

Implementasi Load Balancing Menggunakan Metode Equal Cost Multi Path (ECMP) Pada Interkoneksi Jaringan

Muhammad Khoirul Anwar, Ida Nurhaida

Teknik Informatika, Universitas Mercu Buana, Jakarta
41515010058@student.mercubuana.ac.id¹, ida.nurhaida@mercubuana.ac.id²

Abstrak

Internet merupakan kebutuhan yang penting digunakan dalam sebuah perusahaan. Koneksi internet yang digunakan dalam perusahaan harus stabil dan terhindar dari segala macam gangguan. Sebagian besar perusahaan sudah menggunakan dua *link* koneksi internet yang berguna untuk kelangsungan bisnis perusahaan. Penggunaan dua *link* koneksi tersebut tidak diimbangi dengan penerapan teknik *load balancing* dan *failover*. Pada penelitian ini masalah yang diangkat adalah penerapan *load balancing* yang sesuai dengan kebutuhan untuk mengoptimalkan kedua *link* koneksi internet. Metode *load balancing* yang digunakan adalah *Equal Cost Multi Path (ECMP)*. Selain digunakan untuk membagi beban trafik pada kedua *link*, *load balancing* dengan metode *ECMP* juga dapat memberikan efek *failover*. Dalam penelitian ini implementasi diterapkan pada dua divisi yaitu divisi *Customer Support Engineer (CSE)* dan *Customer Accounting Executive (CAE)* yang terdapat di PT. Cyberindo Aditama. Hasil pengujian menunjukkan bahwa *load balancing* dengan metode *ECMP* dapat menggunakan *link* koneksi internet secara bergantian sehingga dapat mendistribusikan beban trafik secara merata pada kedua *link* sedangkan efek *failover* dihasilkan jika salah satu *link* terputus maka beban trafik akan secara otomatis berpindah ke *link* koneksi lain yang masih aktif. Berdasarkan pengamatan yang dilakukan saat terjadi *load balancing* diperoleh bahwa divisi *CSE* memiliki *throughput* 99.96% (*sangat bagus*), *delay* 1.70 ms (*sangat bagus*), dan *jitter* 0.02 ms (*sangat bagus*) sedangkan divisi *CAE* memiliki *throughput* 99.92% (*sangat bagus*), *delay* 1.69 ms (*sangat bagus*), dan *jitter* 0.01 ms (*sangat bagus*).

Keywords: Internet, *load balancing*, *ECMP*, *failover*

DOI: 10.22441/incomtech.v9i1.5003

1. PENDAHULUAN

Saat ini perkembangan teknologi informasi dan komunikasi tumbuh dengan sangat pesat. Salah satu contoh dari teknologi informasi dan komunikasi adalah internet. Internet sebagai suatu sarana komunikasi yang sangat memudahkan bagi

setiap orang [1]. Internet juga digunakan untuk mencari dan menyebarluaskan informasi dengan cepat dan mudah. Oleh karena itu, internet sangat penting digunakan dalam kehidupan sehari-hari terutama dalam institusi dan perusahaan. Dengan semakin meningkatnya penggunaan internet maka dibutuhkan juga infrastruktur yang baik untuk menjaga kelancaran akses internet [2].

Suatu perusahaan membutuhkan koneksi internet yang stabil untuk menjaga kelancaran bisnis perusahaannya [3]. Banyak perusahaan saat ini yang menggunakan dua jalur atau *link* koneksi internet yang berbeda untuk menjaga agar proses bisnisnya dapat tetap berjalan dengan baik. Dua jalur koneksi tersebut dinamakan sebagai *primary link* dan *secondary link*. Terdapat beberapa perusahaan yang hanya menggunakan *primary link* saja sebagai koneksi utama sedangkan *secondary link* akan digunakan pada saat *primary link* mengalami gangguan [4]. Tetapi tidak sedikit perusahaan yang menggunakan kedua *link* koneksi internet secara bersamaan namun digunakan pada divisi yang berbeda seperti divisi A menggunakan *primary link* sedangkan divisi B menggunakan *secondary link*.

