan tujuan untuk memberikan jaminan kepada user dalam mendapatkan performansi terbaik dari jaringan. Penyediaan performansi terbaik tersebut dapat dibuktikan oleh QoS melalui pemenuhan layanan dengan kebutuhan yang berbeda, dengan berbagai jenis aplikasi, namun dengan infrastruktur yang sama, baik secara kualitatif maupun kuantitatif. QoS dapat dilihat dari tingkat kecepatan dan keandalan dalam mengelola penyampaian data dalam suatu informasi dengan jenis beban yang beragam. Terdapat beberapa parameter yang digunakan untuk mengukur tingkat kecepatan dan keandalan suatu layanan internet, diantaranya *latency (delay)*, *jitter*, *packet loss*, *throughput*. (Auliya, 2013)

## 2.2. Parameter – parameter Quality of Service

Ada beberapa parameter QoS yang digunakan, antara lain :

### 2.2.1.Jitter

*Jitter* merupakan variasi *delay*, yang menyebabkan adanya perbedaan waktu kedatangan paket. *Jitter* disebabkan adanya perbedaan panjang antrian, waktu pengolahan data, dan waktu penghimpunan ulang paket-paket di akhir transmisi *jitter*. (Auliya, 2013)

### 2.2.2.Latency (Delay)

*Latency* atau *delay* merupakan waktu tempuh data dari asal ke tujuan. *Delay* dipengaruhi oleh jarak, media transmisi, dan lamanya proses transmisi. (Auliya, 2013)

### 2.2.3.Packet Loss

*Packet loss* berkaitan dengan adanya paket yang hilang saat ditransmisikan. *Packet loss* dapat terjadi karena adanya beberapa alasan berikut :

1. *Traffic overload*
2. *Congestion*, yakni tabrakan antar dua paket yang sedang ditransmisikan.
3. *Error* pada media fisik
4. *Overflow buffer* yang menyebabkan kegagalan pada sisi penerima

Paket yang hilang saat ditransmisikan akan ditransmisikan ulang dan hal tersebut akan mempengaruhi seluruh jaringan karena memakan waktu. (Auliya, 2013)

### 2.2.4.Throughput

*Throughput* merupakan *rate* transfer efektif yang diukur berdasarkan satuan *byte per second* (bps). *Throughput* merupakan jumlah total dari paket yang datang dan sampai di tujuan. Paket tersebut diamati ketika berada di destinasi selama interval tertentu dan dibagi berdasarkan interval waktu tertentu. (Auliya, 2013)

### 2.3. Definisi Multi Protocol Label Switching

MPLS adalah teknologi penyampaian paket pada jaringan *backbone* berkecepatan tinggi yang menggabungkan beberapa kelebihan dari sistem komunikasi *circuit-switched* dan *packet-switched* yang melahirkan teknologi yang lebih baik. Yang dimaksud *circuit-switched* dan *packet-switched* adalah sebagai berikut :

- *Circuit-switched* adalah model jaringan yang menerapkan sebuah jalur komunikasi yang *dedicated* antara 2 *station*.
- *Packet-switched* adalah metode komunikasi jaringan digital yang ditransmisikan semua data yang terlepas dari struktur paket.

MPLS Label dapat membangun pemetaan *label-to-label* antar *router*. Label ini melekat pada paket IP yang memungkinkan *router* untuk meneruskan jalur lalu lintas dengan melihat label dan bukan alamat IP tujuan. Paket yang diteruskan oleh *Label Switching* bukan *IP Switching*. Teknik *Label Switching* bukanlah hal yang baru. Teknologi yang sebelumnya yaitu Frame Relay dan ATM teknologi tersebut dapat digunakan untuk memindahkan *frame* seluruh jaringan. Pada Frame Relay, *framenya* bisa menjadi sedikit panjang. Sedangkan *Asynchronous Transfer Mode* (ATM), mempunyai *FixedLength* yang terdiri dari 5 *header byte* dan *payload 48 byte*. *Header* pada ATM dan Frame Relay dapat mengacu pada *virtual circuit* yang berada pada *frame*. Frame Relay dan ATM mempunyai kesamaan yaitu setiap *hop* diseluruh jaringan dan nilai label dalam *header* dapat berubah. Hal ini berbeda dari paket *forwarding*, ketika sebuah *router* *menforward* paket IP, nilai yang berkaitan dengan tujuan dari paket tidak merubah alamat IP tujuan. Fakta bahwa MPLS Label digunakan untuk meneruskan paket-

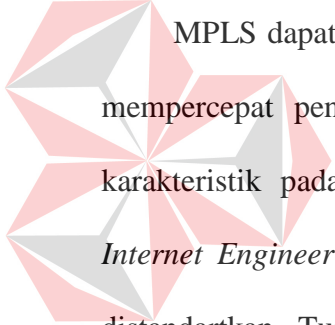
paket. (Ghein, 2006)

#### 2.4. Manfaat MPLS

Bagian ini menjelaskan secara singkat tentang manfaat dari MPLS dalam jaringan. Manfaat dari MPLS sebagai berikut :

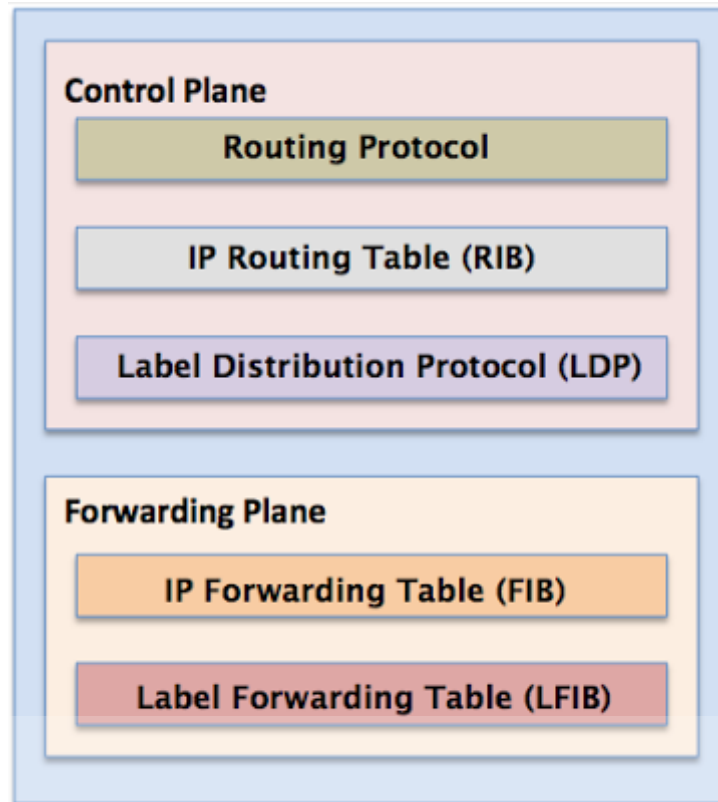
- Penggunaan satu infrastruktur jaringan
- IP yang lebih baik dari integrasi ATM
- *Peer-to-peer* pada model MPLS VPN
- Arus lalu lintas dapat optimal
- Rekayasa lalu lintas (Ghein, 2006)

#### 2.5. Arsitektur MPLS



MPLS dapat didefinisikan sebagai gabungan dari layer 2 dan layer 3 untuk mempercepat pengiriman paket. Arsitektur ini dirancang untuk memenuhi karakteristik pada sebuah jaringan *carrier* berskala besar. Pada tahun 1997 *Internet Engineering Task Force* (IETF) mengembangkan metode umum yang distandartkan. Tujuan dari MPLS ini untuk menstandartkan protokol-protokol yang menggunakan teknik pengiriman label *swapping*. Penggunaan label *swapping* memiliki banyak keuntungan, seperti memisahkan masalah *routing* dari masukan *forwarding*. MPLS memiliki kelebihan yang mampu memperkenalkan *connection stack* kembali ke dalam *dataflow* (Wastuwibowo, 2003).

*Control Plane* merupakan sebuah perancangan dari arsitektur MPLS dapat dilihat pada Gambar 2.1.



Sumber : <http://www.startnetworks.info>

Gambar 2.1 Perancangan arsitektur MPLS

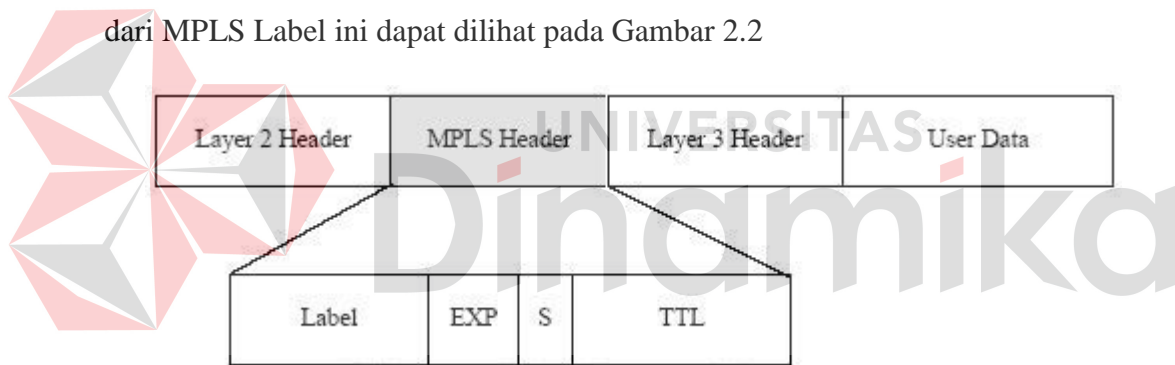
Pada *Control Plane* bertanggung jawab untuk bertukar informasi *routing* pada layer 3 dan label. *Control Plane* mempunyai mekanisme yang kompleks mengenai pertukaran informasi *routing* seperti OSPF, EIGRP, IS-IS, BGP dan pertukaran label seperti TDP (*Tag Distribution Protocol*), LDP (*Label Distribution Protocol*), BGP (*Border Gateway Protocol*), dan RSVP (*Resource Reservation Protocol*). Gambar 2.1 pada *Control Plane* dibagi menjadi 3 pertama *Routing Protocol* digunakan untuk bertanggung jawab pertukaran informasi *routing*, kemudian yang kedua *IP Routing Table* digunakan untuk membangun FIB (*Forwarding Information Base*) pada data plane dan yang ketiga LDP (*Label Distribution Protocol*) bertanggung jawab untuk pertukaran label. Setelah bertukar label dengan LDP, kemudian LFIB akan terbentuk dalam *Data Plane*

atau *Forwarding Plane*. (Kumar , 2010)

Pada *Data Plane* atau *Forwarding Plane* bertugas untuk meneruskan paket berdasarkan label dan header IP. Data Plane mempunyai kesederhanaan dalam mempertahankan LFIB (*Label Forwarding Information Base*) dan FIB (*Forwarding Information Base*). LFIB berisikan table next-hop label dari hasil proses MPLS, jadi LFIB ini mengecek label yang masuk dan diasosiasikan ke label yang dia punya. Kalau FIB berisikan table next-hop IP dan exit interface dari hasil proses *Routing Table*. (Rahman, 2013)

## 2.6. MPLS Label

Dalam satu MPLS Label mempunyai 32 bit dengan struktur tertentu. Sintak dari MPLS Label ini dapat dilihat pada Gambar 2.2



Sumber : <http://digilib.ittelkom.ac.id>

Gambar 2.2 Header MPLS

Label digunakan untuk *forwarding* dan *traffic engineering*. Label MPLS terdiri dari beberapa field, seperti berikut ini :

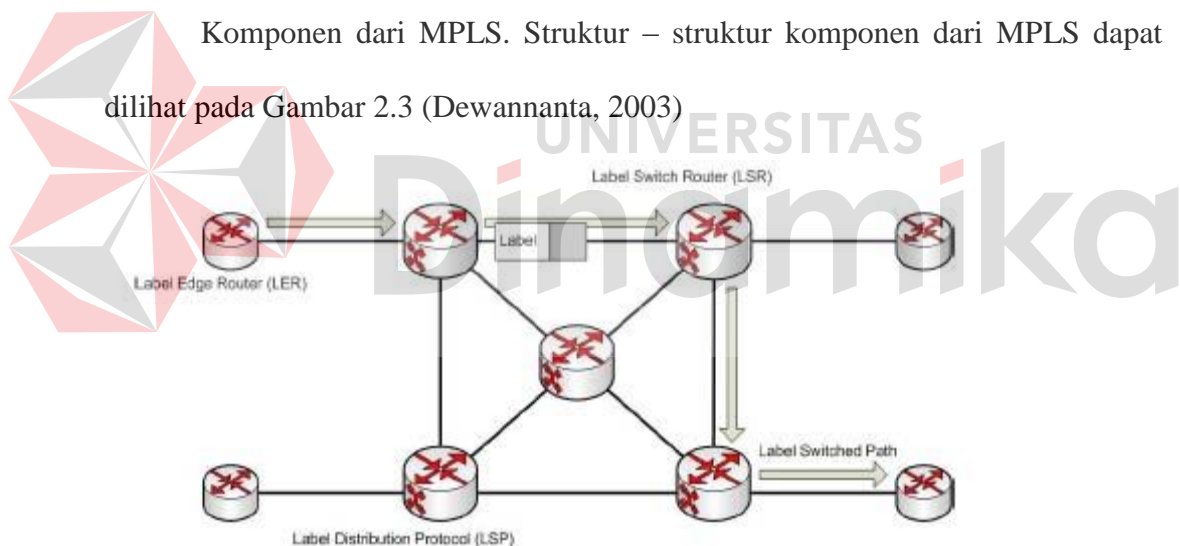
1. *Label*, merupakan bagian yang terdiri dari 20 bit informasi FEC
2. *CoS*, merupakan bagian yang terdiri dari 3bit informasi *Class of Service*.

Bagian ini digunakan untuk menentukan waktu penundaan pembuangan yang berada dalam *router*.

3. *Stack*, merupakan bagian yang terdiri dari sebuah bit informasi posisi label pada sebuah *multilabel hierarchy*. Bagian ini diperlukan ketika label diterapkan pada label tunggal.
4. *TTL*, merupakan bagian yang terdiri dari 8 bit informasi *Time to Live* sebuah paket. TTL berfungsi sama dengan *IP Header*. Hal ini dapat mengalami penurunan sebesar 1 *hop* dan fungsi utamanya untuk menghindari paket dalam *routing loop*. Dan jika *routing loop* terjadi dan tidak ada TTL, maka akan terjadi paket *loop* selamanya. Jika TTL mencapai 0, maka paket tersebut akan dibuang. (Ghein, 2006)

## 2.7. Komponen MPLS

Komponen dari MPLS. Struktur – struktur komponen dari MPLS dapat dilihat pada Gambar 2.3 (Dewannanta, 2003)



Sumber : <http://digilib.ittelkom.ac.id>

Gambar 2.3 Struktur komponen MPLS

Berikut adalah komponen dari MPLS.

### ✧ MPLS Node

*Router* yang ada di jaringan MPLS akan meneruskan paket yang diterima

berdasarkan label.

#### ✧ **MPLS Label**

Merupakan sebuah *header* tambahan yang diletakan diantara *layer 2* dan *IP header*.

#### ✧ **MPLS Ingress Node**

MPLS *node* mengatur lalu lintas saat paket masuk pada MPLS *core* dan *Ingress Node* disebut dengan PE (*Power Edge*) *router*.

#### ✧ **MPLS Egress Node**

MPLS *node* yang mengatur trafik saat paket meninggalkan MPLS *core*. *Egress node* juga disebut dengan PE (*Power Edge*) *router*.

#### ✧ **Label Edge Router (LER)**

MPLS *node* menghubungkan sebuah MPLS *domain* dengan *node* yang berbeda diluar *domain*.

#### ✧ **Label Switched Path (LSP)**

Merupakan jalur yang terbentuk satu atau lebih *Label Switching Hop* yang diteruskan oleh *label swapping* berdasarkan *label Forwarding Equivalent Class* dari satu *node* ke *node* yang lain.

#### ✧ **Label Switching Router (LSR)**

LSR merupakan perangkat pendukung LSP, yakni *router* yang dapat menjalankan MPLS ketika paket berada di domain MPLS. LSR melihat asal paket, IP header, dan data dari *application layer* untuk menentukan FEC mana yang sesuai dan akan diberikan. LSR menghubungkan titik-titik dan menentukan jalur mana yang akan dilewati paket. LSR pertama disebut *ingress* dan LSR terakhir disebut *egress*. (Dewannanta, 2003)



### ✧ ***Forwarding Equivalence Class (FEC)***

FEC merupakan kumpulan paket yang memiliki tipe yang sama dan menerima perlakuan *routing* yang sama. FEC digunakan untuk mendefinisikan kriteria-kriteria yang akan digunakan untuk mengevaluasi paket. Paket-paket yang berkaitan dengan FEC diasosiasikan dan mengalami perlakuan *routing* yang sama. FEC berbasis rute IP *address* yang sama atau berbasis kebutuhan layanan yang sama. Misalnya saja *low latency*. FEC diberikan ke paket oleh sebuah *Label Switch Router (LSR)*. (Dewannanta, 2003)

## **2.8. Struktur MPLS**

Struktur MPLS terdiri dari *Edge Label Switching Routers (ELSR)* yang mengelilingi sebuah *core Label Switching Routers (LSRs)*. Elemen dasar MPLS yaitu :

### **2.8.1. *Edge Label switching Routers (ELSR)***

ELSR terletak pada perbatasan jaringan MPLS dan berfungsi sebagai pengaplikasian label ke dalam paket yang masuk ke dalam jaringan MPLS.

Perangkat *Label Switches* ini berfungsi untuk merubah paket-paket yang telah dilabeli berdasarkan label tersebut dan mendukung layer 3 *routing* atau layer 2 *switching*. *Label switch* tersebut memiliki persamaan yang biasa dikerjakan dalam ATM. (Rijayana, 2005)

### **2.8.2. *Label Distribution Protocol (LDP)***

LDP merupakan protokol yang berperan dalam distribusi label dan digunakan untuk memindahkan informasi berupa definisi label ke seluruh bagian network. LDP membantu memindahkan informasi ke LSR karena LSR membutuhkan informasi dan asosiasi yang sama dari label ke FEC. (Rijayana,

2005)

## 2.9. Jenis MPLS

Ada beberapa jenis MPLS, salah satunya yang digunakan dalam tugas akhir sebagai berikut.

### 2.9.1. Traffic Engineering

*Traffic engineering* merupakan kemampuan yang dimiliki oleh MPLS untuk merencanakan *traffic* jaringan. Prosedur *traffic engineering* ini adalah dengan memberikan batasan dan kondisi tambahan untuk diperhitungkan oleh berbagai LSR ketika melakukan *routing* dan nantinya *route* tersebut akan diambil oleh sebuah paket melalui domain MPLS. Hal tersebut dilakukan dengan tujuan menyeimbangkan beban pada *traffic* pada berbagai jalur dan titik dalam network. Selain itu tujuan lain dari hal tersebut adalah memungkinkan operasional network yang andal dan efisien, sekaligus mengoptimalkan penggunaan *resource* dan *traffic performance*. (Auliya, 2013)

Dalam melakukan *traffic engineering*, MPLS menggunakan beberapa komponen, diantaranya :

#### 1. *Constraint Shortest Path First (CSPF)*

*CSPF* merupakan komponen yang dapat memodifikasi protokol OSPF sehingga memberikan izin untuk menambahkan kelas dan batasan lain untuk paket tertentu. (Auliya, 2013)

#### 2. *Reservations*

RSVP dan CR-LDP mampu menghemat *bandwidth* melalui sebuah LSP dengan tujuan memperluas penggunaan paket tertentu. (Auliya, 2013)

### 3. *Link state*

*Link State* bekerja dengan memperluas IGP untuk memungkinkan perubahan pada network yang dikomunikasikan ke seluruh jaringan kepada berbagai LSR. Terdapat beberapa komponen dalam *traffic engineering*, diantaranya :

#### a. *Manajemen Path*

Manajemen *path* merupakan suatu kegiatan untuk menentukan *route* dan *maintenance tunnel* LSP. Kegiatan tersebut dilakukan berdasarkan karakteristik tertentu secara eksplisit baik secara manual maupun otomatis. *Constraint-based routing* merupakan salah satu metode yang digunakan dalam manajemen *path* secara otomatis dengan melakukan pertimbangan terhadap beberapa opsi alternatif spesifikasi administratif. (Auliya, 2013)

#### b. *Penempatan Traffic*

Penempatan *traffic* dilakukan melalui LSP. Pengalokasian *traffic* ke dalam LSP dilakukan oleh manajemen *traffic* dan meliputi fungsi pemisahan dimana kelas-kelas dibagi-bagi dan dipetakan ke dalam LSP. Hal penting dalam penempatan *traffic* ini adalah pembagian beban dalam LSP. Hal tersebut dilakukan dengan penyusunan semacam pembobotan pada LSP dan dapat dilakukan baik secara implisit maupun eksplisit. (Auliya, 2013)

#### c. *Penyebaran Informasi Kondisi Network dengan Protokol Persinyalan*

Dalam pembentukan LSP, akan dibutuhkan protokol persinyalan untuk *routing* sehingga menghasilkan *route* yang paling tepat. Penggunaan protokol persinyalan akan memudahkan penentuan *path* secara otomatis dalam MPLS *traffic engineering*. Dua macam protokol persinyalan yang sering digunakan adalah CR-LDP dan RSVP-TE. (Auliya, 2013)

- RSVP-TE merupakan pengembangan dari protokol QoS yang menentukan penentuan *route* dan transfer dari label. RSVP-TE merupakan perluasan dari RSVP dan memungkinkan distribusi label sehingga dapat mendukung persinyalan QoS dan *routing* secara eksplisit. RSVP-TE bekerja langsung pada IP.
- CR-LDP merupakan sebuah pengembangan dari LDP asli yang memungkinkan penentuan *route* dan menambahkan QoS. CR-LDP bekerja pada TCP-UDP.