Masalah yang sering terjadi dari penggunaan internet adalah masalah koneksi yang tidak stabil. Meskipun suatu perusahaan sudah memanfaatkan penggunaan kedua *link* koneksi internet untuk dipakai bersamaan tetapi memunculkan permasalahan baru jika salah satu *link* mengalami gangguan. Gangguan tersebut dapat terjadi pada router atau *link* koneksi internet yang digunakan serta trafik pada *link* koneksi mengalami *overload*. Sehingga gangguan yang mungkin terjadi akan menghambat kinerja pada divisi yang menggunakan koneksi internet yang terganggu tersebut.

Failover merupakan kemampuan sistem yang dimiliki pada router untuk berpindah *gateway* secara otomatis [2]. *Load balancing* merupakan teknik untuk membagi beban trafik pada dua atau lebih *link* koneksi [5]. Pada Penelitian ini akan dilakukan penerapan *load balancing* dan *failover* dengan menggabungkan dua *link* koneksi internet guna untuk mengatasi segala permasalahan yang mungkin terjadi terkait gangguan pada koneksi internet.

Terdapat banyak metode *load balancing* salah satunya adalah *Equal Cost Multi Path (ECMP)*. *Load balancing* dengan metode *ECMP* ini dapat memberikan efek *failover* begitu salah satu *link* koneksi terputus maka router melakukan *check-gateway* dan menonaktifkan *gateway* tersebut kemudian menggunakan *gateway* yang masih aktif [6]. Dengan demikian dalam penelitian ini dilakukan implementasi *load balancing* menggunakan metode *ECMP* dengan menggabungkan dua *link* koneksi internet.

2. STUDI LITERATUR

Pada bagian ini penulis melakukan studi literatur guna untuk menambah teori pendukung dalam melakukan penelitian ini.

Saat ini akses internet semakin dibutuhkan dalam oleh masyarakat luas terutama bagi organisasi besar. Riset [7] permasalahan yang dihadapi adalah penggunaan akses internet hanya bergantung pada satu *Internet Service Provider (ISP)*, sehingga bila koneksi pada *ISP* tersebut bermasalah maka akses internet menjadi terganggu. Sehingga dibutuhkan penggunaan dua koneksi sebagai akses ke jaringan internet. Menurut [2] sering terjadinya kendala pada saat mengakses internet yang disebabkan oleh penuhnya trafik dikarenakan banyak pengguna sehingga akses internet mengalami gangguan. Diperlukan *ISP* cadangan untuk memindahkan jalur

akses secara otomatis.

Menurut [3] *load balancing* merupakan teknik yang digunakan untuk memisahkan antara dua atau lebih *network link*. Menurut [5] *load balancing* adalah teknik untuk mendistribusikan internet dua jalur koneksi secara seimbang. Dengan penerapan *load balancing* trafik akan berjalan lebih optimal, memaksimalkan *throughput* dan menghindari *overload* pada jalur koneksi.

Menurut [3] *failover* merupakan teknik yang menerapkan beberapa jalur untuk mencapai *network* tujuan. Menurut [8] *failover* suatu teknik untuk berpindah ke sistem cadangan jika sistem utama mengalami kegagalan.

Menurut [8] dalam penelitiannya *load balancing* dan *failover* yang diterapkan pada server mampu menstabilkan dan menjaga performa sistem dapat dijadikan solusi untuk situs dengan trafik tinggi sehingga tidak terjadi kegagalan pada sistem dan aplikasi web yang terpasang pada sistem.