### 2.9.2. Manfaat MPLS pada Traffic Engineering

Rekayasa Lalu Lintas di MPLS melibatkan teknik mengarahkan lalu lintas yang mengalir dalam jaringan. Beberapa prosedur routing yang menerapkan paket forwarding untuk transmisi yang aman. Keuntungan meningkatkan rekayasa lalu lintas, sebagai berikut :

- ***Minimize network congestion*** : Jaringan MPLS dapat menerapkan TE untuk mengurangi penyumbatan jaringan dan menaikkan kinerja. Semua teknik routing yang digunakan dimodifikasi untuk memetakan data paket ke sumber jaringan. Seperti proses pemetaan yang dapat menangani kemacetan paket dengan faktor *latency*, *jitter*, dan *packet loss*. MPLS TE memungkinkan eksploitasi bandwidth yang digunakan daripada mengalokasikan bandwidth yang baru untuk mengoperasikan jalur rekayasa lalu lintasnya. (Ghein, 2006)
- ***Deployment flexibility*** : TE mempunyai mekanisme ketika trafik meningkat dan mengalami kemacetan, maka jalur yang dilaluinya akan tetap stabil. (Ghein, 2006)

- ***Class of Service (CoS)*** : CoS memiliki nilai 3bit yang digunakan untuk memberikan sebuah antrian pada saat mentransmisikan ke jalur lalu lintas. Pada pintu masuk, paket IP ditandai dengan nilai CoS dan dikodekan ke dalam header MPLS. Digunakan untuk menghindari kemacetan paket yang ditransmisikan pada jaringan. CoS memiliki fungsi, antara lain *Committed Access Rate (CAR)*, *Weighted Random Early Detection (WRED)*, and *Weighted Fair Queuing (WFQ)*. Setiap CoS menerapkan rekayasa lalu lintas dengan mengelompokkan lalu lintas berdasarkan bandwidth yang tersedia pada link, untuk mengelola paket pada tepi router, drop probabilitas dan mengontrol lalu lintas jaringan menggunakan algoritma (seperti *round-robin*).

(Ghein, 2006)

### 2.9.3. Keterbatasan MPLS-TE

MPLS mempunyai keterbatasan pada kegunaanya, dapat dilihat dibawah ini keterbatasan yang dimiliki oleh MPLS-TE :

- ***Over-utilization of secondary links*** : Seperti pada contoh *link failur*, Dengan cepat mengubah rute MPLS TE menggunakan jalur alternative atau jalur cadangan untuk mengubah rute lalu lintas melalui link sekunder. Meskipun mempunyai metode cadangan, sering terjadi kegagalan pada node jaringan yang menjadi kendala kemacetan lalu lintas pada jalur alternatif dan tidak dapat mengurangi kendala tersebut secara keseluruhan. (Ghein, 2006)
- ***Manual path setup*** : Untuk menerapkan rekayasa lalu lintas, memerlukan konfigurasi secara manual terlepas dari Internet Protocol untuk paket routing. Perhitungan jalur fisik menunjukkan hop berturut - turut terjadi dari sumber ke tujuan. Namun, pengaturan manual ini membutuhkan solusi

provider yang profesional untuk mengkonfigurasi secara manual untuk digunakan dalam penerapan. Selain itu, jika node perantara tidak dikonfigurasi secara manual, maka lalu lintas pada MPLS-TE tidak berjalan. (Ghein, 2006)

- ***Protocol dependency for automatic rerouting*** : Jika jaringan menggunakan protokol routing OSPF, maka perhitungan pathnya secara otomatis dan sistematis pada IP traffic dapat dilakukan pada jalan MPLS-TE. Hal ini berlaku untuk jaringan yang menggunakan protokol IS-IS. Keberadaan LDP pada MPLS-VPN sangat penting untuk proses tunneling. Ditambahkan dengan Area Border Routers (ABRs) memerlukan identifikasi pada jalan TE dan optimalisasi yang layak. Berdasarkan pengaturan konfigurasi di IS-IS protokol, TE tunneling diatur statis atau dinamis. (Ghein, 2006)
- ***Performance variation in MPLS Fast reroute*** : Mekanisme QoS yaitu mempertahankan bandwidth untuk tunnel yang beroperasi sebagai jalur cadangan. Node intermediate sebagai konfigurasi manual jalur TE, fast reroute mengubah rute alternatif bila terjadi link failur. Selain itu, node intermediate dikonfigurasi dalam jaringan MPLS secara manual.
- ***Lack of systematic mapping system*** : Pemetaan IP traffic yang dinamis pada MPLS TE mengenal router yang mengatur jalan. (Ghein, 2006)

## 2.10. Routing

Beberapa jenis *routing* yang sudah diterapkan dan digunakan sebagai berikut :

### 2.10.1. OSPF

*Open Shortest Path First* (OSPF) merupakan protokol *routing link state* dan digunakan untuk menghubungkan *router-router* yang berada dalam satu

*Autonomous System* (AS) sehingga protokol *routing* ini termasuk juga kategori *Interior Gateway Protocol* (IGP). OSPF pertama kali dikembangkan pada tahun 1987 oleh *Internet Engineering Task Force* (IETF) dan pertama kali dipublikasikan adalah OSPFv1. Pada tahun 1991, OSPFv2 mulai dipublikasikan sampai tahun 1998 perkembangan OSPF menjadi OSPFv3 hingga tahun 2008 OSPFv3 ini disempurnakan. (Towidjojo, 2012)

### 2.10.2. Karakteristik OSPF

Protokol *routing* OSPF memiliki beberapa karakteristik sebagai berikut :

- Merupakan protokol *routing link state*, sehingga setiap *router* memiliki gambaran topologi jaringan.
- Menggunakan *Hello Packet* untuk mengetahui keberadaan *neighbor router*.
- *Routing update* hanya dikirimkan bila terjadi perubahan dalam jaringan dan dikirimkan secara *multicast*.
- Dapat bekerja dengan konsep hirarki karena dapat dibagi berdasarkan konsep *area*.
- Menggunakan *cost* sebagai *metric*, dengan *cost* terendah yang akan menjadi *metric* terbaik.
- Tidak memiliki keterbatasan *hop count*
- Merupakan *routing* protokol *classless*
- Nilai secara *default Administrative Distance* (AD)
- Memiliki fitur *authentication* saat mengirim *routing update*. (Towidjojo, 2012)

## 2.11. Definisi Internet Protocol version 4

IPv4 adalah sebuah jenis pengalamatan jaringan yang digunakan di dalam protokol jaringan TCP/IP yang menggunakan protokol IP versi 4. IP versi ini memiliki keterbatasan yakni hanya mampu mengalami sebanyak 4 miliar host komputer di seluruh dunia. (Setiawan, 2014)

### 2.11.1. Representasi alamat

Alamat IP versi 4 umumnya diekspresikan dalam notasi desimal bertitik (*dotted-decimal notation*), yang dibagi ke dalam empat buah oktet berukuran 8-bit. Dalam beberapa buku referensi, format bentuknya adalah w.x.y.z. Karena setiap oktet berukuran 8-bit, maka nilainya berkisar antara 0 hingga 255 (meskipun begitu, terdapat beberapa pengecualian nilai). Alamat IP yang dimiliki oleh sebuah *host* dapat dibagi dengan menggunakan subnet mask jaringan ke dalam dua buah bagian, yakni : (Setiawan, 2014)

- *Host Identifier/HostID* atau *Host address* (alamat host) yang digunakan khusus untuk mengidentifikasi alamat host (dapat berupa workstation, server atau sistem lainnya yang berbasis teknologi TCP/IP) di dalam jaringan. Nilai *host identifier* tidak boleh bernilai 0 atau 255 dan harus bersifat unik di dalam *network identifier/segmen* jaringan di mana ia berada.
- *Network Identifier/NetID* atau *Network Address* (alamat jaringan) yang digunakan khusus untuk mengidentifikasi alamat jaringan di mana host berada. Semua sistem di dalam sebuah jaringan fisik yang sama harus memiliki alamat *network identifier* yang sama. *Network identifier* juga harus bersifat unik dalam sebuah *Internetwork*. (Setiawan, 2014)



### 2.11.2. Jenis Alamat

Alamat IPv4 terbagi menjadi beberapa jenis, yakni sebagai berikut:

- **Alamat Unicast**, merupakan alamat IPv4 yang ditentukan untuk sebuah antarmuka jaringan yang dihubungkan ke sebuah *Internetwork* IP. Alamat *unicast* digunakan dalam komunikasi *point-to-point* atau *one-to-one*. (Setiawan, 2014)
- **Alamat Broadcast**, merupakan alamat IPv4 yang didesain agar diproses oleh setiap *node* IP dalam segmen jaringan yang sama. Alamat broadcast digunakan dalam komunikasi *one-to-everyone*. (Setiawan, 2014)
- **Alamat Multicast**, merupakan alamat IPv4 yang didesain agar diproses oleh satu atau beberapa node dalam segmen jaringan yang sama atau berbeda. Alamat multicast digunakan dalam komunikasi *one-to-many*. (Setiawan, 2014)

### 2.11.3. Kelas IPv4

Kelas-kelas alamat jaringan versi 4 :

1. **Kelas A** adalah alamat jaringan berskala besar yang mempunyai nilai oktet pertama 1-126 (desimal) dan oktet pertama 0xxx xxx (biner).
2. **Kelas B** adalah alamat jaringan yang digunakan khusus untuk jaringan berskala menengah sampai besar yang mempunyai nilai oktet pertama 128-191 (desimal) dan oktet pertama 10xx xxx (biner).
3. **Kelas C** adalah alamat jaringan yang digunakan untuk jaringan berskala kecil yang mempunyai nilai oktet pertama 192-223 (desimal) dan oktet pertama 110x xxx (biner).
4. **Kelas D** adalah alamat jaringan yang disediakan khusus hanya untuk IP *multicast* yang mempunyai nilai oktet pertama 224-239 (desimal) dan oktet

pertama 1110 xxx (biner).

5. **Kelas E** adalah alamat jaringan yang bersifat eksperimental atau percobaan dan dicadangkan untuk kegunaan di kemudian hari nantinya yang mempunyai nilai oktet pertama 240-255 (desimal) dan oktet pertama 1111 xxxx (biner). (Setiawan, 2014)

## 2.12. TCP/IP (Transmission Control Protocol)

TCP adalah sekumpulan protocol yang didesain untuk melakukan fungsi komunikasi pada jaringan computer. TCP/IP terdiri dari sekumpulan protocol komunikasi yang bertanggung jawab atas bagian tertentu dari komunikasi data. Jadi, TCP/IP inilah yang memungkinkan kumpulan computer untuk berkomunikasi dan bertukar data dalam suatu jaringan. TCP/IP dapat diterapkan dengan mudah disetiap jenis computer dan interface jaringan karena sebagian besar isi kumpulan protocol ini tidak spesifik terhadap satu computer atau peralatan jaringan tertentu. *Transmission Control Protocol* berfungsi untuk melakukan transmisi data pada *segmen*. Model protokol TCP disebut *connection oriented protocol*. Berbeda dengan model *User Datagram Protocol* (UDP) yang disebut *connectionless protocol*. (Sugeng, 2010)

Dalam konsep komunikasi data suatu jaringan computer, ada mekanisme data dari computer sumber ke computer yang dituju. Tentunya dalam proses pengiriman yang terjadi tidak semudah yang dipikirkan. Alasan pertama, computer tujuan berada jauh dari computer sumber sehingga packet data yang dikirimkan bias saja hilang atau rusak ditengah jalan. Alasan lainnya, mungkin computer tujuan sedang mengirim atau menunggu data dari computer sumber yang lain. Tentunya paket data yang akan dikirimkan diharapkan sampai dengan

tepat tanpa terjadi kerusakan. Untuk mengatur mekanisme komunikasi data tersebut dibutuhkan pengaturan proses pengiriman data yang dikenal sebagai protocol. Protokol adalah sebuah perangkat lunak yang melekat pada system operasi. (Sugeng, 2010)

### **2.13. Lapisan Network**

Lapisan network bertanggung jawab mengirim dan menerima data dari media fisik. Media fisik ini berupa kabel, serat optic atau gelombang radio. Karena tugasnya ini, protocol pada layer ini harus mampu menerjemahkan sinyal listrik menjadi data digital yang dimengerti oleh computer yang berasal dari peralatan lain yang sejenis. Pada lapisan internet bertanggung jawab dalam proses pengiriman paket ke alamat yang tepat. Pada layer ini terdapat tiga macam protokol, yaitu IP, ARP, dan ICMP. IP (Internet Protocol) berfungsi untuk menyampaikan paket data ke alamat yang tepat. ARP (Address Resolution Protocol) ialah protokol yang digunakan untuk menemukan alamat hardware dari host/komputer yang terletak pada network yang sama. Sedangkan ICMP (Internet Control Message Protocol) ialah protokol yang digunakan untuk mengirimkan pesan dan melaporkan kegagalan pengiriman data. (Sugeng, 2010)

### **2.14. Lapisan Transport**

Layer transport, berisi protocol yang bertanggung jawab untuk berkomunikasi antara dua host. Pada lapisan transport ini menggunakan *Acknowledgement positif* dan *Acknowledgement negative* pada aliran datanya. *Acknowledgement positif* akan memberitahukan pesan apabila data yang ditransfer telah sampai sedangkan *Acknowledgement negative* jika paket yang

ditransfer tidak sampai ke tujuan maka akan terjadi pengiriman ulang. Kedua protocol tersebut ialah TCP dan UDP. (Sugeng, 2010)

### **2.15. Prinsip Kerja TCP/IP**

Pada saat melakukan tugasnya, protokol TCP memiliki beberapa prinsip kerja. Prinsip kerja sebuah protokol ini akan menjadi referensi bagi pembuat program atau admin jaringan untuk memilih protokol apa yang nanti akan digunakan untuk bisa melakukan transmisi data. (Sugeng, 2010)

#### **1. Connection Oriented**

Sebelum data dapat ditransmisikan antara dua host, dua proses yang berjalan pada lapisan aplikasi harus melakukan negosiasi untuk membuat sesi koneksi terlebih dahulu. Proses pembuatan koneksi TCP disebut juga dengan "Three-way Handshake". Tujuan metode ini adalah agar dapat melakukan sinkronisasi terhadap nomor urut dan nomor acknowledgement yang dikirimkan oleh kedua pihak dan saling bertukar ukuran TCP Window.

Client : SYN -> Server : Client akan mengirimkan SYN ke server

Server : SYN-ACK -> Client : Server merespon SYN Client dengan mengirimkan SYN-ACK ke Client

Client : ACK -> Server : Setelah menerima SYN-ACK dari server, client mengirim ACK ke Server.

Setelah melewati handshake tadi, baru kemudian koneksi terbentuk (established).

Bisa dikatakan device yang menggunakan protokol TCP ini akan melakukan kesepakatan terlebih dahulu sebelum transmisi data terjadi. TCP menggunakan

proses jabat tangan yang sama untuk mengakhiri koneksi yang dibuat. Hal ini menjamin dua host yang sedang terkoneksi tersebut telah menyelesaikan proses transmisi data dan semua data yang ditransmisikan telah diterima dengan baik. Koneksi TCP ditutup dengan menggunakan proses terminasi koneksi FIN (TCP connection termination). (Sugeng, 2010)

## **2. Reliable Transmission**

Data yang dikirimkan ke sebuah koneksi TCP akan diurutkan dengan sebuah nomor urut yang unik disetiap byte data dengan tujuan agar data dapat disusun kembali setelah diterima. Pada saat transmisi, bisa jadi data dipecah/difragmentasi, hilang, atau tiba di device tujuan tidak lagi urut. Pada saat data diterima, paket data yang duplikat akan diabaikan dan paket yang datang tidak sesuai dengan urutannya akan diurutkan agar dapat disusun kembali.

(Sugeng, 2010)

## **3. Error Detection**

Jika terjadi error, misalnya ada paket data yang hilang pada saat proses transmisi, bisa dilakukan pengiriman ulang data yang hilang. Untuk menjamin integritas setiap segmen TCP, TCP mengimplementasikan penghitungan TCP Checksum. (Sugeng, 2010)

## **4. Flow Control**

Mendeteksi supaya satu host tidak mengirimkan data ke host lainnya terlalu cepat. Flow Control akan menjadi sangat penting ketika bekerja di lingkungan dimana device satu dengan device yang lain memiliki kecepatan komunikasi jaringan yang beragam. Sebagai contoh, ketika PC mengirimkan data ke smartphone. kemampuan PC dengan smartphone tentu berbeda. Smartphone lebih lambat

dalam memproses data yang diterima daripada PC, maka TCP akan mengatur aliran data agar smartphone tidak kewalahan. (Sugeng, 2010)

### **5. Segment Size Control**

Mendeteksi besaran MSS (maximum segment size) yang bisa dikirimkan supaya tidak terjadi IP fragmentation. MSS adalah informasi ukuran data terbesar yang dapat ditransmisikan oleh TCP dalam bentuk segment tunggal. Informasi MSS ini dalam format Bytes. Untuk performa terbaik, MSS bisa ditetapkan dengan ukuran yang cukup kecil untuk menghindari fragmentasi IP. Fragmentasi IP dapat menyebabkan hilangnya paket dan retransmisi yang berlebihan. (Sugeng, 2010)

### **6. Congestion Control**

Prinsip kerja TCP terakhir yang cukup penting adalah Congestion Control. TCP menggunakan beberapa mekanisme untuk mencegah terjadinya congestion pada network. mekanisme yang dilakukan salah satunya adalah mengatur aliran data yang masuk ke dalam network. (Sugeng, 2010)

### **2.16. Mikrotik**

Mikrotik adalah sistem operasi dan perangkat lunak yang dapat digunakan untuk menjadikan komputer menjadi router network yang handal, mencakup berbagai fitur yang dibuat untuk ip network dan jaringan wireless, cocok digunakan oleh ISP dan provider hotspot. Untuk instalasi Mikrotik tidak dibutuhkan piranti lunak tambahan atau komponen tambahan lain. Mikrotik didesain untuk mudah digunakan dan sangat baik digunakan untuk keperluan administrasi jaringan komputer seperti merancang dan membangun sebuah sistem jaringan komputer skala kecil hingga yang kompleks sekalipun. (Sinaga, 2013)

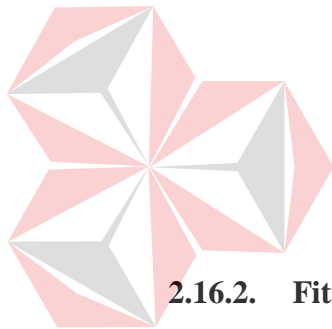
### 2.16.1. Mikrotik RB750

RB750 adalah produk routerboard yang sangat mungil dan diperuntukkan bagi penggunaan SOHO. Memiliki 5 buah port ethernet 10/100, dengan prosesor baru Atheros 400MHz. (Sinaga, 2013)



Sumber : google.com

Gambar 2.4 Mikrotik RB750

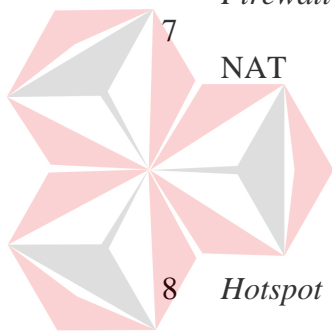


### 2.16.2. Fitur Mikrotik

Beberapa fitur yang diberikan oleh Mikrotik yaitu :

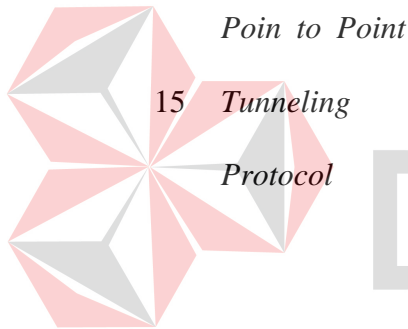
- 1 Address List : Pengelompokan IP Address berdasarkan nama  
Mendukung serial *PPP dial-in/dial-out*, dengan otentikasi CHAP,PAP, MSCHAPv1 dan
- 2 Asynchronous : MSCHAPv2, Radius, *dial on demand*, modem pool hingga 128 *ports*.  
Mendukung dalam pengkombinasian beberapa
- 3 Bonding : antarmuka *ethernet* ke dalam 1 pipa pada koneksi cepat.

- 4 Bridge : Mendukung fungsi *bridge spinning tree*, *multiple bridge interface*, *bridging firewalling*.
- 5 Data Rate Management : QoS berbasis HTB dengan penggunaan burst, PCQ, RED, SFQ, FIFO *queue*, CIR, MIR, *limit antar peer to peer*.
- 6 DHCP : Mendukung DHCP tiap antarmuka; *DHCP Relay*; *DHCP Client*, *multiple network DHCP*; *static and dynamic DHCP leases*.
- 7 Firewall dan NAT : Mendukung penyaringan koneksi *peer to peer*, *source NAT* dan *destination NAT*. Mampu memfilter berdasarkan MAC, *IP address*, *range port*, protokol IP, pemilihan opsi protokol seperti ICMP, *TCP Flags* dan MSS.
- 8 Hotspot : *Hotspot gateway* dengan otentikasi RADIUS. Mendukung *limit data rate*, SSL, HTTPS.
- 9 IPSec : Protokol AH dan ESP untuk IPSec; MODP *Diffie-Hellmann groups* 1, 2, 5; MD5 dan *algoritma SHA1 hashing*; *algoritma enkripsi* menggunakan DES, 3DES, AES-128, AES-192, AES-256; *Perfect Forwarding Secresy (PFS)* MODP *groups* 1, 2,5
- 10 ISDN : Mendukung ISDN *dial-in/dial-out*. Dengan otentikasi PAP, CHAP, MSCHAPv1 dan MSCHAPv2, Radius. Mendukung 128K *bundle*,





- Cisco HDLC, x751, x75ui, x75bui *line* protokol.
- 11 M3P : Mikrotik *Protokol Paket Packer* untuk *wireless links* dan *ethernet*.
- 12 MNDP : Mikrotik *Discovery Neighbour Protokol*, juga mendukung *Cisco Discovery Protokol* (CDP).
- 13 *Monitoring / Accounting* : Laporan *Traffic IP*, *log*, *statistik graph* yang dapat diakses melalui HTTP.
- 14 NTP : *Network Time Protokol* untuk *server* dan *clients*; sinkronisasi menggunakan *system GPS*.
- PPTP, PPPoE dan L2TP *Access Concentrator*;
- 15 *Poin to Point Tunneling Protocol* : protokol otentikasi menggunakan PAP, CHAP, MSCHAPv1, MSCHAPv2; otentikasi dan laporan Radius; enkripsi 28MPPE; kompresi untuk PPoE; *limit data rate*.
- Cache untuk FTP dan HTTP *proxy server*,
- 16 Proxy : HTTPS *proxy*; *transparent proxy* untuk DNS dan HTTP; mendukung protokol SOCKS; mendukung *parent proxy*; static DNS.
- 17 Routing : *Routing* statik dan dinamik; RIP v1/v2, OSPF v2, BGP v4.
- 18 SDSL : Mendukung *Single Line DSL*; mode pemutusan jalur koneksi dan jaringan.
- 19 *Simple Tunnel* : Tunnel IPIP dan EoIP (*Ethernet over IP*).
- 20 SNMP : *Simple Network Monitoring Protokol* mode akses



UNIVERSITAS  
Dinamika

*read-only.*

V.35, V.24, E1/T1, X21, DS3 (T3) *media types*;

*sync- PPP, Cisco HDLC; Frame Relay line*

21 Synchronous : *protokol; ANSI-617d (ANDI atau annex D) dan Q933a (CCITT atau annex A); Frame Relay jenis LMI.*

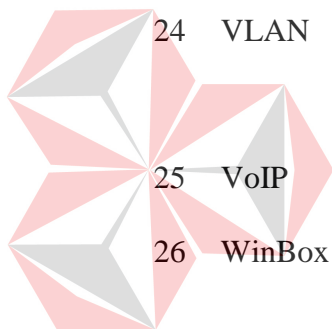
22 Tool : *Ping, Traceroute; bandwidthtest; ping flood; telnet; SSH; packet sniffer; Dinamik DNS update.*

23 UPnP : *Mendukung antarmuka Universal Plug and Play Mendukung Virtual LAN IEEE 802.1q untuk*

24 VLAN : *jaringan ethernet dan wireless; multiple VLAN; VLAN bridging.*

25 VoIP : *Mendukung aplikasi voice over IP.*

26 WinBox : *Aplikasi mode GUI untuk meremote dan mengkonfigurasi MikroTik RouterOS serta VRRP yang mendukung Virtual Router Redundant Protocol. (Sinaga, 2013)*



## 2.17. Layanan

Sebuah sistem yang terdiri atas komputer-komputer yang didesain untuk dapat berbagi sumber daya, berkomunikasi, dan dapat mengakses informasi. Tujuannya agar setiap bagian dari jaringan komputer dapat meminta dan memberikan layanan. Ada beberapa layanan untuk media pengiriman seperti FTP (*File Transfer Protocol*). FTP tersebut memiliki 2 jenis, yaitu FTP server dan FTP client. Beberapa ringkasan mengenai FTP server dan FTP client, sebagai berikut :

### 2.17.1. FTP Server

*File Transfer Protocol (FTP) Server* merupakan perangkat lunak yang bertanggung jawab untuk menerima permintaan protokol FTP dari *Client*. FTP ini berfungsi untuk *mendownload* atau *mengupload file* antar komputer. (Ozan, 2012)

### 2.17.2. FTP Client

FTP *Client* merupakan aplikasi untuk mengelola dan *mentransfer file* antar *Client* dan *Server*. Pada umumnya digunakan untuk *mendownload file* ke *Server*. Ada beberapa aplikasi FTP diantaranya Filezilla, FireFTP, dan masih banyak lagi. (Ozan, 2012)

## 2.18. Network Monitoring

Monitoring jaringan dibutuhkan untuk melakukan pengawasan pada jaringan yang dilakukan, agar jaringan tersebut selalu terkontrol dan apabila terputus dapat diketahui langsung oleh *user*. Pada tugas akhir ini *software* yang digunakan untuk monitoring jaringan yaitu Wireshark.

### 2.18.1. Wireshark

Wireshark merupakan salah satu *tool monitoring* jaringan yang berfungsi untuk mengawasi lalu lintas pada jaringan komputer dan dapat menganalisa keseluruhan jaringan komputer. ( Cahyaningtyas, 2013) Logo wireshark dapat dilihat pada Gambar 2.5



Sumber: <http://www.wireshark.org>

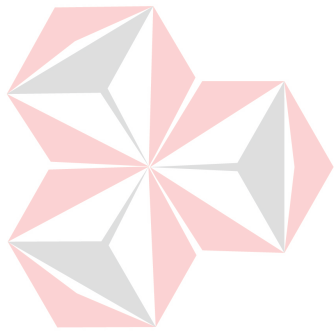
Gambar 2.5 Wireshark

Wireshark dapat melihat dan menyimpan informasi mengenai paket keluar dan masuk dalam jaringan yang terkirim dan diterima.

#### **2.18.2. Tujuan dan Manfaat Wireshark**

Manfaat dari software Wireshark, sebagai berikut :

- Menangkap informasi yang dikirim dan diterima
- Mengetahui aktivitas dalam jaringan komputer
- Mengetahui dan menganalisa kinerja jaringan komputer
- Mengamati keamanan jaringan komputer ( Cahyaningtyas, 2013)



UNIVERSITAS  
**Dinamika**

## BAB III

### METODE PENELITIAN

#### 3.1. Metode Penelitian

Metode penelitian yang digunakan dalam tugas akhir ini adalah studi kepustakaan, percobaan dan analisis.

##### 3.1.1. Studi Kepustakaan

Awal mulanya, jaringan internet tidak didesain agar mampu membedakan perlakuan untuk setiap tipe trafik yang berbeda. Hal ini menjadi masalah ketika terdapat berbagai macam trafik yang mempunyai tingkat kebutuhan pelayanan yang berbeda-beda. Ketika trafik yang sensitif terhadap *delay* harus bersaing untuk memperebutkan sumber daya jaringan, trafik tersebut akan dirugikan. Apabila ada aplikasi yang mengirimkan trafik dan menghabiskan sumber daya pada jaringan, maka trafik lain yang ingin masuk tentu saja akan dirugikan oleh trafik yang rakus sumber daya tersebut. Selain *Quality of Service* (QoS), dibutuhkan pula mekanisme *traffic engineering* untuk mengatasi permasalahan yang muncul ketika jalur mengalami keadaan kongesti. Salah satu protokol yang mampu menyediakan layanan *traffic engineering* dengan fleksibel adalah *Multiprotocol Label Switching* (MPLS). Pada MPLS ini mempunyai banyak *type*, salah satunya yang digunakan dalam penelitian ini yaitu MPLS-TE. MPLS-TE (*Multiprotocol Label Switching – Traffic Engineering*) ini digunakan untuk mengatasi dan mengoptimalkan jalur trafik (Rijayana, 2005). Dalam pengujian pada tugas akhir ini, menggunakan parameter QoS (*Quality of Service*) antara lain *latency (delay)*, *jitter*, *throughput* dan *packet loss*. *Latency (delay)* digunakan

untuk mengukur waktu transmisi yang dibutuhkan dari sumber ke tujuan. *Jitter* didefinisikan sebagai variasi delay yang diakibatkan oleh panjang queue data suatu pengolahan data dan proses penggabungan paket – paket data diakhir pengiriman akibat kegagalan sebelumnya. Berikutnya adalah *Throughput* merupakan jumlah total kedatangan paket yang diamati pada *destination* selama interval waktu tertentu dibagi oleh durasi interval waktu dan yang terakhir adalah *Packet loss* merupakan paket data yang hilang pada saat pengiriman.

### 3.1.2. Percobaan

Prosedur percobaan adalah prosedur dimana melakukan percobaan system, yaitu melakukan konfigurasi alat dan melakukan percobaan pengiriman data.

Dengan mencoba fitur – fitur yang ada. Setelah itu dilakukan percobaan pengujian alat dan aplikasi, untuk mengetahui apa benar dapat diterapkan pada alat tersebut atau tidak.

### 3.1.3. Analisis

Prosedur analisis dilakukan untuk mengetahui sebab akibat dari suatu masalah yang ditemukan. Prosedur ini masih berhubungan dengan prosedur percobaan dengan prosedur studi kepustakaan. Jadi, masalah pada kedua prosedur tersebut akan dianalisis untuk menemukan sumber permasalahannya.

## 3.2. Prosedur Penelitian

Prosedur ini menjelaskan tentang langkah – langkah yang akan dilakukan untuk membangun sistem ini, serta langkah – langkah apa saja yang akan dilakukan untuk menguji system tersebut. Berikut merupakan prosedur penelitian:

Tabel 3.1 Prosedur penelitian

<b>1. Pengumpulan data dan parameter</b>	Mikrotik, QoS, MPLS, MPLS-TE	Data Video dan Data Audio	Menentukan parameter analisis
<b>2. Desain dan pembuatan topologi</b>	Menentukan topologi MPLS-TE	Menentukan server dan client	
<b>3. Konfigurasi system</b>	Konfigurasi MPLS dan MPLS-TE pada mikrotik	Instalasi aplikasi Filezilla pada PC server dan PC client	Instalasi wireshark pada PC server dan PC client
<b>4. Menjalankan system</b>	Mengirimkan data audio melalui server ke client	Mengirimkan data video melalui server ke client	
<b>5. Mengolah data</b>	Monitoring data dari wireshark	Diolah menggunakan rumus pada Ms. Excel	
<b>6. Pengujian system dan plotting</b>	Menganalisis pengujian data	Membuat hasil plotting dari analisis pengujian data berupa grafik	

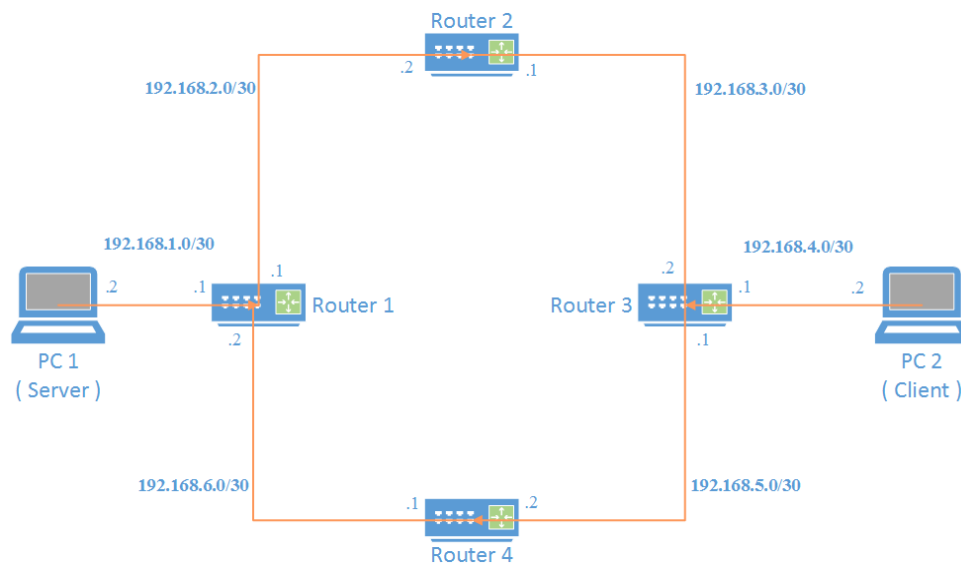
### 3.2.1. Pengumpulan Data dan Parameter Penelitian

Dalam tahap ini akan dilakukan pengumpulan data yang akan digunakan untuk melakukan pengujian. Terdapat beberapa data yang akan digunakan dalam pengujian system yaitu data *audio* dan data *video*. Kedua data tersebut didapatkan pada saat browsing di *internet*. Ada 3 jenis *video* dan 3 jenis *audio* dengan ukuran yang berbeda – beda dan menggunakan aplikasi FTP salah satunya *FileZilla* yang akan digunakan untuk mengirimkan data dari *Server* ke *Client*. Setelah itu, PC *Server* dan *Client* menjalankan aplikasi *Wireshark*. *Wireshark* akan diset agar memonitoring paket data TCP dan hasil monitoring tersebut akan dilakukan pada sisi *Server* dan *Client*. Setelah itu hasil dari monitoring tersebut akan diolah untuk mendapatkan nilai dari *latency (delay)*, *jitter*, *throughput* dan *packet loss*.

### 3.2.2. Desain dan Pembuatan Topologi

Implementasi dan analisis MPLS-TE pada jaringan berbasis mikrotik ini akan dijelaskan lebih baik melalui desain topologi yang dapat dilihat pada Gambar 3.1. Terdapat 2 router sebagai *Label Switching Router* (LSR), 2 router berikutnya untuk *Label Edge Router* (LER) serta 2 *Personal Computer* (PC) sebagai *Server* dan *Client* yang masing – masing tersambung pada LER. Pada PC *Server* yang diinstall aplikasi *FileZilla Server* yang dijadikan sebagai penerima permintaan dari *Client* dan PC *Client* akan diinstall *FileZilla Client* yang digunakan untuk mengelola atau mentransfer data. Data dari PC *Server* yang terus menerus dikirimkan melalui router LER yang akan diteruskan ke router LSR yang berfungsi mengatur trafik saat paket memasuki jaringan MPLS serta dalam jaringan MPLS berperan menetapkan LSP dengan menggunakan teknik label swapping dengan kecepatan yang ditetapkan. Kemudian data keluar dari jaringan MPLS melewati LSR yang berfungsi untuk mengatur trafik pada saat paket meninggalkan jaringan MPLS dan menuju ke LER. Dalam tugas akhir ini panelis menggunakan topologi seperti Gambar 3.1 dikarenakan untuk mengatasi jika ada jalur atau node yang *down*, maka akan lewat jalur yang lainnya agar paket sampai ke tujuannya.





Gambar 3.1 Topologi MPLS-TE

Berikut adalah alamat IP yang digunakan untuk menghubungkan antar

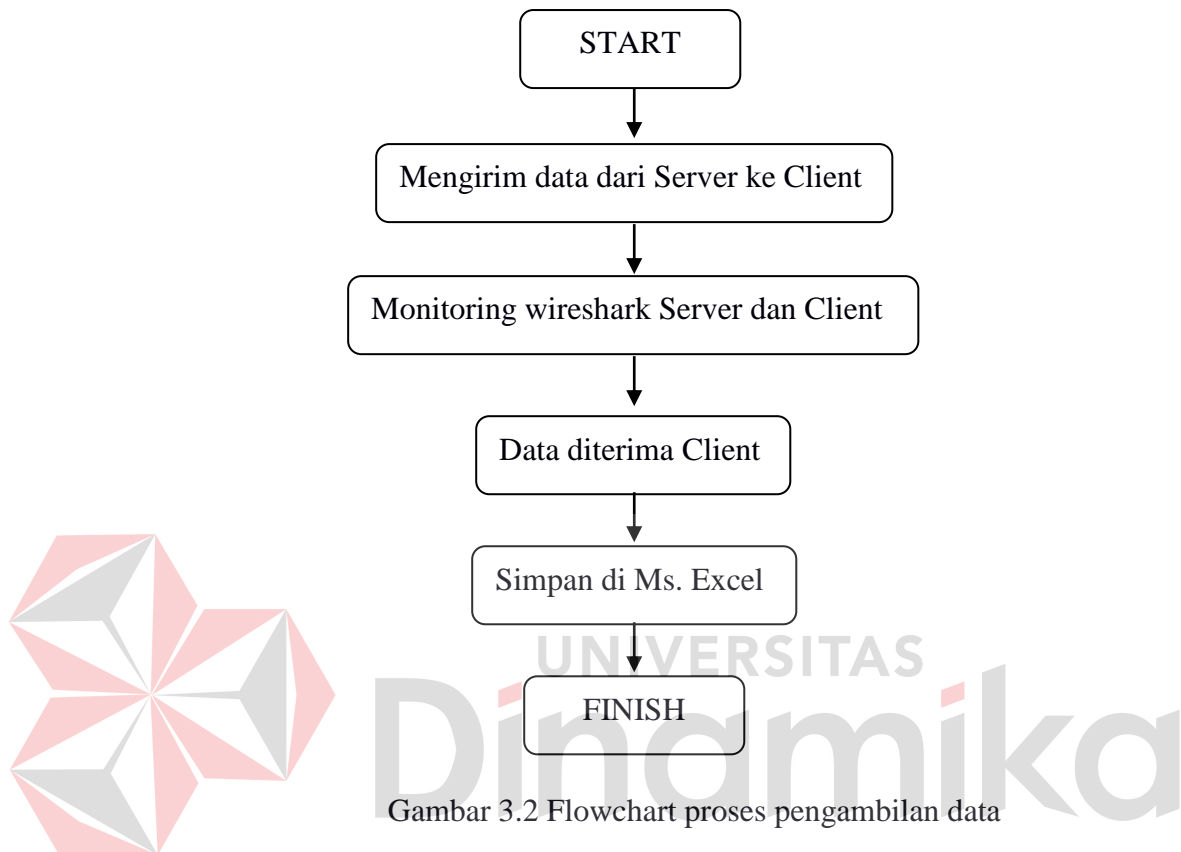
*Router* dan PC (Personal Computer). Dapat dilihat pada tabel 3.2.

Tabel 3.2 IP Address pada Router

Device	Interface	IP Address version 4	IP Interface Loopback
<b>Router 1</b>	Ether 1	192.168.1.1/30	
	Ether 2	192.168.2.1/30	
	Ether 3	192.168.6.2/30	
	Loopback		1.1.1.1/32
<b>Router 2</b>	Ether 3	192.168.2.2/30	
	Ether 2	192.168.3.1/30	
	Loopback		2.2.2.2/32
<b>Router 3</b>	Ether 1	192.168.4.1/30	
	Ether 2	192.168.5.1/30	
	Ether 3	192.168.3.2/30	
	Loopback		3.3.3.3/32
<b>Router 4</b>	Ether 3	192.168.5.2/30	
	Ether 2	192.168.6.1/30	
	Loopback		4.4.4.4/32

PC 1 ( Server )	Port 1	192.168.1.2/30	
PC 2 ( Client )	Port 1	192.168.4.2/30	

### 3.2.3. Proses Monitoring dan Pengambilan Data



Gambar 3.2 Flowchart proses pengambilan data

Pada gambar 3.2 dapat dilihat bahwa dari PC *Server* akan mengirimkan data ke PC *Client* melalui aplikasi *FileZilla* dan data dikirim melewati router mikrotik sebelum data dikirim, aktifkan wireshark pada PC *Server* dan *Client* yang akan memonitoring pengiriman paket. Selanjutnya, data akan masuk pada PC *Client* dan wireshark akan memonitoring paket yang sampai dan setelah selesai, maka hasil dari monitoring tersebut disimpan ke dalam ms.excel untuk diproses lebih lanjut.