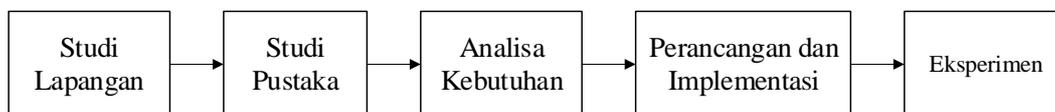
Riset [9] menggunakan *load balancing* dengan metode *ECMP*. Hasil dari penelitian yang dilakukan dapat diketahui *load balancing* dengan metode *ECMP* tidak dapat mengakumulasi *bandwidth* tetapi dapat membagi beban trafik menjadi sama rata berdasarkan perbandingan kecepatan internet. *ECMP* merupakan *persistent-per-connection* dengan begitu jika salah satu link koneksi *down*, *check-gateway* akan memutuskan *link* tersebut dan menggunakan *link* koneksi lain yang masih aktif. Menurut [10] *load balancing* dengan metode *ECMP* merupakan suatu teknik untuk mengatur rute paket melalui beberapa *link* yang mempunyai nilai yang sama.

Menurut [11] *Quality of Service (QoS)* merupakan metode untuk mengetahui tentang seberapa baik jaringan yang sudah terpasang. Menurut [12] *QoS* digunakan sebagai cara satu cara untuk mengetahui seberapa besar kualitas layanan data yang harus di penuhi. Parameter *QoS* yang digunakan untuk analisa layanan komunikasi data adalah *throughput*, *delay*, *jitter*, dan *packet loss*. Menurut [13] dalam penelitiannya *QoS* digunakan untuk menilai seberapa baik jaringan dan upaya untuk menentukan karakteristik dari layanan.

3. METODE PENELITIAN

3.1. Metodologi

Metodologi penelitian yang digunakan dalam melakukan implementasi menggunakan rancangan penelitian seperti terlihat pada Gambar 1.



Gambar 1. Rancangan penelitian

Tahap pertama yaitu studi lapangan. Penelitian ini dilakukan berdasarkan permasalahan pada beberapa perusahaan yang menggunakan lebih dari satu *link* koneksi internet tetapi tidak digunakan secara optimal yaitu tidak ada penerapan *load balancing* dan *failover*.

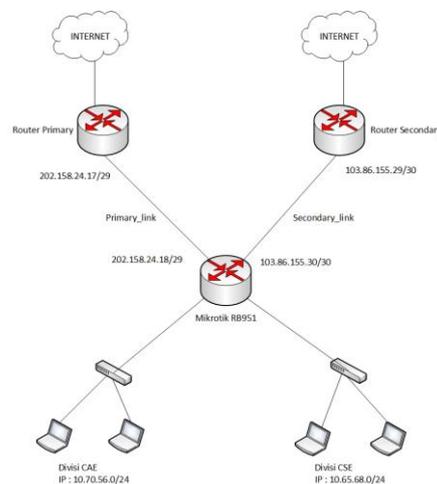
Topik dari penelitian ini adalah menerapkan *load balancing* dengan metode *ECMP*. Tahap kedua yaitu studi pustaka, pada tahap ini penulis mencari informasi terkait dengan penelitian yang dilakukan. Informasi tersebut diperoleh dari buku, tesis, jurnal dan *website* mikrotik Indonesia.

Tahap ketiga yaitu analisa kebutuhan. Tahap ini dilakukan untuk menentukan berupa perangkat keras dan perangkat lunak yang akan dipakai dalam penelitian ini. Spesifikasi perangkat yang digunakan dapat dilihat pada Tabel 1.

Tabel 1. Spesifikasi perangkat

No.	Spesifikasi Perangkat	Keterangan
1	MikroTik RB951Ui-2HnD	Router yang digunakan untuk implementasi
2	Windows 10	Sistem operasi laptop pengguna
3	Mikrotik routerOS 6.42.7	Sistem operasi pada mikroTik RB951Ui-2HnD
4	Winbox v3.17	Perangkat lunak yang digunakan untuk melakukan konfigurasi router mikrotik.