### 3.2.4. Konfigurasi Sistem

#### 1. Konfigurasi Nama *Router*, *Interface* dan *IP Address*

Pada masing – masing *router* diberi nama sesuai dengan topologi. Setelah itu beri nama masing – masing *interface router* dan konfigurasi *IP Address* pada masing – masing *interface* sesuai dengan tabel 3.2.

#### 2. Konfigurasi *Loopback Interface*

Konfigurasi *loopback interface* pada masing – masing *interface* yang ada pada *router*.

#### 3. Konfigurasi *Dynamic Routing*

Mengkonfigurasi *routing* dengan menggunakan *routing Open Shortest Path First* (OSPF) pada setiap *router*.

#### 4. Konfigurasi *MPLS*

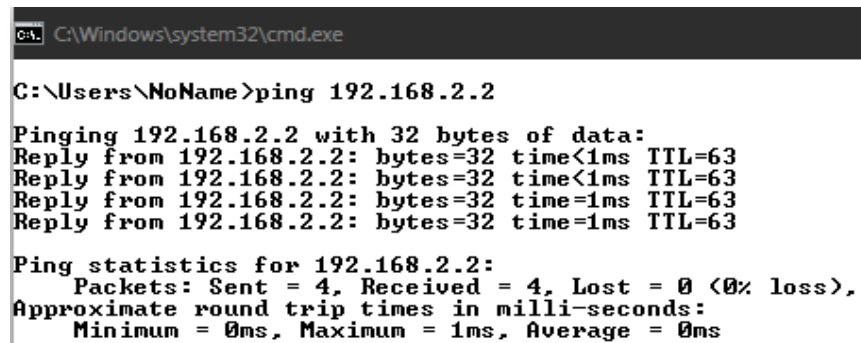
Konfigurasi *MPLS* dengan menambahkan *LDP* pada *interface* dan *transport address* pada masing – masing *router* untuk mengaktifkan *DP* yang berfungsi untuk menambahkan label serta mendistribusikannya.

#### 5. Konfigurasi *Traffic Engineering*

Konfigurasi *Traffic engineering* pada *interface router*, pengaturan *resource reservation*. Pada dasarnya parameter ini mengontrol seberapa sering jalan *RSVP* dan setiap *host penerima*, mengirimkan permintaan pesan pada *interface tertentu*. Jadi, seperti *hello-interval* yang ada pada *protocol OSPF* dan *bandwidth* yang digunakan pada *MPLS-TE* bukan untuk menciptakan *bandwidth* yang baru, tetapi *bandwidth* tersebut digunakan untuk memanfaatkan *path-protection*.

### 3.2.5. Uji Koneksi

Pada uji koneksi ini merupakan untuk melihat jaringan, apakah pada sisi client sudah terhubung dengan 4 router dan pc server yang telah dikonfigurasi. Pada gambar 3.3 uji koneksi melalui client ke router 2, untuk mengetahui apakah router 2 sudah bisa terhubung dengan client.



```
C:\Windows\system32\cmd.exe

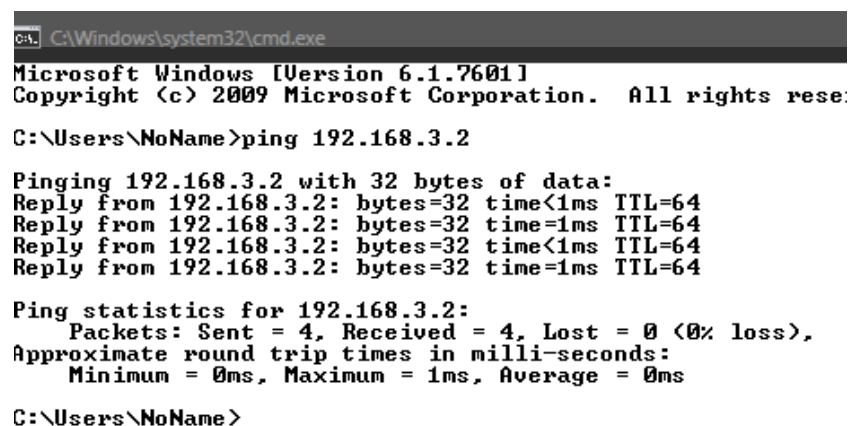
C:\Users\NoName>ping 192.168.2.2

Pinging 192.168.2.2 with 32 bytes of data:
Reply from 192.168.2.2: bytes=32 time<1ms TTL=63
Reply from 192.168.2.2: bytes=32 time<1ms TTL=63
Reply from 192.168.2.2: bytes=32 time=1ms TTL=63
Reply from 192.168.2.2: bytes=32 time=1ms TTL=63

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

Gambar 3.3 Ping Client ke Router 2

Pada gambar 3.4 uji koneksi melalui client ke router 3 yang terhubung dengan server, untuk mengetahui apakah router 3 sudah bisa terhubung dengan client.



```
C:\Windows\system32\cmd.exe

Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\NoName>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:
Reply from 192.168.3.2: bytes=32 time<1ms TTL=64
Reply from 192.168.3.2: bytes=32 time=1ms TTL=64
Reply from 192.168.3.2: bytes=32 time<1ms TTL=64
Reply from 192.168.3.2: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\NoName>
```

Gambar 3.4 Ping Client ke Router 3

Pada gambar 3.5 uji koneksi melalui client ke router 4, untuk mengetahui apakah router 4 sudah bisa terhubung dengan client.

```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\NoName>ping 192.168.6.1

Pinging 192.168.6.1 with 32 bytes of data:
Reply from 192.168.6.1: bytes=32 time<1ms TTL=63
Reply from 192.168.6.1: bytes=32 time=1ms TTL=63
Reply from 192.168.6.1: bytes=32 time<1ms TTL=63
Reply from 192.168.6.1: bytes=32 time=1ms TTL=63

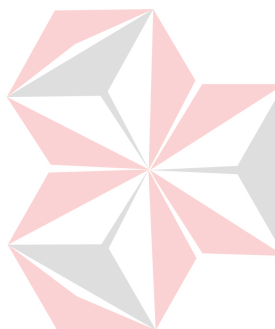
Ping statistics for 192.168.6.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\NoName>

```

Gambar 3.5 Ping Client ke Router 4

Pada gambar 3.6 uji koneksi melalui client ke router 3 yang terhubung dengan server, untuk mengetahui apakah router 3 sudah bisa terhubung dengan client.



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\NoName>ping 192.168.5.1

Pinging 192.168.5.1 with 32 bytes of data:
Reply from 192.168.5.1: bytes=32 time<1ms TTL=64
Reply from 192.168.5.1: bytes=32 time<1ms TTL=64
Reply from 192.168.5.1: bytes=32 time=1ms TTL=64
Reply from 192.168.5.1: bytes=32 time=1ms TTL=64

Ping statistics for 192.168.5.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

```

Gambar 3.6 Ping Client ke Router 3

Pada gambar 3.7 uji koneksi melalui client ke server, untuk mengetahui apakah server sudah bisa terhubung dengan client.

```

C:\Windows\system32\cmd.exe

C:\Users\NoName>ping 192.168.4.2

Pinging 192.168.4.2 with 32 bytes of data:
Reply from 192.168.4.2: bytes=32 time<1ms TTL=128
Reply from 192.168.4.2: bytes=32 time<1ms TTL=128
Reply from 192.168.4.2: bytes=32 time<1ms TTL=128
Reply from 192.168.4.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.4.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

```

Gambar 3.7 Ping Client ke Server

Setelah semua terhubung, selanjutnya adalah menyiapkan data audio dan data video yang akan dikirim. Pada tabel 3.3 dan 3.4 terdapat 3 (tiga) audio dan 3 (tiga) video dengan ukuran yang berbeda – beda serta bandwidth yang berbeda, dimana nantinya akan digunakan dalam penelitian ini.

Tabel 3.3 Data audio (1)

Bandwidth	$\Sigma$ Data	Ukuran Audio
128 Kbps	3	2.037 KB
		6.883 KB
		9.040 KB
512 Kbps	3	2.037 KB
		6.883 KB
		9.040 KB
1Mbps	3	2.037 KB
		6.883 KB
		9.040 KB

Tabel 3.4 Data video (2)

Bandwidth	$\Sigma$ Data	Ukuran Video
128 Kbps	3	6.434 KB
		10.045 KB
		13.839 KB
512 Kbps	3	6.434 KB
		10.045 KB
		13.839 KB
1Mbps	3	6.434 KB
		10.045 KB
		13.839 KB

### 3.2.6. Pengolahan Data

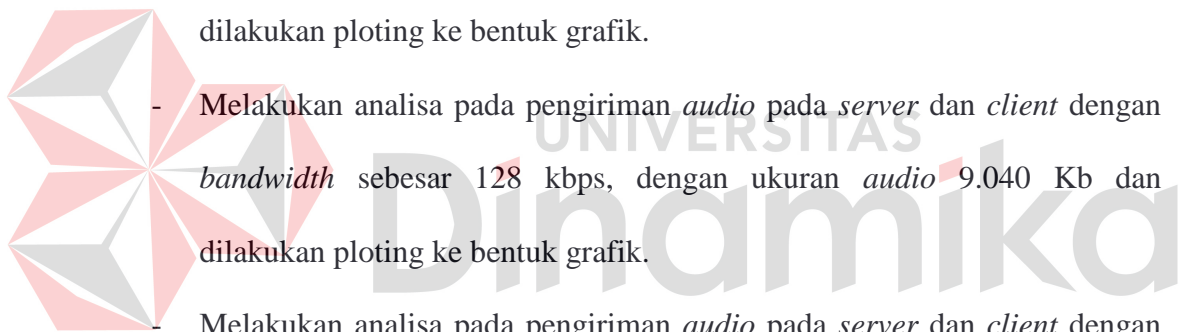
Pada pengolahan data ini, akan menjelaskan yang dilakukan untuk mendapatkan nilai dari tiap parameter yang digunakan. Parameter tersebut meliputi *Latency (delay)*, *Jitter*, *Throughput* dan *Packet Loss*. Pertama, menjalankan aplikasi *FileZilla* pada *PC Server* dan *Client*, setelah itu menyiapkan data yang akan dikirimkan, setting *bandwidth router*, setelah itu aktifkan *wireshark* untuk memonitoring jalannya suatu proses pengiriman data dari *Server* ke *Client*. Sebelumnya pada *wireshark* di *set* TCP agar yang tertangkap yaitu data TCP, data TCP meliputi FTP, FTP-DATA dan TCP. Proses selanjutnya, menganalisa hasil yang tertangkap oleh aplikasi *wireshark* dan dihitung dengan menggunakan parameter parameter QoS agar dapat memberikan kesimpulan pada hasil yang didapat.

### 3.2.7. Pengujian Sistem dan Plotting

Plotting ini menampilkan hasil dari analisis yang telah dilakukan. Berikut penjelasannya :

1. Pengujian pengiriman *Audio* berdasarkan *bandwidth* yang digunakan dan ukuran *audio* yang berbeda.

- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 128 kbps, dengan ukuran *audio* 2.037 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 128 kbps, dengan ukuran *audio* 6.883 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 128 kbps, dengan ukuran *audio* 9.040 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 512 kbps, dengan ukuran *audio* 2.037 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 512 kbps, dengan ukuran *audio* 6.883 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 512 kbps, dengan ukuran *audio* 9.040 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan



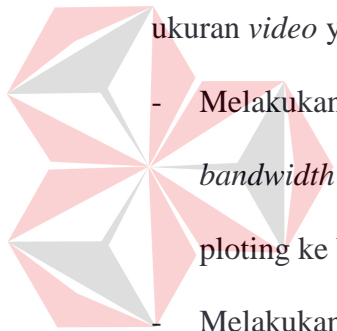


*bandwidth* sebesar 1 Mbps, dengan ukuran *audio* 2.037 Kb dan dilakukan plotting ke bentuk grafik.

- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 1 Mbps, dengan ukuran *audio* 6.883 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *audio* pada *server* dan *client* dengan *bandwidth* sebesar 1 Mbps, dengan ukuran *audio* 9.040 Kb dan dilakukan plotting ke bentuk grafik.

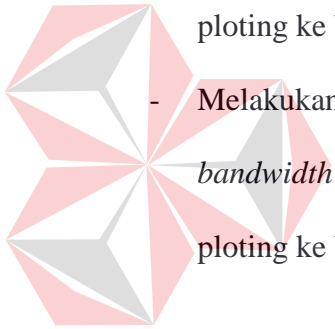
2. Pengujian pengiriman *Video* berdasarkan *bandwidth* yang digunakan dan ukuran *video* yang berbeda.

- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 128 kbps, dengan ukuran *video* 6.434 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 128 kbps, dengan ukuran *video* 10.045 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 128 kbps, dengan ukuran *video* 13.839 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 512 kbps, dengan ukuran *video* 6.434 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan



*bandwidth* sebesar 512 kbps, dengan ukuran *video* 10.045 Kb dan dilakukan plotting ke bentuk grafik.

- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 512 kbps, dengan ukuran *video* 13.839 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 1 Mbps, dengan ukuran *video* 6.434 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 1 Mbps, dengan ukuran *video* 10.045 Kb dan dilakukan plotting ke bentuk grafik.
- Melakukan analisa pada pengiriman *video* pada *server* dan *client* dengan *bandwidth* sebesar 1 Mbps, dengan ukuran *video* 13.839 Kb dan dilakukan plotting ke bentuk grafik.



UNIVERSITAS  
Dinamika

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1. Kebutuhan Sistem

Sebelum melakukan pengujian MPLS-TE yang berjalan pada jaringan di router mikrotik, dibutuhkan *hardware* dan *software* yang digunakan agar implementasi dapat berjalan lancar. Adapun kebutuhan *hardware* dan *software* yang digunakan dalam penelitian ini, dapat dilihat pada tabel 4.1 dan 4.2.

Tabel 4.1 kebutuhan *Hardware*

Hardware	Jumlah Unit	Keterangan
PC Server	1	Sony Vaio, Core i5, RAM 2 Gb, Harddisk 500 Gb
PC Client	1	Acer Aspire V3-471G, Core i5, RAM 6 Gb, Harddisk 750 Gb
Mikrotik	4	Routerboard 750, MISP-BE AR7241 400MHz, RAM 32 Mb, NAND 64Mb, 5 eth port

Tabel 4.2 kebutuhan *Software*

Software	Keterangan
Windows 7 Profesional 64 bit	Sebagai Operating System PC Server
Windows 7 Profesional 64 bit	Sebagai Operating System PC Client
RouterOS-MIPSBE 6.26	Sebagai Operating System Router Mikrotik RB 750
Wireshark	Sebagai Aplikasi Penangkap Paket Data
Microsoft Excel	Sebagai Aplikasi Pengolahan data dan Pembuatan Grafik
FileZilla Server dan Client	Sebagai Aplikasi pengiriman data

## 4.2. Hasil Penelitian

Pada hasil penelitian ini membahas mengenai hasil analisa perbandingan antara MPLS dan MPLS-TE dengan parameter QoS, seperti *latency*, *jitter*, *throughput* dan *packet loss* untuk mengetahui kinerja dari kedua sistem tersebut.

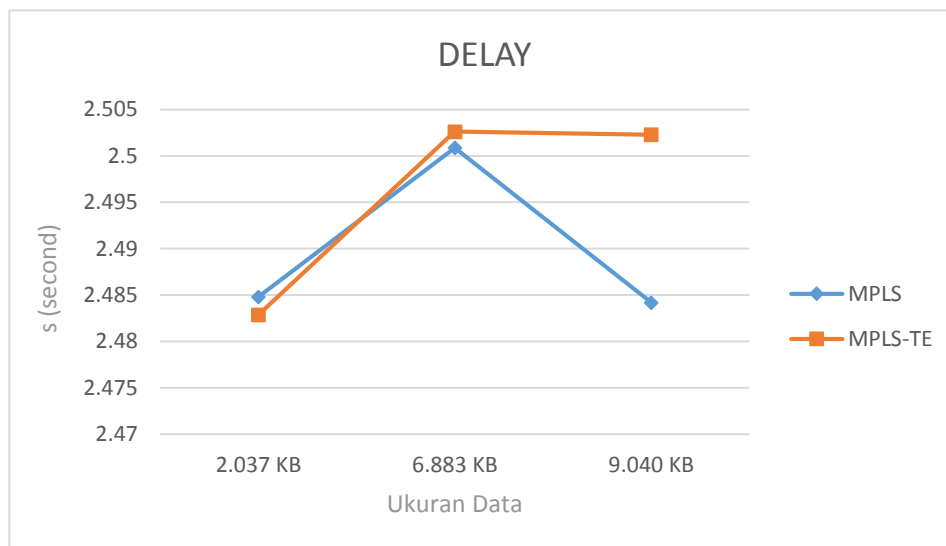
### 4.2.1. Analisis Latency bandwidth 128 Kbps

Pada tabel 4.3 merupakan hasil *latency* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 128 Kbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *latency* sebesar 2.484816 s, data audio sebesar 6.883 Kb menghasilkan nilai *latency* sebesar 2.500871 s, data audio sebesar 9.040 Kb menghasilkan nilai *latency* sebesar 2.484188 s dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *latency* 2.482884 s, data audio sebesar 6.883 Kb menghasilkan nilai *latency* 2.502616 s, data audio sebesar 9.040 Kb menghasilkan nilai *latency* 2.502294 s. Waktu *latency* pada data audio dengan *bandwidth* yang dihasilkan dari teknologi MPLS dan MPLS-TE tidak berbeda secara signifikan.

Tabel 4.3 hasil perbandingan delay audio

Bandwidth 128Kbps		
Ukuran Audio	Latency ( s)	
	MPLS	MPLS-TE
<b>2.037 KB</b>	2.484816	2.482884
<b>6.883 KB</b>	2.500871	2.502616
<b>9.040 KB</b>	2.484188	2.502294

Pada gambar 4.1 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



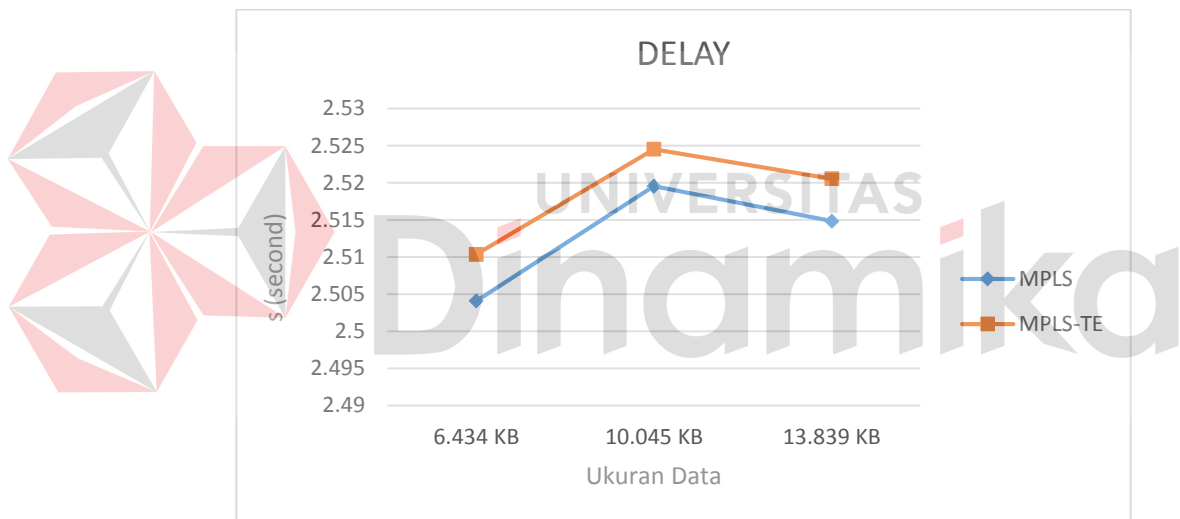
Gambar 4.1 Grafik hasil perbandingan delay audio

Pada tabel 4.4 merupakan hasil *latency* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 128 Kbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *latency* 2.504092 s, data video sebesar 10.045 Kb menghasilkan nilai *latency* 2.519528 s, data video sebesar 13.839 Kb menghasilkan nilai *latency* 2.514833 s dan pada teknologi MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai 2.510346 s, data video sebesar 10.045 Kb menghasilkan nilai *latency* 2.524521 s, data video sebesar 13.895 Kb menghasilkan nilai *latency* 2.520554 s. Dari hasil *latency* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.4 hasil perbandingan delay video

Bandwidth 128Kbps		
Ukuran Video	Latency (s)	
	MPLS	MPLS-TE
6.434 KB	2.504092	2.510346
10.045 KB	2.519528	2.524521
13.839 KB	2.514833	2.520554

Pada gambar 4.2 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.2 Grafik hasil perbandingan delay video

Dari data audio dan data video memiliki hasil yang tidak berbeda secara signifikan. Pada data audio sebesar 2.037 Kb memiliki nilai *latency* yang kecil dikarenakan ukuran datanya lebih kecil sehingga waktu *latency* yang didapat lebih kecil daripada nilai *latency* yang lainnya.

Penyebab terjadi *delay* pada teknologi MPLS dan MPLS-TE yang tidak berbeda secara signifikan, dikarenakan pada kedua teknologi tersebut terdapat proses pelabelan pada paket yang datang kemudian akan dikirimkan ke *router*

berikutnya. Penyebab selanjutnya yaitu, karena pada kedua teknologi sama-sama menggunakan *bandwidth* sebesar 128 Kbps, sehingga mendapatkan hasil *delay* yang tidak berbeda secara signifikan.

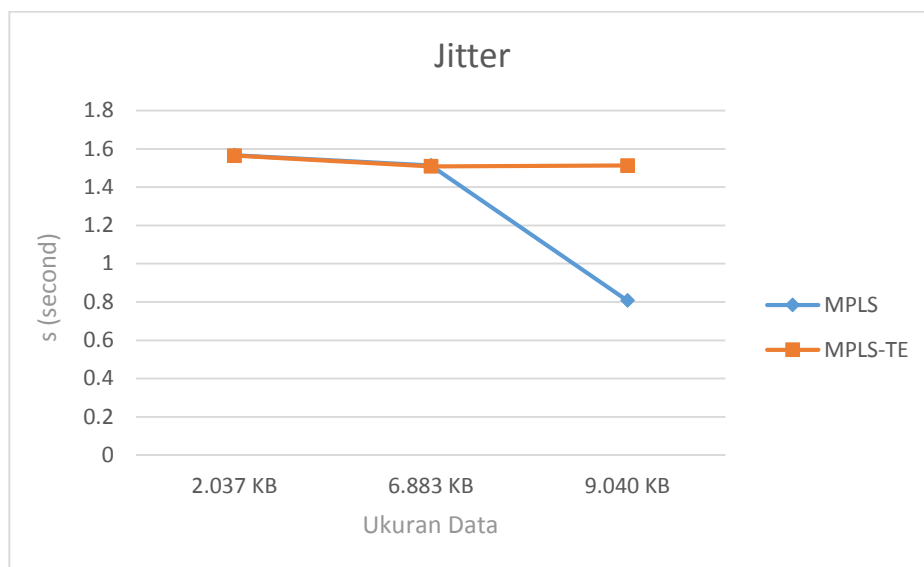
#### 4.2.2. Analisis Jitter bandwidth 128Kbps

Pada tabel 4.5 merupakan hasil *jitter* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 128 Kbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *jitter* sebesar 1.56739 s, data audio sebesar 6.883 Kb menghasilkan nilai *jitter* sebesar 1.513768 s, data audio sebesar 9.040 Kb menghasilkan nilai *jitter* sebesar 0.807514 s dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *jitter* 1.564982 s, data audio sebesar 6.883 Kb menghasilkan nilai *jitter* sebesar 1.508881 s, data audio sebesar 9.040 Kb menghasilkan nilai *jitter* sebesar 1.512484 s. Hasil yang didapat dari *variasi delay* atau *jitter* pada data audio dengan *bandwidth* 128 Kbps memiliki nilai yang tidak berbeda secara signifikan.

Tabel 4.5 hasil perbandingan jitter audio

Bandwidth 128Kbps		
Ukuran	Jitter ( s )	
Audio	MPLS	MPLS-TE
2.037 KB	1.56739	1.564982
6.883 KB	1.513768	1.508881
9.040 KB	0.807514	1.512484

Pada gambar 4.3 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE.



Gambar 4.3 Grafik hasil perbandingan jitter audio

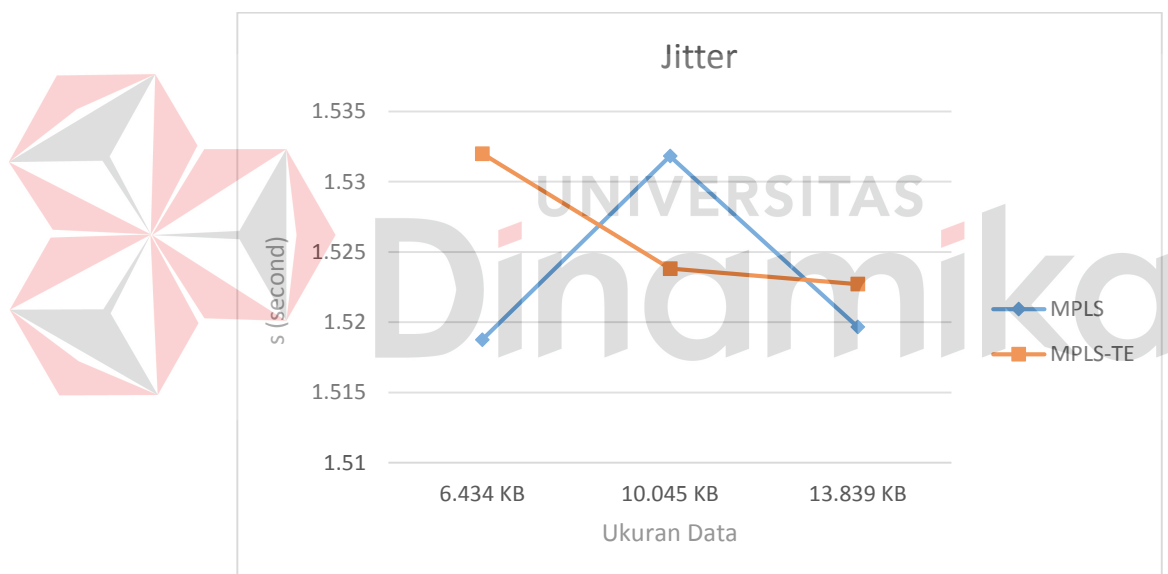
Pada tabel 4.6 merupakan hasil *jitter* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 128 Kbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *jitter* 1.518757 s, data video sebesar 10.045 Kb menghasilkan nilai *jitter* 1.531827 s, data video sebesar 13.839 Kb menghasilkan nilai *jitter* 1.51966 s dan pada teknologi MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai *jitter* 1.531996 s, data video sebesar 10.045 Kb menghasilkan nilai *jitter* 1.52382 s, data video sebesar 13.895 Kb menghasilkan nilai *jitter* 1.522715 s. Dari hasil *jitter* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.



Tabel 4.6 hasil perbandingan jitter video

Bandwidth 128Kbps		
Ukuran Video	Jitter ( s )	
	MPLS	MPLS-TE
<b>6.434 KB</b>	1.518757	1.531996
<b>10.045 KB</b>	1.531827	1.52382
<b>13.839 KB</b>	1.51966	1.522715

Pada gambar 4.4 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE.



Gambar 4.4 Grafik hasil perbandingan jitter video

Pada teknologi MPLS dan MPLS-TE dengan data audio dan video dan menggunakan *bandwidth* yang digunakan sebesar 128 Kbps memiliki hasil yang tidak berbeda secara signifikan.

Faktor-faktor yang mempengaruhi nilai *Jitter* diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan *jitter*. *Jitter* biasanya disebut

variasi *delay* yang berhubungan erat dengan *latency*, yang menunjukkan banyaknya variasi *delay* pada transmisi data dalam jaringan. Besarnya nilai *jitter* yang tidak berbeda secara signifikan, dikarenakan ada proses pelabelan pada paket yang datang kemudian akan dikirimkan ke router berikutnya dan semakin kecil *bandwidth* yang digunakan maka akan semakin besar nilai *jitter*nya.

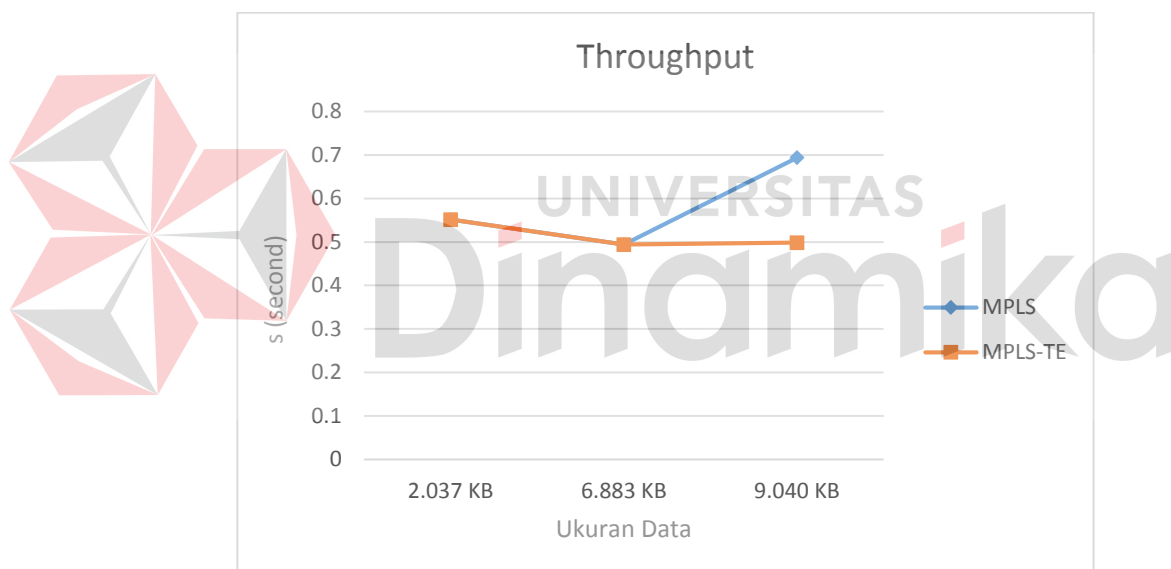
#### 4.2.3. Analisis Throughput bandwidth 128 Kbps

*Throughput* merupakan jumlah total kedatangan paket yang diamati pada waktu interval tertentu. Digunakan untuk mengukur atau menghitung kecepatan data. Pada tabel 4.7 merupakan hasil *throughput* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 128 Kbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *throughput* sebesar 0.551374, data audio sebesar 6.883 Kb menghasilkan nilai *throughput* sebesar 0.493872, data audio sebesar 9.040 Kb menghasilkan nilai *throughput* sebesar 0.693913 dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *throughput* 0.551377, data audio sebesar 6.883 Kb menghasilkan nilai *throughput* sebesar 0.493969, data audio sebesar 9.040 Kb menghasilkan nilai *throughput* sebesar 0.498437. Hasil yang didapat dari *throughput* pada data audio dengan *bandwidth* 128 Kbps memiliki nilai yang tidak berbeda secara signifikan.

Tabel 4.7 hasil perbandingan Throughput audio

Bandwidth 128Kbps		
Ukuran Audio	Throughput (bps)	
	MPLS	MPLS-TE
<b>2.037 KB</b>	0.551374	0.551377
<b>6.883 KB</b>	0.493872	0.493969
<b>9.040 KB</b>	0.693913	0.498437

Pada gambar 4.5 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.5 Grafik hasil perbandingan throughput audio

Pada tabel 4.8 merupakan hasil *throughput* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 128 Kbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *throughput* 0.495774, data video sebesar 10.045 Kb menghasilkan nilai *throughput* 0.484607, data video sebesar 13.839 Kb menghasilkan nilai *throughput* 0.480534 dan pada teknologi

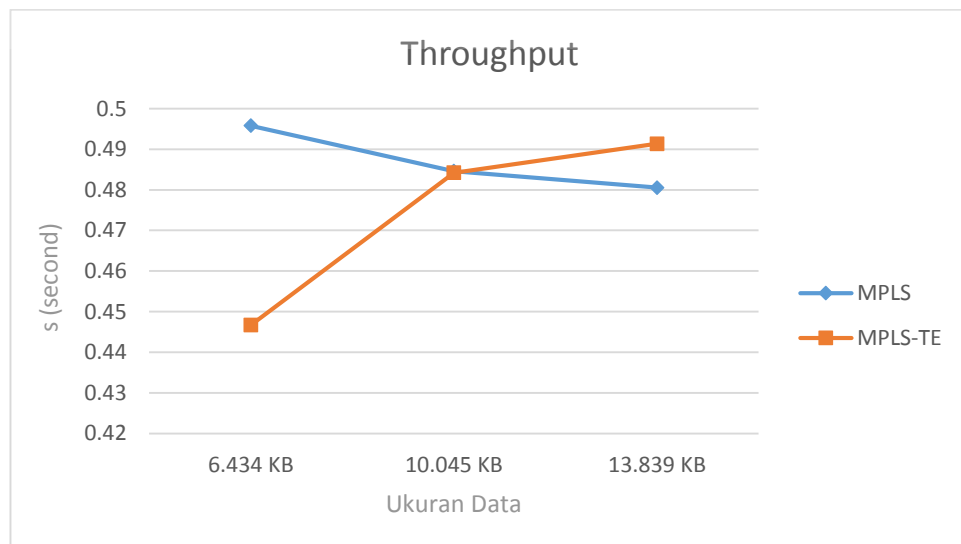
MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai 0.446686, data video sebesar 10.045 Kb menghasilkan nilai *throughput* 0.484202, data video sebesar 13.895 Kb menghasilkan nilai *throughput* 0.491345. Dari hasil *throughput* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.8 hasil perbandingan throughput video

Bandwidth 128Kbps		
Ukuran Video	Throughput (bps)	
	MPLS	MPLS-TE
6.434 KB	0.495774	0.446686
10.045 KB	0.484607	0.484202
13.839 KB	0.480534	0.491345



Pada gambar 4.6 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.6 Grafik hasil perbandingan throughput video


Pada analisis *throughput* diatas dari data audio dan video dengan bandwidth

yang digunakan sebesar 128 kbps memperlihatkan MPLS-TE dan MPLS tidak memiliki perbedaan yang cukup signifikan. Hal ini disebabkan karena pada MPLS-TE mempunyai mekanisme *tunnel* atau jalur alternative yang digunakan untuk mengatasi kemacetan dan menggunakan ukuran bandwidth yang sama dengan teknologi MPLS, sehingga hasil *throughput* tidak berbeda secara signifikan.

#### 4.2.4. Analisis Packet Loss bandwidth 128Kbps

Packet loss merupakan paket yang hilang disebabkan beberapa factor. Hasil untuk *bandwidth* 128 Kbps pada tabel 4.9 dan tabel 4.10 tidak ada data yang hilang, maka kedua data tersebut memiliki nilai *packet loss* 0%.

Tabel 4.9 hasil perbandingan packet loss audio



Bandwidth 128Kbps		
Ukuran Audio	Packet Loss (%)	
	MPLS	MPLS-TE
2.037 KB	0	0
6.883 KB	0	0
9.040 KB	0	0

Tabel 4.10 hasil perbandingan packet loss video

Bandwidth 128Kbps		
Ukuran Video	Packet Loss (%)	
	MPLS	MPLS-TE
6.434 KB	0	0
10.045 KB	0	0
13.839 KB	0	0

Dari analisis *packet loss* diatas, kedua data (audio dan video) dengan bandwidth yang digunakan 128 Kbps memiliki nilai *packet loss* yang sama yaitu 0%, dikarenakan pada saat pengiriman data tidak mengalami gangguan pada jalur yang digunakan.

#### 4.2.5. Analisis Latency bandwidth 512Kbps

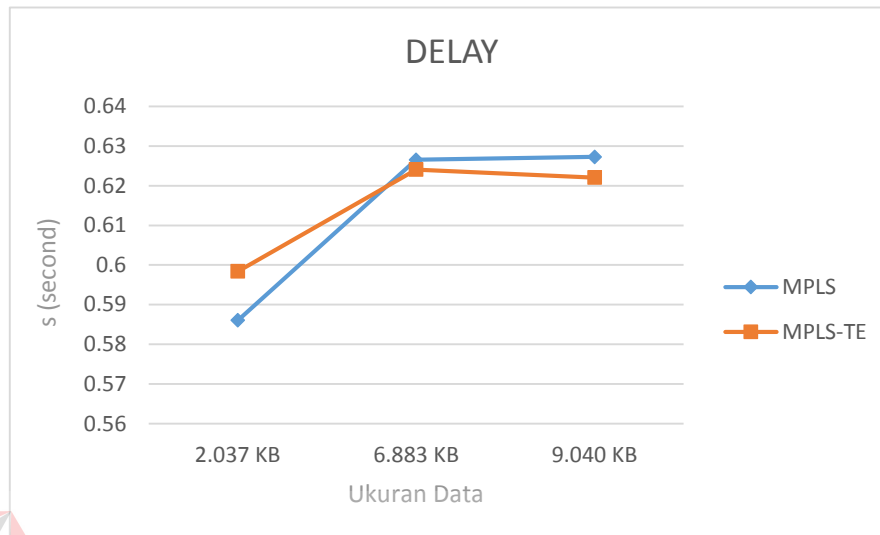
Pada tabel 4.11 merupakan hasil *latency* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 512 Kbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *latency* sebesar 0.586037 s, data audio sebesar 6.883 Kb menghasilkan nilai *latency* sebesar 0.626528 s, data audio sebesar 9.040 Kb menghasilkan nilai *latency* sebesar 0.627276 s dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *latency* 0.598383 s, data audio sebesar 6.883 Kb menghasilkan nilai *latency* 0.624078 s, data audio sebesar 9.040 Kb menghasilkan nilai *latency* 0.622035 s.

Waktu *latency* pada data audio dengan *bandwidth* yang dihasilkan dari teknologi MPLS dan MPLS-TE tidak berbeda secara signifikan.

Tabel 4.11 hasil perbandingan delay audio

Bandwidth 512Kbps		
Ukuran Audio	Latency (s)	
	MPLS	MPLS-TE
2.037 KB	0.586037	0.598383
6.883 KB	0.626528	0.624078
9.040 KB	0.627276	0.622035

Pada gambar 4.7 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



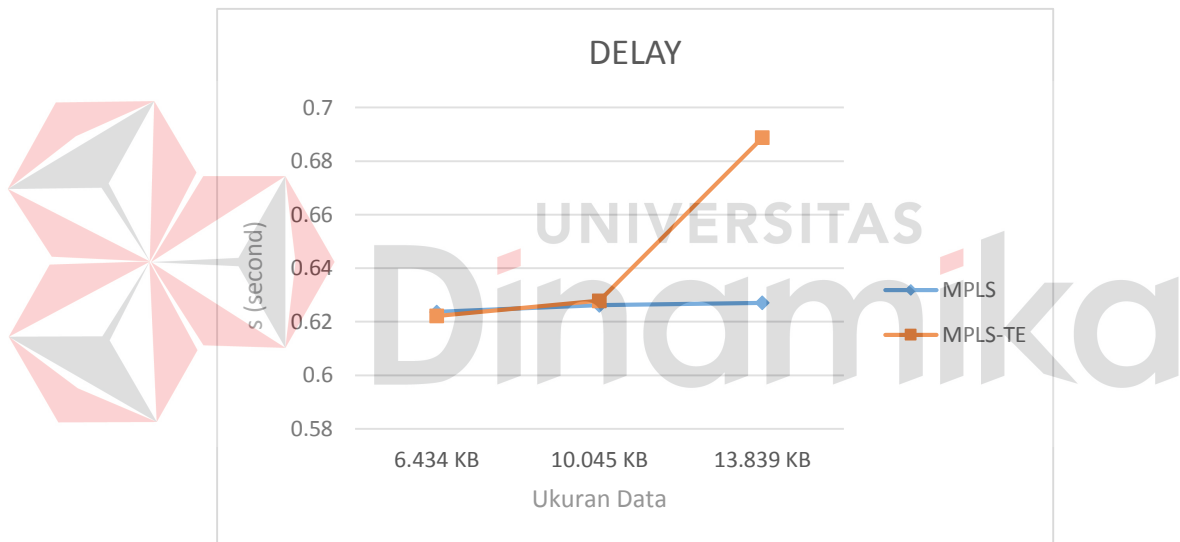
Gambar 4.7 Hasil grafik perbandingan delay audio

Pada tabel 4.12 merupakan hasil *latency* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 512 Kbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *latency* 0.623667 s, data video sebesar 10.045 Kb menghasilkan nilai *latency* 0.626156 s, data video sebesar 13.839 Kb menghasilkan nilai *latency* 0.627042 s dan pada teknologi MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai 0.62215 s, data video sebesar 10.045 Kb menghasilkan nilai *latency* 0.627872 s, data video sebesar 13.895 Kb menghasilkan nilai *latency* 0.68877 s. Dari hasil *latency* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.12 hasil perbandingan delay video

Bandwidth 512Kbps		
Ukuran Video	Latency (s)	
	MPLS	MPLS-TE
6.434 KB	0.623667	0.62215
10.045 KB	0.626156	0.627872
13.839 KB	0.627042	0.68877

Pada gambar 4.8 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.8 Hasil grafik perbandingan delay video

Dari data audio dan data video memiliki hasil yang tidak berbeda secara signifikan, dikarenakan ada proses pelabelan pada paket yang datang kemudian akan dikirimkan ke *router* berikutnya. Penyebab selanjutnya yaitu, karena pada kedua teknologi sama-sama menggunakan *bandwidth* sebesar 512 Kbps, sehingga mendapatkan hasil delay yang tidak berbeda secara signifikan.