Tahap keempat yaitu perancangan dan implementasi. Pada tahap perancangan penulis membuat topologi jaringan yang akan diimplementasikan pada divisi *Customer Support Engineer (CSE)* dan *Customer Accounting Executive (CAE)* pada PT. Cyberindo Aditama. Topologi jaringan yang akan dibangun dapat dilihat pada Gambar 2.



Gambar 2. Topologi Jaringan

Setelah melakukan perancangan topologi jaringan kemudian memasuki tahap implementasi. Implementasi yang diterapkan pada dua divisi memiliki *Network address* pada divisi CSE adalah 10.65.68.0/24 dan divisi CAE adalah 10.70.56.0/24.

Tahap terakhir yaitu eksperimen. Pada tahap ini dilakukan skenario eksperimen sebagai berikut:

- Skenario satu pengujian saat terjadi *load balancing* atau kedua *link* koneksi internet aktif dilakukan dengan melakukan aktivitas *download file* (unduh) dengan menggunakan *tools internet download manager* dan *streaming* di youtube. Ukuran *file* ubuntu server yang diunduh adalah 812 MB. *Streaming* youtube dengan kualitas 720p (resolusi 1280x720) dan 30 *frames per second* atau *fps*.
- Skenario dua pengujian saat terjadi *failover* atau salah satu *link* koneksi internet terputus dilakukan dengan melakukan aktivitas *download file* (unduh) dengan menggunakan *tools internet download manager* dan *streaming* di youtube. Ukuran *file* ubuntu server yang diunduh adalah 812 MB. *Streaming* youtube dengan kualitas 720p (resolusi 1280x720) dan 30 *frames per second* atau *fps*.

- Skenario tiga pengukuran *Quality of Service (QoS)* dengan parameter *throughput*, *delay*, dan *jitter*. Pengukuran dilakukan saat terjadi *load balancing* dan *failover*. Pengukuran dilakukan dengan pengujian *download file* (unduh) dengan menggunakan *tools internet download manager* dan *streaming* di youtube. *File* yang diunduh adalah fedora workstation yang berukuran 23,501 MB. *Streaming* youtube dengan kualitas 720p (resolusi 1280x720) dan 30 *frames per second* atau *fps*.

3.2. *Quality of Service (QoS)*

QoS merupakan metode pengukuran untuk mendefinisikan karakteristik atau untuk mengukur seberapa baik jaringan yang sudah dibangun [13]. *QoS* yang digunakan pada penelitian ini menggunakan standarisasi *Telecommunication and internet protocol (TIPHON)*. *TIPHON* dirancang untuk mendukung komunikasi suara dan aspek multimedia terkait antara pengguna jaringan berbasis IP [14]. Untuk parameter *throughput*, *delay*, dan *jitter* menurut *TIPHON* adalah sebagai berikut :

- *Throughput*
Throughput merupakan jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu.

Tabel 2. Kategori *throughput*

Kategori <i>Throughput</i>	<i>Throughput</i> (%)	Indeks
Sangat Bagus	100	4
Bagus	75	3
Sedang	50	2
Buruk	<25	1

- *Delay*
Delay merupakan waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan.

Tabel 3. Kategori *delay*

Kategori <i>Delay</i>	<i>Delay</i> (ms)	Indeks
Sangat Bagus	<150	4
Bagus	150 s/d 300	3
Sedang	300 s/d 450	2
Buruk	>450	1

- *Jitter*
Jitter atau variasi *delay* menunjukkan banyaknya variasi *delay* pada transmisi jaringan. Hal ini diakibatkan oleh variasi-variasi panjang antrian, waktu pengolahan data, dan waktu penghimpunan ulang paket-paket.