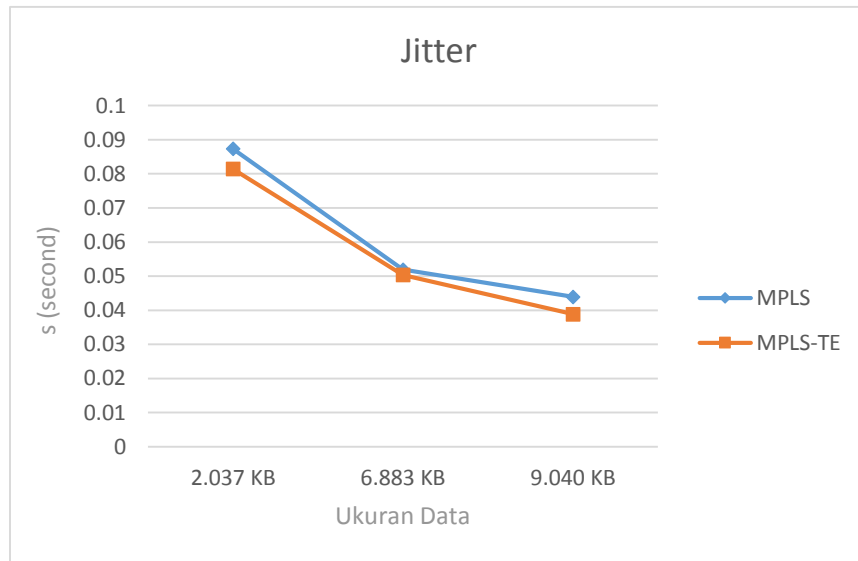
#### 4.2.6. Analisis Jitter bandwidth 512Kbps

*Jitter* atau variasi *delay* merupakan waktu kedatangan dari suatu paket ke penerima dengan waktu yang diharapkan. Digunakan untuk mengetahui waktu tunda paket yang diterima. Pada tabel 4.13 merupakan hasil *jitter* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 512 Kbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *jitter* sebesar 0.087347 s, data audio sebesar 6.883 Kb menghasilkan nilai *jitter* sebesar 0.051923 s, data audio sebesar 9.040 Kb menghasilkan nilai *jitter* sebesar 0.043929 s dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *jitter* 0.08144 s, data audio sebesar 6.883 Kb menghasilkan nilai *jitter* sebesar 0.050362 s, data audio sebesar 9.040 Kb menghasilkan nilai *jitter* sebesar 0.038864 s. Hasil yang didapat dari *variasi delay* atau *jitter* pada data audio dengan *bandwidth* 512 Kbps memiliki nilai yang tidak berbeda secara signifikan.

Tabel 4.13 hasil perbandingan jitter audio

Bandwidth 512Kbps		
Ukuran  Audio	Jitter (s)	
	MPLS	MPLS-TE
2.037 KB	0.087347	0.08144
6.883 KB	0.051923	0.050362
9.040 KB	0.043929	0.038864

Pada gambar 4.9 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



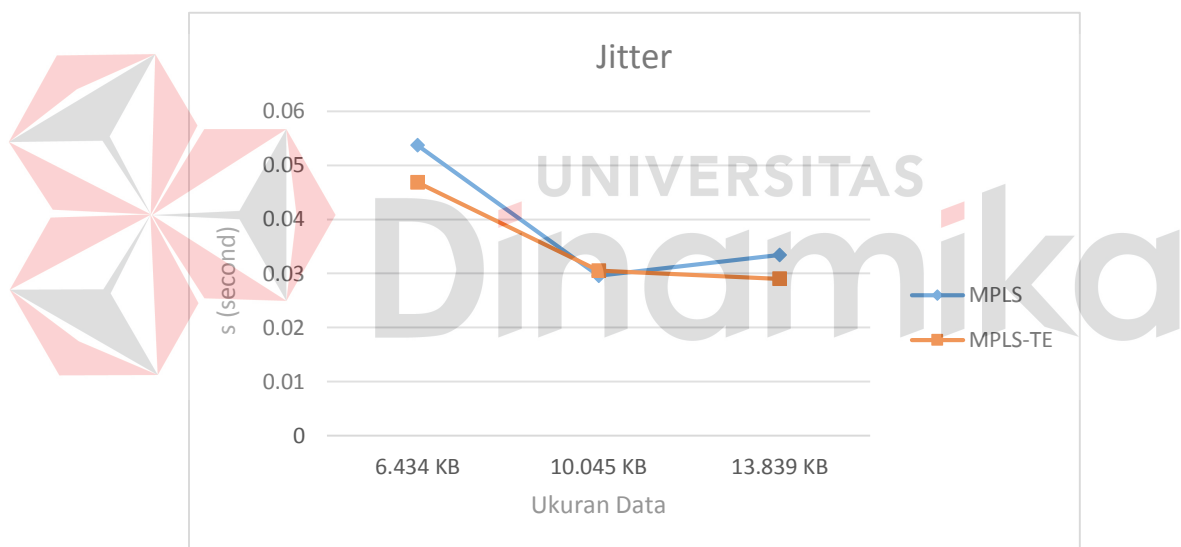
Gambar 4.9 Hasil grafik perbandingan jitter audio

Pada tabel 4.14 merupakan hasil *jitter* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 512 Kbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *jitter* 0.053685 s, data video sebesar 10.045 Kb menghasilkan nilai *jitter* 0.029567 s, data video sebesar 13.839 Kb menghasilkan nilai *jitter* 0.033423 s dan pada teknologi MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai *jitter* 0.046823 s, data video sebesar 10.045 Kb menghasilkan nilai *jitter* 0.030515 s, data video sebesar 13.895 Kb menghasilkan nilai *jitter* 0.029017 s. Dari hasil *jitter* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.14 hasil perbandingan jitter video

Bandwidth 512Kbps		
Ukuran Video	Jitter (s)	
	MPLS	MPLS-TE
6.434 KB	0.053685	0.046823
10.045 KB	0.029567	0.030515
13.839 KB	0.033423	0.029017

Pada gambar 4.10 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE.



Gambar 4.10 Hasil grafik perbandingan jitter video

Pada teknologi MPLS dan MPLS-TE dengan data audio dan video dan menggunakan *bandwidth* yang digunakan sebesar 512 Kbps memiliki hasil yang tidak berbeda secara signifikan. Dengan data video sebesar 6.434 Kb dan 13.839 Kb mengalami kenaikan *jitter*.

Faktor-faktor yang mempengaruhi nilai *Jitter* diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan *jitter*. *Jitter* biasanya disebut

variasi *delay* yang berhubungan erat dengan *latency*, yang menunjukkan banyaknya variasi *delay* pada transmisi data dalam jaringan. Besarnya nilai *jitter* yang tidak berbeda secara signifikan, dikarenakan ada proses pelabelan pada paket yang datang kemudian akan dikirimkan ke router berikutnya. Dengan bandwidth sebesar 512 Kbps nilainya semakin kecil daripada nilai bandwidth 128 Kbps. Semakin besar *bandwidth* yang digunakan maka akan semakin kecil nilai *jitter*nya.

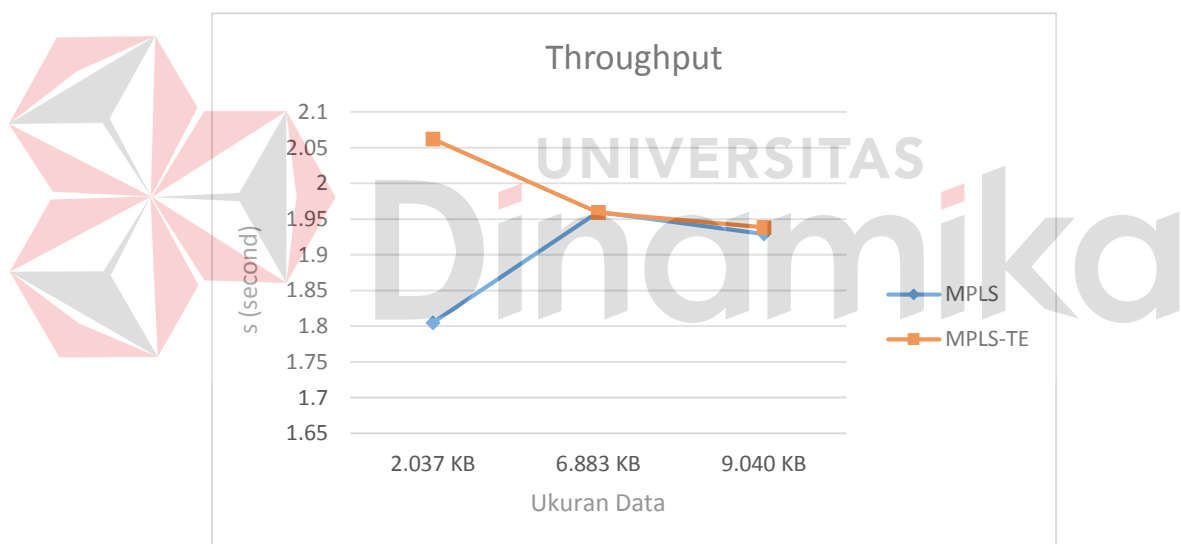
#### 4.2.7. Analisis Throughput bandwidth 512Kbps

*Throughput* merupakan jumlah total kedatangan paket yang diamati pada waktu interval tertentu. Digunakan untuk mengukur atau menghitung kecepatan data. Pada tabel 4.15 merupakan hasil *throughput* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 512 Kbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *throughput* sebesar 1.804683, data audio sebesar 6.883 Kb menghasilkan nilai *throughput* sebesar 1.959849, data audio sebesar 9.040 Kb menghasilkan nilai *throughput* sebesar 1.92907 dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *throughput* 2.061903, data audio sebesar 6.883 Kb menghasilkan nilai *throughput* sebesar 1.959074, data audio sebesar 9.040 Kb menghasilkan nilai *throughput* sebesar 1.938261. Hasil yang didapat dari *throughput* pada data audio dengan *bandwidth* 512 Kbps memiliki nilai yang tidak berbeda secara signifikan.

Tabel 4.15 hasil perbandingan throughput audio

Bandwidth 512Kbps		
Ukuran Audio	Throughput (bps)	
	MPLS	MPLS-TE
2.037 KB	1.804683	2.061903
6.883 KB	1.959849	1.959074
9.040 KB	1.92907	1.938261

Pada gambar 4.11 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.11 Hasil grafik perbandingan throughput audio

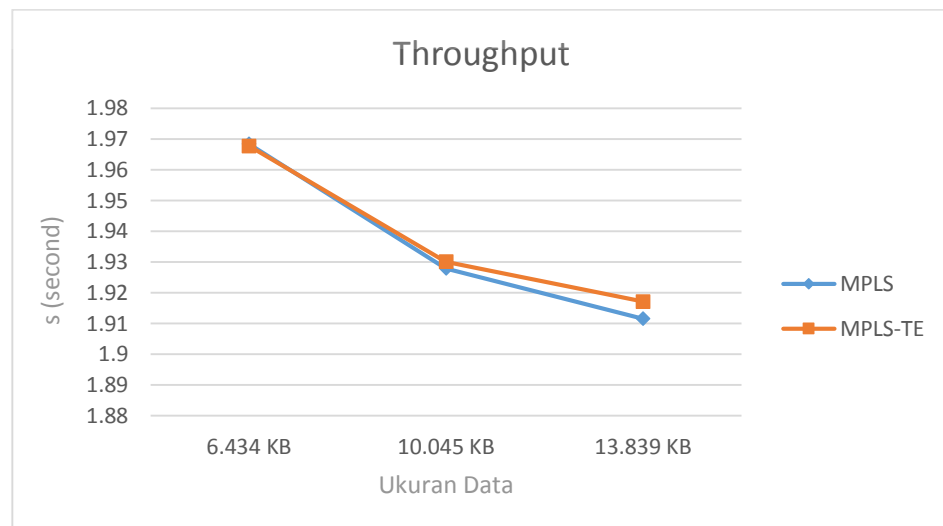
Pada tabel 4.16 merupakan hasil *throughput* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 512 Kbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *throughput* 1.968388, data video sebesar 10.045 Kb menghasilkan nilai *throughput* 1.927849, data video sebesar 13.839 Kb menghasilkan nilai *throughput* 1.911504 dan pada teknologi

MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai 1.96773, data video sebesar 10.045 Kb menghasilkan nilai *throughput* 1.930079, data video sebesar 13.895 Kb menghasilkan nilai *throughput* 1.917093. Dari hasil *throughput* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.16 hasil perbandingan throughput video

Bandwidth 512Kbps		
Ukuran Video	Throughput (bps)	
	MPLS	MPLS-TE
6.434 KB	1.968388	1.96773
10.045 KB	1.927849	1.930079
13.839 KB	1.911504	1.917093

Pada gambar 4.12 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.12 Hasil grafik perbandingan throughput video

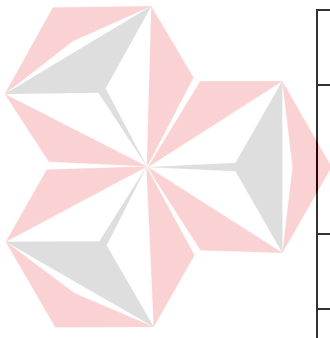
Pada analisis *throughput* diatas dari data audio dan video dengan bandwidth yang digunakan sebesar 512 kbps memperlihatkan MPLS-TE dan MPLS tidak

memiliki perbedaan yang cukup signifikan. Hal ini disebabkan karena pada MPLS-TE mempunyai mekanisme *tunnel* atau jalur alternative yang digunakan untuk mengatasi kemacetan dan menggunakan ukuran bandwidth yang sama dengan teknologi MPLS, sehingga hasil *throughput* tidak berbeda secara signifikan.

#### 4.2.8. Analisis Packet Loss bandwidth 512Kbps

Packet loss merupakan paket yang hilang disebabkan beberapa factor. Hasil untuk *bandwidth* 1Mbps pada tabel 4.17 dan tabel 4.18 tidak ada data yang hilang, maka kedua data tersebut memiliki nilai *packet loss* 0 %.

Tabel 4.17 hasil perbandingan packet loss audio



Bandwidth 512Kbps		
Ukuran Video	Packet Loss (%)	
	MPLS	MPLS-TE
2.037 KB	0	0
6.883 KB	0	0
9.040 KB	0	0

Tabel 4.18 hasil perbandingan packet loss video

Bandwidth 512Kbps		
Ukuran Video	Packet Loss (%)	
	MPLS	MPLS-TE
6.434 KB	0	0
10.045 KB	0	0
13.839 KB	0	0

Dari analisis *packet loss* diatas menunjukkan bahwa data audio dan video dengan bandwidth yang digunakan 512 Kbps memiliki nilai *packet loss* yang sama yaitu 0%, dikarenakan pada saat pengiriman data tidak mengalami gangguan pada jalur yang digunakan.

#### 4.2.9. Analisis latency bandwidth 1 Mbps

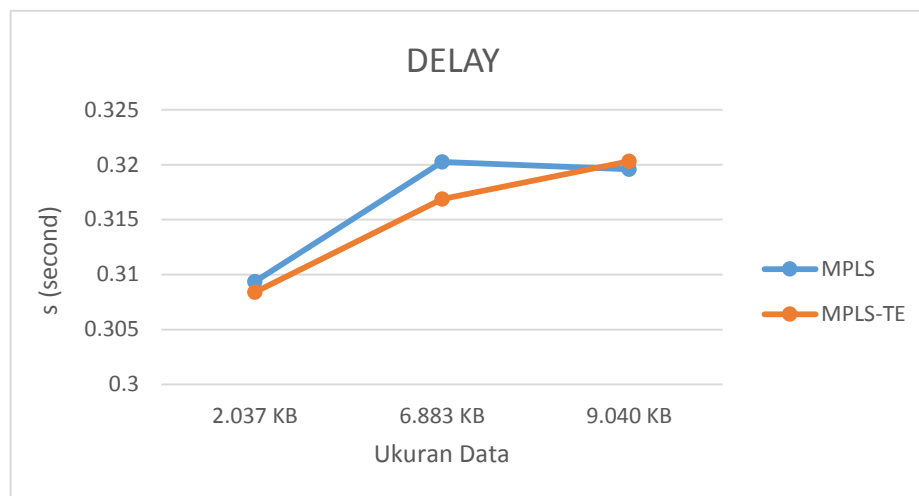
Pada tabel 4.19 merupakan hasil *latency* perbandingan antara MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883Kb dan 9.040 Kb. Pada tabel 4.3 memperlihatkan hasil *latency* yang didapat pada perbandingan MPLS dan MPLS-TE. Pada teknologi MPLS data audio sebesar 2.037 Kb sampai ke tujuan dengan waktu 0.309355 s, data audio sebesar 6.883 Kb sampai ke tujuan dengan waktu 0.320244 s, data audio sebesar 9.040 Kb sampai ke tujuan dengan waktu 0.319569 s dan teknologi MPLS-TE pada data audio sebesar 2.037 Kb sampai ke tujuan dengan waktu 0.3084 s, data audio sebesar 6.883 Kb sampai ke tujuan dengan waktu 0.316877 s, data audio sebesar 9.040 Kb sampai ke tujuan dengan waktu 0.320307 s. Dengan data diatas menghasilkan nilai *latency* yang tidak berbeda secara signifikan.

Tabel 4.19 hasil perbandingan latency audio

Bandwidth 1Mbps		
Ukuran Audio	Latency ( s )	
	MPLS	MPLS-TE
<b>2.037 KB</b>	0.309355	0.3084
<b>6.883 KB</b>	0.320244	0.316877
<b>9.040 KB</b>	0.319569	0.320307



Pada gambar 4.13 hasil grafik dari perbandingan MPLS dan MPLS-TE



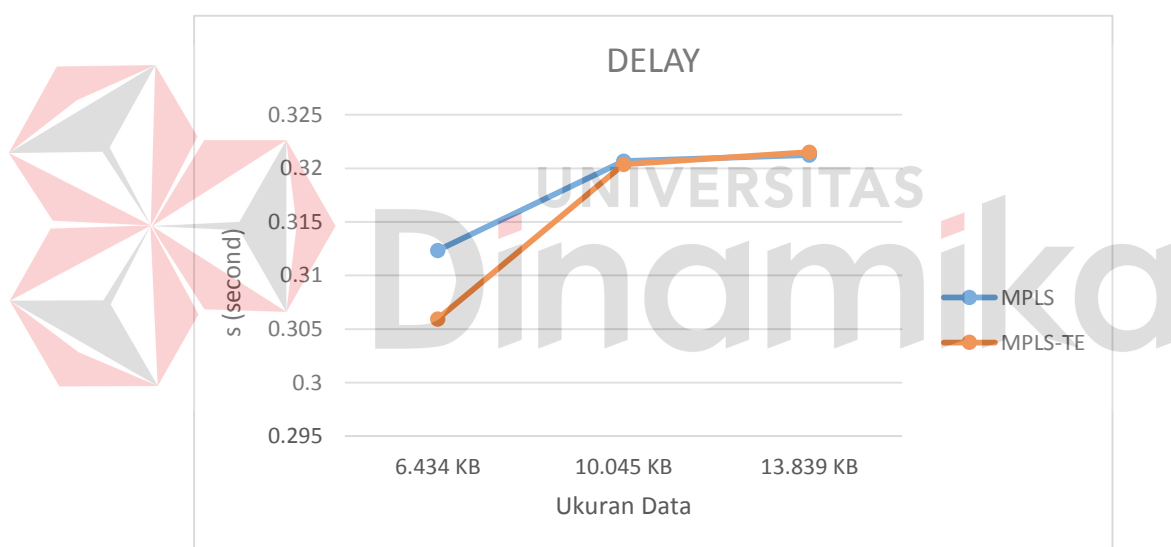
Gambar 4.13 Hasil grafik perbandingan delay audio

Pada tabel 4.20 merupakan hasil *latency* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 1 Mbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *latency* 0.31233 s, data video sebesar 10.045 Kb menghasilkan nilai *latency* 0.320673 s, data video sebesar 13.839 Kb menghasilkan nilai *latency* 0.321253 s dan pada teknologi MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai 0.305943 s, data video sebesar 10.045 Kb menghasilkan nilai *latency* 0.320384 s, data video sebesar 13.895 Kb menghasilkan nilai *latency* 0.321531 s. Dari hasil *latency* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.20 hasil perbandingan delay video

Bandwidth 1Mbps		
Ukuran Video	Latency ( s )	
	MPLS	MPLS-TE
<b>6.434 KB</b>	0.31233	0.305943
<b>10.045 KB</b>	0.320673	0.320384
<b>13.839 KB</b>	0.321253	0.321531

Pada gambar 4.14 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.14 Hasil grafik perbandingan delay video

Dari data audio dan data video memiliki hasil yang tidak berbeda secara signifikan, dikarenakan ada proses pelabelan pada paket yang datang kemudian akan dikirimkan ke *router* berikutnya. Penyebab selanjutnya yaitu, karena pada kedua teknologi sama-sama menggunakan *bandwidth* sebesar 1 Mbps, sehingga mendapatkan hasil delay yang tidak berbeda secara signifikan.

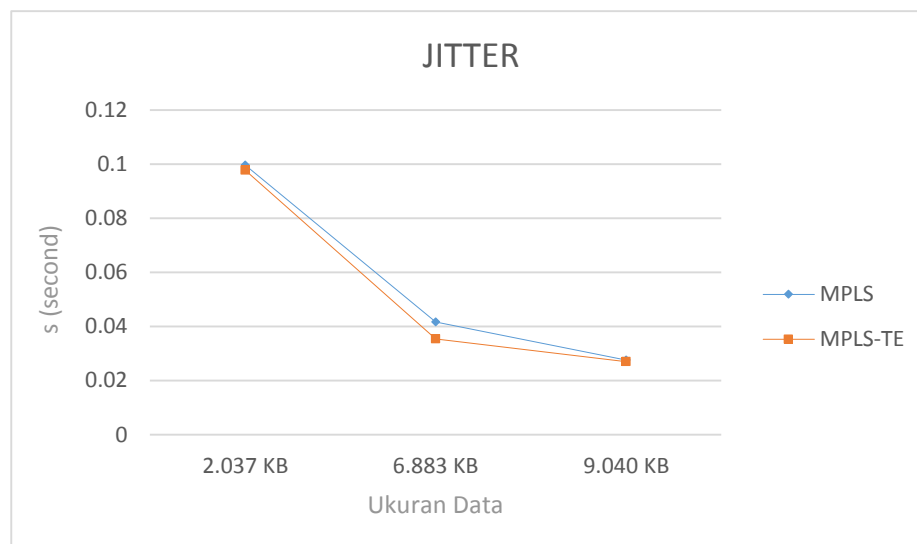
#### 4.2.10. Analisis Jitter bandwidth 1Mbps

*Jitter* atau variasi *delay* merupakan waktu kedatangan dari suatu paket ke penerima dengan waktu yang diharapkan. Digunakan untuk mengetahui waktu tunda paket yang diterima. Pada tabel 4.21 merupakan hasil *jitter* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 1 Mbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *jitter* sebesar 0.099624 s, data audio sebesar 6.883 Kb menghasilkan nilai *jitter* sebesar 0.04165 s, data audio sebesar 9.040 Kb menghasilkan nilai *jitter* sebesar 0.027565 s dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *jitter* 0.097772 s, data audio sebesar 6.883 Kb menghasilkan nilai *jitter* sebesar 0.035404 s, data audio sebesar 9.040 Kb menghasilkan nilai *jitter* sebesar 0.026975 s. Hasil yang didapat dari *variasi delay* atau *jitter* pada data audio dengan *bandwidth* 1 Mbps memiliki nilai yang tidak berbeda secara signifikan.

Tabel 4.21 Hasil perbandingan jitter audio

Bandwidth 1Mbps		
Ukuran Audio	Jitter ( s )	
	MPLS	MPLS-TE
2.037 KB	0.099624	0.097772
6.883 KB	0.04165	0.035404
9.040 KB	0.027565	0.026975

Pada gambar 4.15 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



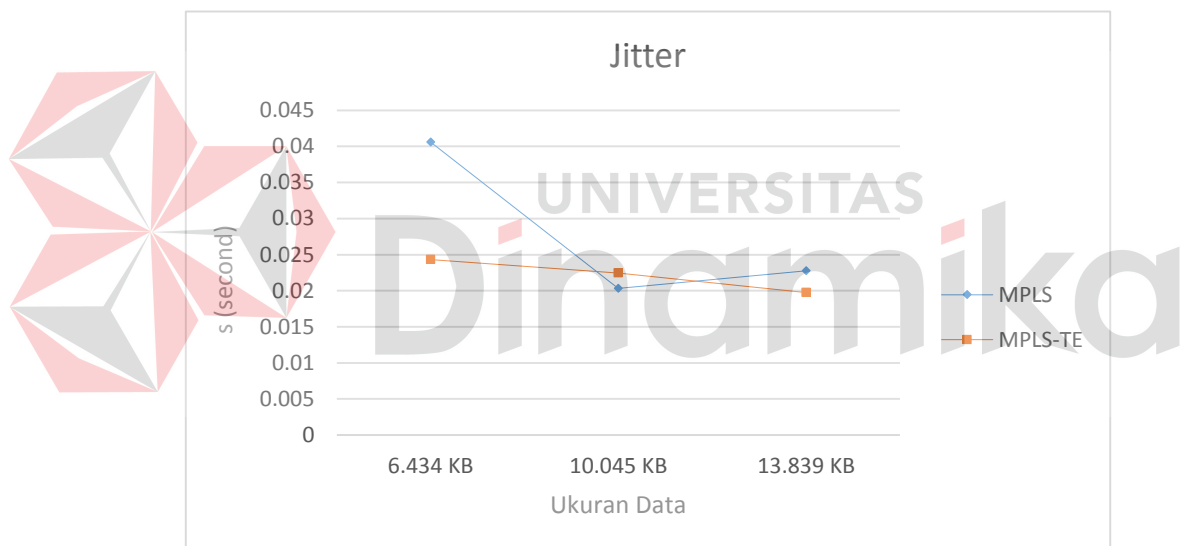
Gambar 4.15 Hasil grafik perbandingan jitter audio

Pada tabel 4.22 merupakan hasil *jitter* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 1 Mbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *jitter* 0.040576 s, data video sebesar 10.045 Kb menghasilkan nilai *jitter* 0.020331 s, data video sebesar 13.839 Kb menghasilkan nilai *jitter* 0.022777 s dan pada teknologi MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai *jitter* 0.024329 s, data video sebesar 10.045 Kb menghasilkan nilai *jitter* 0.022465 s, data video sebesar 13.895 Kb menghasilkan nilai *jitter* 0.01972423 s. Dari hasil *jitter* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.22 Hasil perbandingan jitter video

Bandwidth 1Mbps		
Ukuran Video	Jitter (s)	
	MPLS	MPLS-TE
<b>6.434 KB</b>	0.040576	0.024329
<b>10.045 KB</b>	0.020331	0.022465
<b>13.839 KB</b>	0.022777	0.01972423

Pada gambar 4.16 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.16 Hasil grafik perbandingan jitter video

Pada teknologi MPLS dan MPLS-TE dengan data audio dan video dan menggunakan *bandwidth* yang digunakan sebesar 1 Mbps memiliki hasil yang tidak berbeda secara signifikan. Dengan data video sebesar 6.434 Kb mengalami kenaikan *jitter*.

Faktor-faktor yang mempengaruhi nilai *Jitter* diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan *jitter*. *Jitter* biasanya

disebut variasi *delay* yang berhubungan erat dengan *latency*, yang menunjukkan banyaknya variasi *delay* pada transmisi data dalam jaringan. Besarnya nilai *jitter* yang tidak berbeda secara signifikan, dikarenakan ada proses pelabelan pada paket yang datang kemudian akan dikirimkan ke *router* berikutnya. Dengan *bandwidth* sebesar 1 Mbps nilai *jitter* akan semakin kecil daripada nilai *bandwidth* 512 Kbps dan 128 Kbps. Semakin besar *bandwidth* yang digunakan maka akan semakin kecil nilai *jitter*nya.

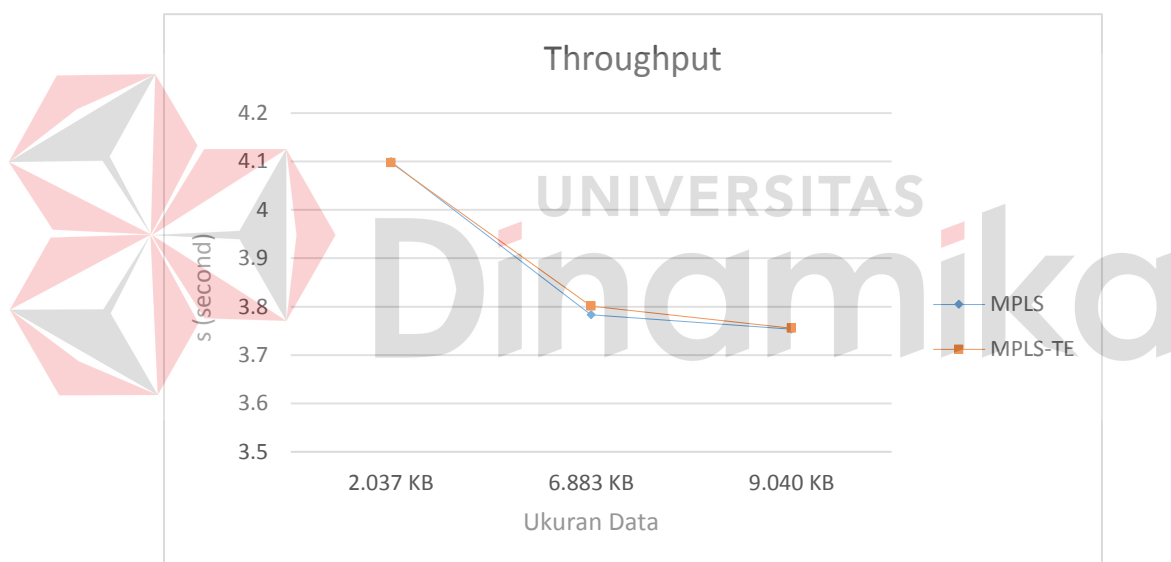
#### 4.2.11. Analisis Throughput bandwidth 1Mbps

*Throughput* merupakan jumlah total kedatangan paket yang diamati pada waktu interval tertentu. Digunakan untuk mengukur atau menghitung kecepatan data. Pada tabel 4.23 merupakan hasil *throughput* dari perbandingan teknologi MPLS dan MPLS-TE yang mempunyai ukuran data audio sebesar 2.037 Kb, 6.883 Kb dan 9.040 Kb. Pada *bandwidth* 1 Mbps dengan teknologi MPLS data audio yang dikirimkan sebesar 2.037 Kb menghasilkan nilai *throughput* sebesar 4.099163, data audio sebesar 6.883 Kb menghasilkan nilai *throughput* sebesar 3.783199, data audio sebesar 9.040 Kb menghasilkan nilai *throughput* sebesar 3.75329 dan pada teknologi MPLS-TE dengan data audio sebesar 2.037 Kb menghasilkan nilai *throughput* 4.097584, data audio sebesar 6.883 Kb menghasilkan nilai *throughput* sebesar 3.801126, data audio sebesar 9.040 Kb menghasilkan nilai *throughput* sebesar 3.755522. Hasil yang didapat dari *throughput* pada data audio dengan *bandwidth* 1 Mbps memiliki nilai yang tidak berbeda secara signifikan.

Tabel 4.23 hasil perbandingan throughput audio

Bandwidth 1Mbps		
Ukuran Audio	Throughput ( bps )	
	MPLS	MPLS-TE
<b>2.037 KB</b>	4.099163	4.097584
<b>6.883 KB</b>	3.783199	3.801126
<b>9.040 KB</b>	3.75329	3.755522

Pada gambar 4.17 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE.



Gambar 4.17 Hasil grafik perbandingan throughput audio

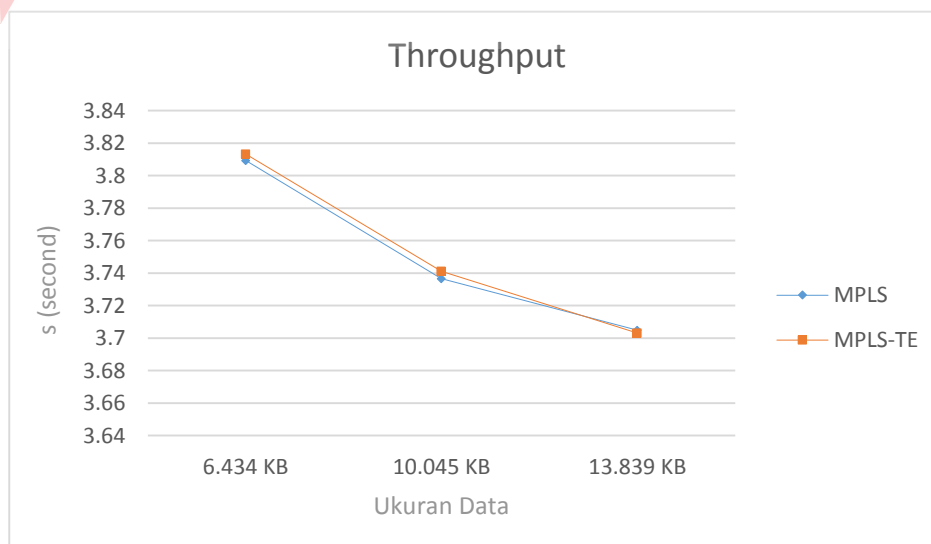
Pada tabel 4.24 merupakan hasil *throughput* dari perbandingan MPLS dan MPLS-TE yang mempunyai ukuran data video sebesar 6.434 Kb, 10.045 Kb dan 13.839 Kb. Pada Teknologi MPLS dengan *bandwidth* 1 Mbps pada data video dengan ukuran sebesar 6.434 Kb menghasilkan nilai *throughput* 3.809274, data video sebesar 10.045 Kb menghasilkan nilai *throughput* 3.736541, data video sebesar 13.839 Kb menghasilkan nilai *throughput* 3.704814 dan pada teknologi

MPLS-TE dengan data video ukuran sebesar 6.434 Kb menghasilkan nilai 3.813336, data video sebesar 10.045 Kb menghasilkan nilai *throughput* 3.741136, data video sebesar 13.895 Kb menghasilkan nilai *throughput* 3.703128. Dari hasil *throughput* yang didapat pada data video memiliki hasil yang tidak berbeda secara signifikan.

Tabel 4.24 hasil perbandingan throughput video

Bandwidth 1Mbps		
Ukuran Video	Throughput (bps)	
	MPLS	MPLS-TE
<b>6.434 KB</b>	3.809274	3.813336
<b>10.045 KB</b>	3.736541	3.741136
<b>13.839 KB</b>	3.704814	3.703128

Pada gambar 4.18 merupakan hasil grafik dari perbandingan antara MPLS dan MPLS-TE



Gambar 4.18 Hasil grafik perbandingan throughput video

Dari analisis *throughput* diatas menunjukkan bahwa data audio dan video dengan *bandwidth* yang digunakan sebesar 1 Mbps memperlihatkan MPLS-TE



dan MPLS tidak memiliki perbedaan yang cukup signifikan. Menggunakan *bandwidth* sebesar 1 Mbps *throughput* yang dihasilkan lebih bagus dari pada *bandwidth* 128 Kbps dan 512 Kbps. Hal ini disebabkan karena pada MPLS-TE mempunyai mekanisme *tunnel* atau jalur alternative yang digunakan untuk mengatasi kemacetan dan menggunakan ukuran *bandwidth* yang sama dengan teknologi MPLS, sehingga hasil *throughput* tidak berbeda secara signifikan.

#### 4.2.12. Analisis Packet Loss bandwidth 1Mbps

*Packet loss* merupakan paket yang hilang disebabkan beberapa factor. Hasil untuk *bandwidth* 1Mbps pada tabel 4.25 dan tabel 4.26 tidak ada data yang hilang, maka kedua data tersebut memiliki nilai *packet loss* 0%.

Tabel 4.25 hasil perbandingan packet loss audio

Bandwidth 1Mbps		
Ukuran Audio	Packet Loss (%)	
	MPLS	MPLS-TE
2.037 KB	0	0
6.883 KB	0	0
9.040 KB	0	0

Tabel 4.26 hasil perbandingan packet loss video

Bandwidth 1Mbps		
Ukuran Video	Packet Loss (%)	
	MPLS	MPLS-TE
6.434 KB	0	0
10.045 KB	0	0
13.839 KB	0	0

Dari analisis *packet loss* diatas menunjukkan bahwa data audio dan video dengan bandwidth yang digunakan sebesar 1 Mbps memiliki nilai *packet loss* yang sama yaitu 0%, dikarenakan pada saat pengiriman data tidak mengalami gangguan pada jalur yang digunakan.

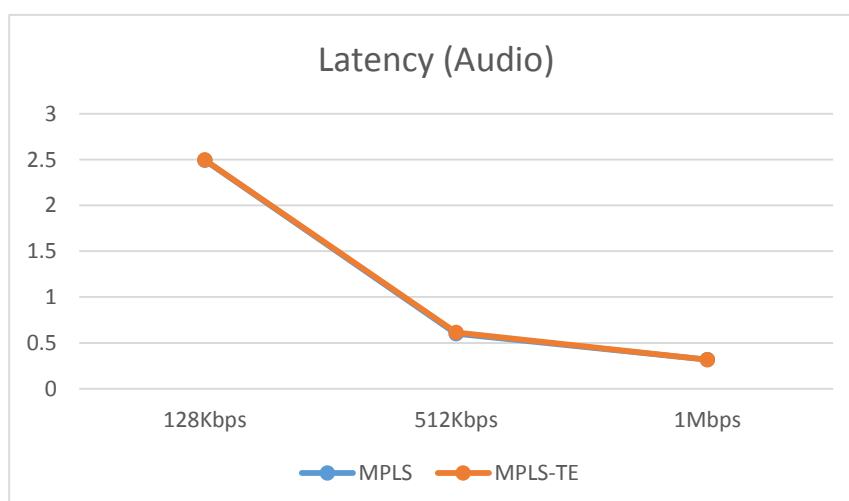
#### 4.2.13. Hasil analisis dari Qos dengan bandwidth yang berbeda

Hasil rata-rata dari Qos, antara lain *Latency (delay)*, *Jitter*, *Throughput* dan *Packet loss* dengan membandingkan dengan *bandwidth* yang berbeda – beda.

1. *Delay* pada audio dan video dengan perbandingan *bandwidth* yang berbeda.