Tabel 4. Kategori *jitter*

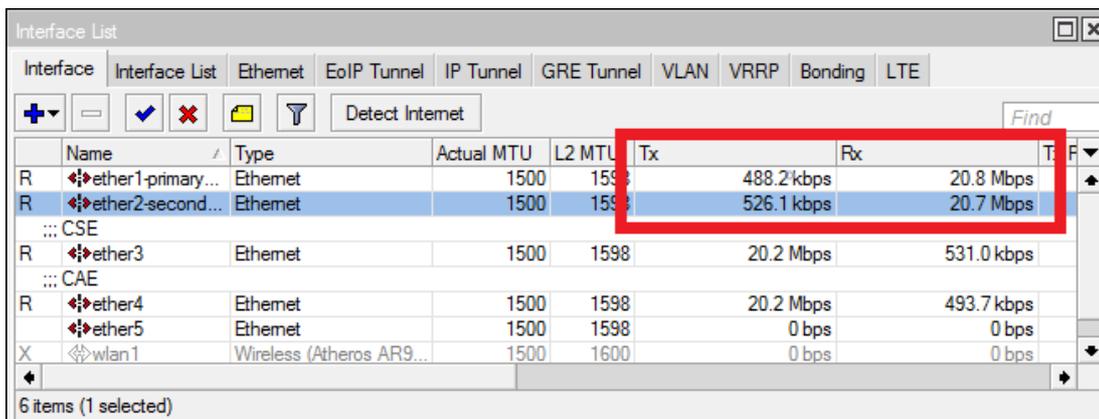
Kategori <i>Jitter</i>	<i>Jitter</i> (ms)	Indeks
Sangat Bagus	0	4
Bagus	0 s/d 75	3
Sedang	75 s/d 125	2
Buruk	125 s/d 225	1

4. HASIL DAN PEMBAHASAN

Pada penelitian ini digunakan dua *link* koneksi internet dengan perbandingan *bandwidth* 1:1 yaitu 20 Mbps dan 20 Mbps. Pada bagian ini dilakukan pengujian *load balancing* dan *failover* serta dilakukan pengukuran *QoS*. Pengujian yang dilakukan pada penelitian ini dapat dilihat pada bagian skenario eksperimen pada bab metode penelitian.

4.1. Hasil Load Balancing

Pada bagian ini dilakukan pengujian terhadap *load balancing* dengan metode *ECMP* untuk melihat pembagian beban trafik pada kedua jalur koneksi yang dapat dilihat bagian *interface list* melalui *tools* Winbox.



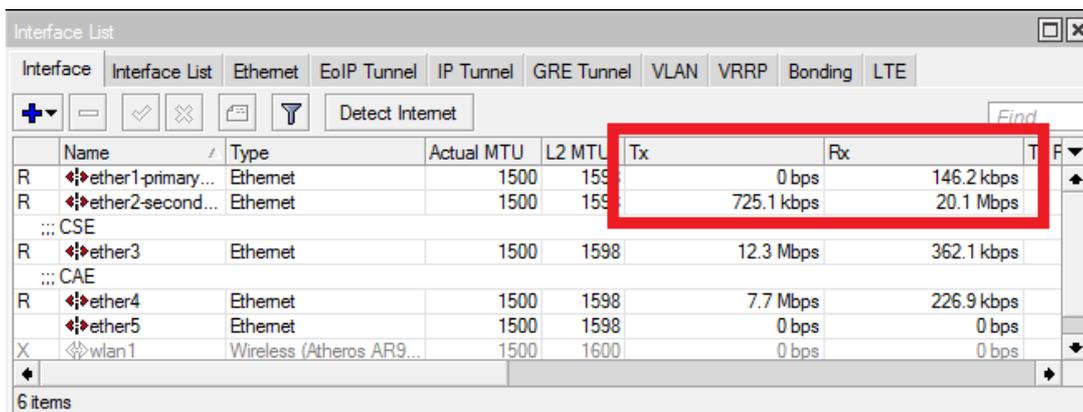
Name	Type	Actual MTU	L2 MTU	Tx	Rx
R ether1-primary...	Ethernet	1500	1598	488.2 kbps	20.8 Mbps
R ether2-second...	Ethernet	1500	1598	526.1 kbps	20.7 Mbps
R ether3	Ethernet	1500	1598	20.2 Mbps	531.0 kbps
R ether4	Ethernet	1500	1598	20.2 Mbps	493.7 kbps
R ether5	Ethernet	1500	1598	0 bps	0 bps
X wlan1	Wireless (Atheros AR9...	1500	1600	0 bps	0 bps