Tabel 4.27 Hasil dari keseluruhan bandwidth delay pada audio

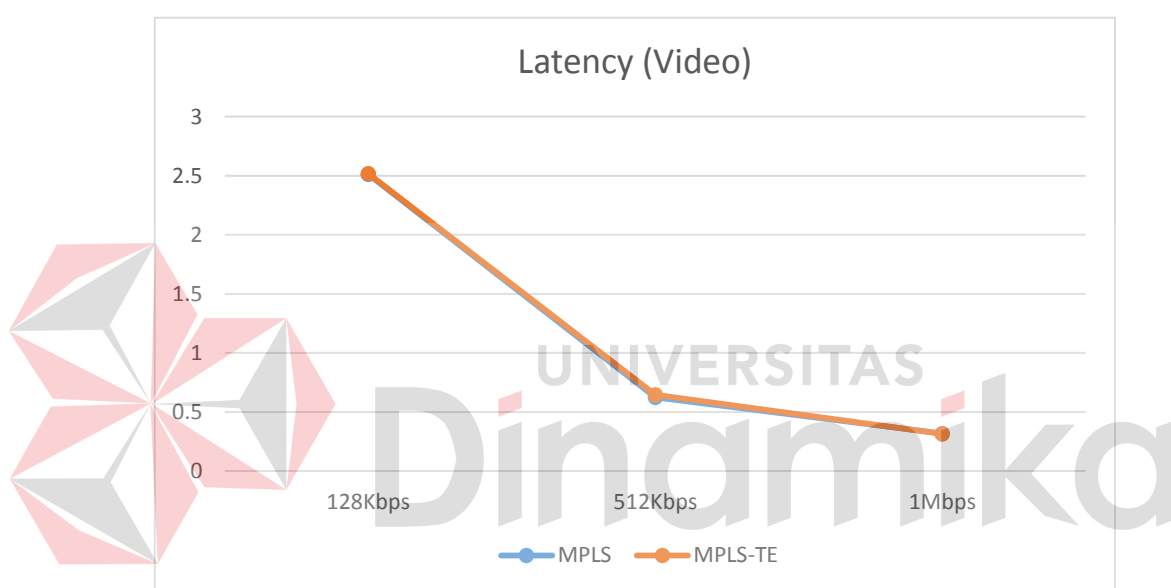
Ukuran Bandwidth	MPLS	MPLS-TE
128Kbps	2.492844	2.495931
512Kbps	0.599947	0.614832
1Mbps	0.316389	0.315195



Gambar 4.19 Grafik hasil dari keseluruhan bandwidth delay pada audio

Tabel 4.28 Hasil dari keseluruhan bandwidth delay pada video

Ukuran Bandwidth	MPLS	MPLS-TE
<b>128Kbps</b>	2.512818	2.518474
<b>512Kbps</b>	0.625622	0.646229
<b>1Mbps</b>	0.318073	0.315953



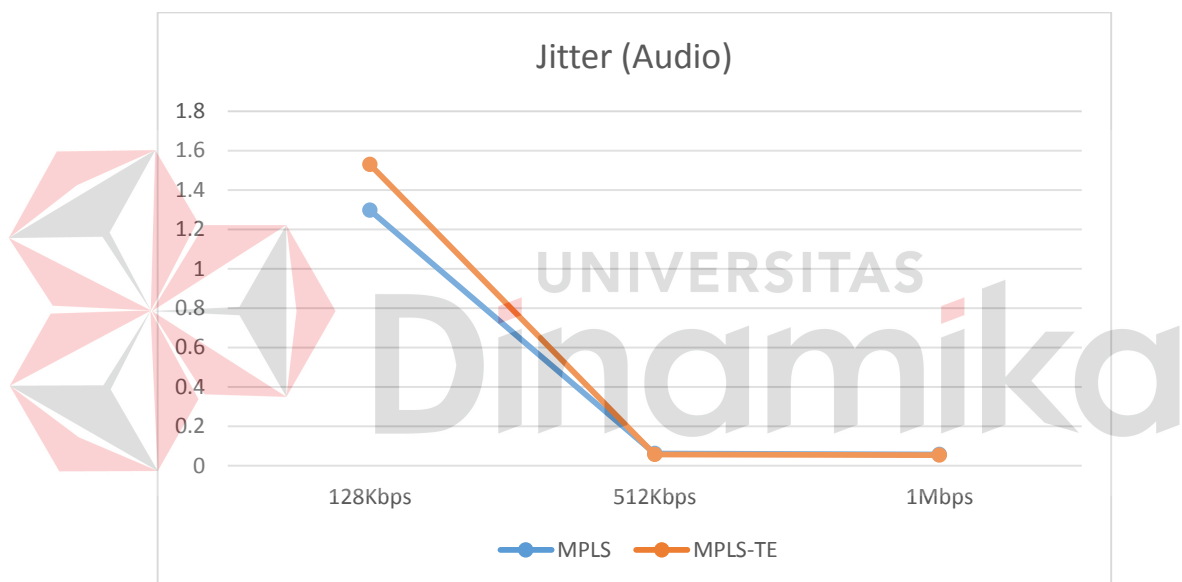
Gambar 4.20 Grafik hasil dari keseluruhan bandwidth delay pada video

Dari analisis *latency* diatas menunjukkan bahwa data audio dan data video dengan bandwidth yang digunakan 128 Kbps, 512 Kbps dan 1 Mbps tidak memiliki perbedaan secara signifikan. Hal ini disebabkan karena pada penelitian ini, simulasi jaringan yang tidak diimplementasikan pada jaringan public (karena adanya kendala mendapatkan IP *Public*). Selain itu simulasi jaringan memiliki jumlah hop yang sedikit, sehingga hasil perbandingan delay antara MPLS dan MPLS-TE tidak berbeda secara signifikan.

2. *Jitter* pada audio dan video dengan perbandingan *bandwidth* yang berbeda.

Tabel 4.29 Hasil dari keseluruhan bandwidth jitter pada audio

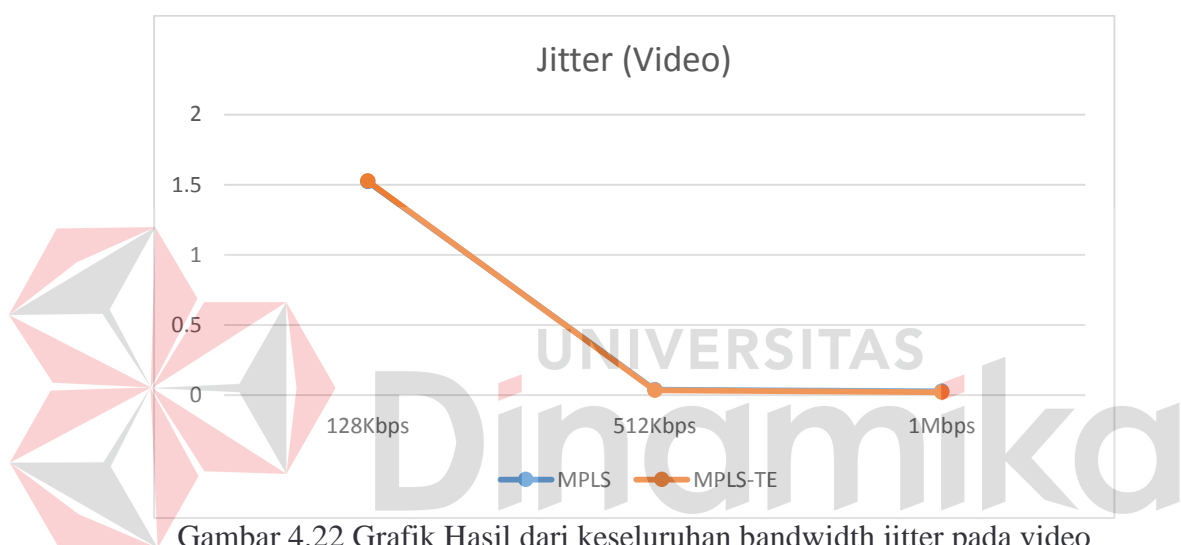
Ukuran Bandwidth	MPLS	MPLS-TE
128Kbps	1.296224	1.528782
512Kbps	0.061066	0.056889
1Mbps	0.056278	0.053384



Gambar 4.21 Grafik hasil dari keseluruhan bandwidth jitter pada audio

Tabel 4.30 Hasil dari keseluruhan bandwidth jitter pada video

Ukuran Bandwidth	MPLS	MPLS-TE
<b>128Kbps</b>	1.523415	1.526177
<b>512Kbps</b>	0.038892	0.035452
<b>1Mbps</b>	0.027895	0.022173



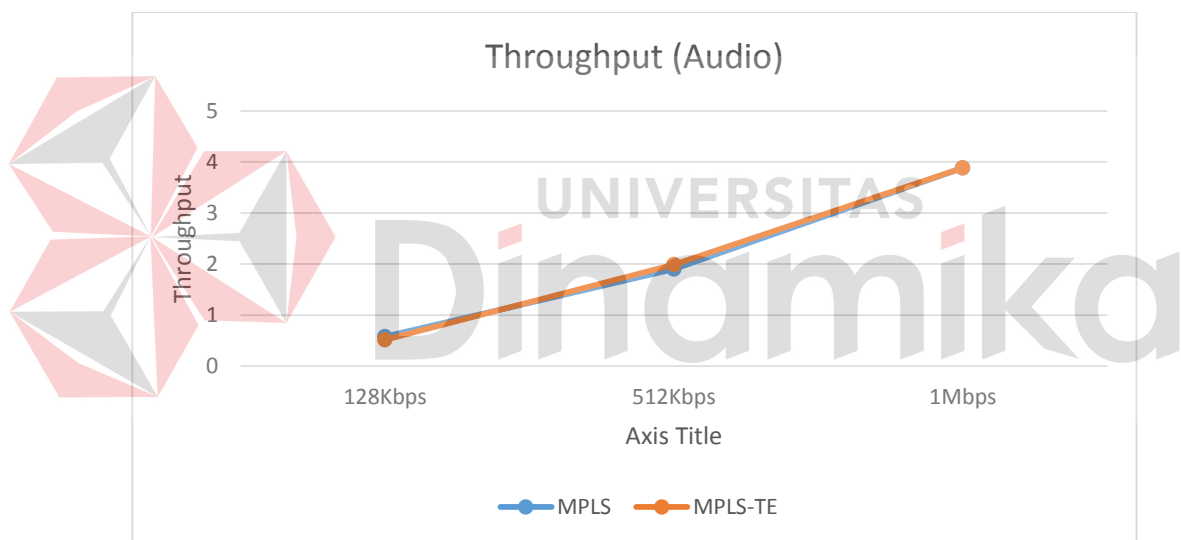
Gambar 4.22 Grafik Hasil dari keseluruhan bandwidth jitter pada video

Pada jitter audio bandwidth 128Kbps pada MPLS mempunyai nilai 1.296224 dan MPLS-TE mempunyai nilai 1.528782 kedua teknologi tersebut tidak mempunyai perbedaan secara signifikan. Data video pada delay bandwidth 1Mbps memiliki nilai yang dihasilkan oleh MPLS yaitu sebesar 0.027895 dan MPLS-TE memiliki nilai jitter sebesar 0.022173 hasil tersebut tidak berbeda secara signifikan. Karena pada penelitian ini tidak diimplementasikan pada jaringan public dan memiliki jumlah hop yang sedikit, sehingga hasil jitter tidak berbeda secara signifikan.

3. *Throughput* pada audio dan video dengan perbandingan *bandwidth* yang berbeda.

Tabel 4.31 Hasil dari keseluruhan bandwidth throughput pada audio

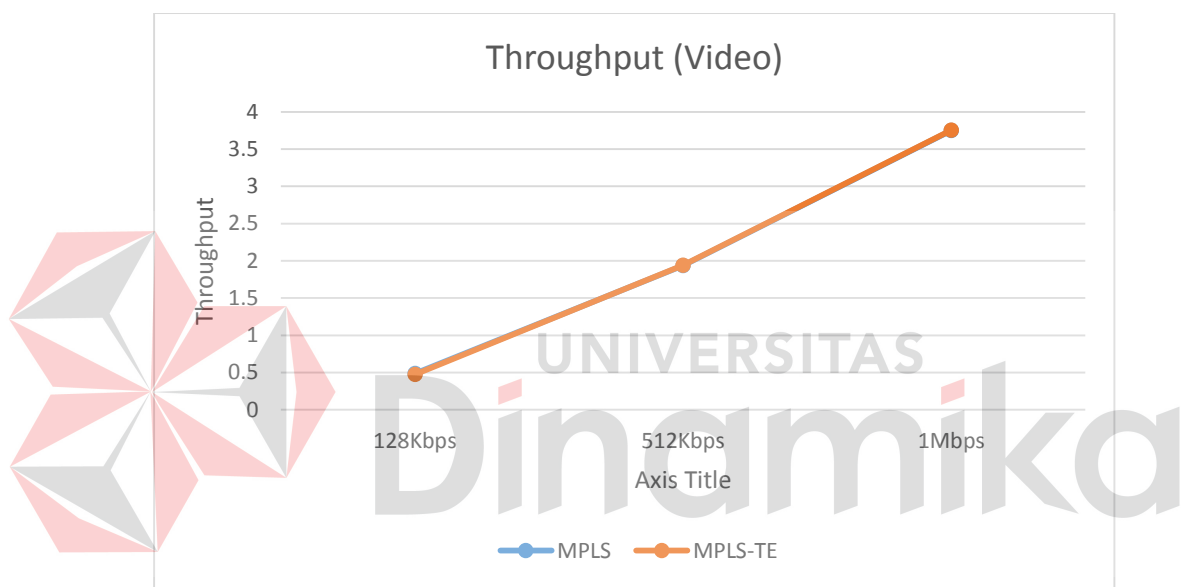
Ukuran Bandwidth	MPLS	MPLS-TE
<b>128Kbps</b>	0.57972	0.514591
<b>512Kbps</b>	1.897884	1.986404
<b>1Mbps</b>	3.878551	3.884744



Gambar 4.23 Grafik hasil dari keseluruhan bandwidth throughgput pada audio

Tabel 4.32 Hasil dari keseluruhan bandwidth throughput pada video

Ukuran Bandwidth	MPLS	MPLS-TE
<b>128Kbps</b>	0.486972	0.474078
<b>512Kbps</b>	1.935914	1.938301
<b>1Mbps</b>	3.75021	3.752533



Gambar 4.24 Grafik hasil dari keseluruhan bandwidth throughput pada video

Dari analisis *throughput* diatas menunjukkan bahwa data audio dengan *bandwidth* 128Kbps pada teknologi MPLS mempunyai nilai 0.57972 dan teknologi MPLS-TE mempunyai nilai 0.514591, kedua teknologi tersebut tidak mempunyai perbedaan secara signifikan. Data video pada delay bandwidth 128Kbps memiliki nilai yang dihasilkan oleh MPLS yaitu sebesar 0.486972 dan MPLS-TE memiliki nilai throughput sebesar 0.474078 hasil tersebut tidak berbeda secara signifikan. Hal ini disebabkan karena pada penelitian ini, simulasi jaringan yang tidak diimplementasikan pada jaringan public (karena adanya kendala mendapatkan IP *Public*). Selain itu simulasi jaringan memiliki jumlah

hop yang sedikit, sehingga hasil perbandingan throughput antara MPLS dan MPLS-TE tidak berbeda secara signifikan.

4. Packet Loss pada audio dan video dengan perbandingan bandwidth yang berbeda

Tabel 4.33 Hasil dari keseluruhan bandwidth packet loss pada audio

Ukuran Bandwidth	MPLS	MPLS-TE
<b>128 Kbps</b>	0	0
<b>512Kbps</b>	0	0
<b>1 Mbps</b>	0	0

Tabel 4.34 Hasil dari keseluruhan bandwidth packet loss pada video

Ukuran Bandwidth	MPLS	MPLS-TE
<b>128 Kbps</b>	0	0
<b>512Kbps</b>	0	0
<b>1 Mbps</b>	0	0

Dari analisis *packet loss* diatas, kedua data (audio dan video) dengan bandwidth yang digunakan 128 Kbps, 512 Kbps, 1 Mbps memiliki nilai *packet loss* yang sama yaitu 0%, dikarenakan pada saat pengiriman data tidak mengalami gangguan pada jalur yang digunakan.



## BAB V

### PENUTUP

#### 5.1. Kesimpulan

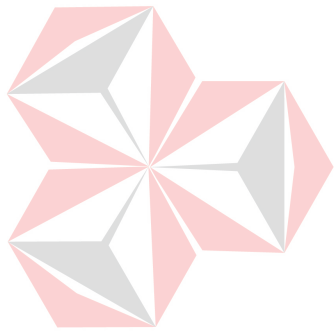
Berdasarkan hasil pengujian yang dilakukan pada implementasi dan analisis MPLS-TE pada jaringan berbasis mikrotik didapatkan beberapa poin kesimpulan sebagai berikut :

1. Nilai *latency (delay)*, *jitter*, *throughput* dan *packet loss* pada *bandwidth 128 Kbps* dengan ukuran data audio sebesar 2.037Kb, 6.883Kb, 9.040Kb dan data video dengan ukuran sebesar 6.434Kb, 10.045Kb, 13.839Kb memiliki nilai yang tidak berbeda secara signifikan.
2. Nilai *latency (delay)*, *jitter*, *throughput* dan *packet loss* pada *bandwidth 512 Kbps* dengan ukuran data audio sebesar 2.037Kb, 6.883Kb, 9.040Kb dan data video dengan ukuran sebesar 6.434Kb, 10.045Kb, 13.839Kb memiliki nilai yang tidak berbeda secara signifikan.
3. Nilai *latency (delay)*, *jitter*, *throughput* dan *packet loss* pada *bandwidth 1 Mbps* dengan ukuran data audio sebesar 2.037Kb, 6.883Kb, 9.040Kb dan data video dengan ukuran sebesar 6.434Kb, 10.045Kb, 13.839Kb memiliki nilai yang tidak berbeda secara signifikan.
4. Berdasarkan pengujian parameter uji kinerja QoS, antara lain *latency (delay)*, *jitter*, *throughput* dan *packet loss* dengan *bandwidth* yang bervariasi menunjukkan bahwa unjuk kerja MPLS dibandingkan dengan MPLS-TE nilainya tidak berbeda secara signifikan.

## 5.2. Saran

Berikut ini terdapat beberapa saran yang penulis berikan untuk penelitian berikutnya apabila ingin mengembangkan penelitian yang telah dibuat agar menjadi lebih baik adalah sebagai berikut :

1. Penelitian berikutnya diharapkan mengimplementasikan pada jaringan public.
2. Memperbanyak jumlah perangkat *router* maupun *client*.



UNIVERSITAS  
**Dinamika**

## DAFTAR PUSTAKA

- Aloriadi, R. (2011). Analisis pemodelan sistem jaringan MPLS-TE pada layanan multimedia berbasis IP Multimedia Subsystem (IMS). *telkomuniversity*, 7.
- Auliya, R. S. (2013). Penerapan Multiprotocol Label Switching (MPLS) untuk Mengatasi Permasalahan pada Best-effort Service. *scribd*, 3.
- Cahyaningtyas, A. (2013). *Pengenalan Dan Dasar Penggunaan Wireshark*. Diakses Februari 2015 dari Ilmu Komputer: <http://ilmukomputer.org/2013/04/22/pengenalan-dan-dasar-penggunaan-wireshark/>
- Dewannanta, D. (2003). *Mendesain Jaringan dengan Multi Protocol Label Switching (MPLS)*. Diakses Februari 2015 dari Ilmu Komputer: <http://ilmukomputer.org/wp-content/uploads/2013/02/mppls.pdf>
- Ghein, L. D. (2006). *MPLS Fundamental*. Cisco Press.
- Kristianta, H., & Setiawan, B. (2014). Traffic Engineering Analysis Menggunakan LSP (Label-Switched-Path) Pada Jaringan Metro Ethernet Alcatel Lucent. *palcomtech*.
- Kumar, U. (2010). MPLS - Control Plane and Data Plane. *Start Networks*, 2. Diakses Februari 2015 dari Networking: <http://www.startnetworks.info/2010/08/mppls-control-plane-and-data-plane.html>
- Ozan. (2012). *FTP Server dan FTP Client*. Diakses Februari 2015 dari <https://melengo.wordpress.com/tag/ftp-server/>
- Permadi, R. A. (2009). Implementasi Differentiated Services pada Jaringan Multiprotocol Label Switching untuk Rural Next Generation Network. *Institut Teknologi Bandung*.
- Rahman, M. (2013). *MPLS (Multiprotocol Layer Switching)*. Computer Network. Diakses Februari 2015 dari Miftah Rahman (Go)-Blog: <https://belajarcomputernetwork.wordpress.com/tag/mppls-te/>
- Rahmawati, I. D. (2011). Analisis QoS pada jaringan MPLS ipv6 berbasis routing ospf. *eeepis-its*, 6.
- Rijayana, I. (2005). Teknologi Multi Protocol Label Switching (MPLS) Untuk Meningkatkan Performa Jaringan. *Seminar Nasional Aplikasi Teknologi Informasi 2005*, 2.

Setiawan, W. (2014). *Pengertian Dan Penjelasan IPv4 atau Alamat IP (Internet Protocol) Versi 4*. Diakses Februari 2015 dari Blangkonsoft.

Simatupang, S. A. (2011). Analisis dan Implementasi Jaringan Backbone MPLS-TE pada Layanan VoIP. *telkomuniversity*.

Sinaga, I. (2013). *Pengenalan Mikrotik*. Netkrom. Diakses Februari 2015 dari Tutorial Netkrom: <http://tutorial.netkromsolution.com/?p=899>

Sugeng, W. (2010). *Jaringan Komputer dengan TCP/IP*. Bandung: Modula.

Towidjojo, R. (2012). *Konsep dan Implementasi Routing dengan Router Mikrotik*. Jasakom.

Wastuwibowo, K. (2003). *Jaringan MPLS*. Diakses Februari 2015 dari Academia: [http://www.academia.edu/2585297/Jaringan\\_MPLS](http://www.academia.edu/2585297/Jaringan_MPLS)



UNIVERSITAS  
**Dinamika**