Gambar 3. Hasil *load balancing* kedua *link* koneksi aktif

Dari Gambar 3 terlihat bahwa dengan dilakukan pengujian saat kedua *link* koneksi aktif maka beban trafik dibagi secara seimbang pada kedua *link* koneksi.

4.2. Hasil Failover

Di bagian ini *interface list* berguna untuk mengetahui beban trafik yang terdapat pada kedua *link* koneksi internet saat terjadi *failover* atau salah satu *link* koneksi terputus.



Name	Type	Actual MTU	L2 MTU	Tx	Rx
R ether1-primary...	Ethernet	1500	1598	0 bps	146.2 kbps
R ether2-second...	Ethernet	1500	1598	725.1 kbps	20.1 Mbps
R ether3	Ethernet	1500	1598	12.3 Mbps	362.1 kbps
R ether4	Ethernet	1500	1598	7.7 Mbps	226.9 kbps
R ether5	Ethernet	1500	1598	0 bps	0 bps
X wlan1	Wireless (Atheros AR9...	1500	1600	0 bps	0 bps

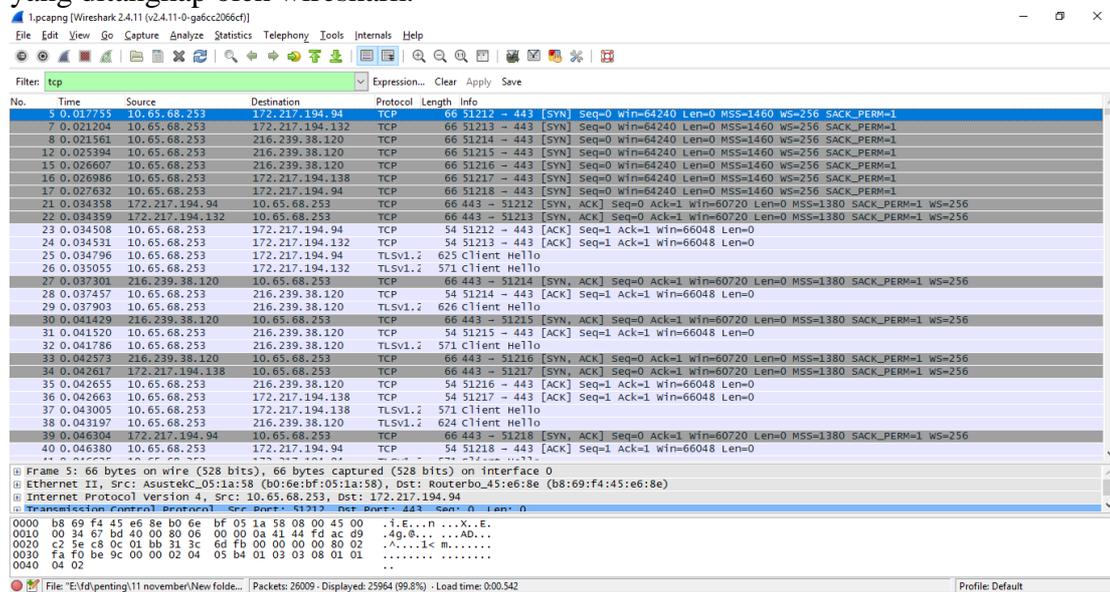
Gambar 4. Hasil *failover* setelah *primary link* terputus

Berdasarkan Gambar 4 menjelaskan bahwa dari penerapan *load balancing* dengan metode *ECMP* dapat memberikan efek *failover*. Hal tersebut ditandai saat

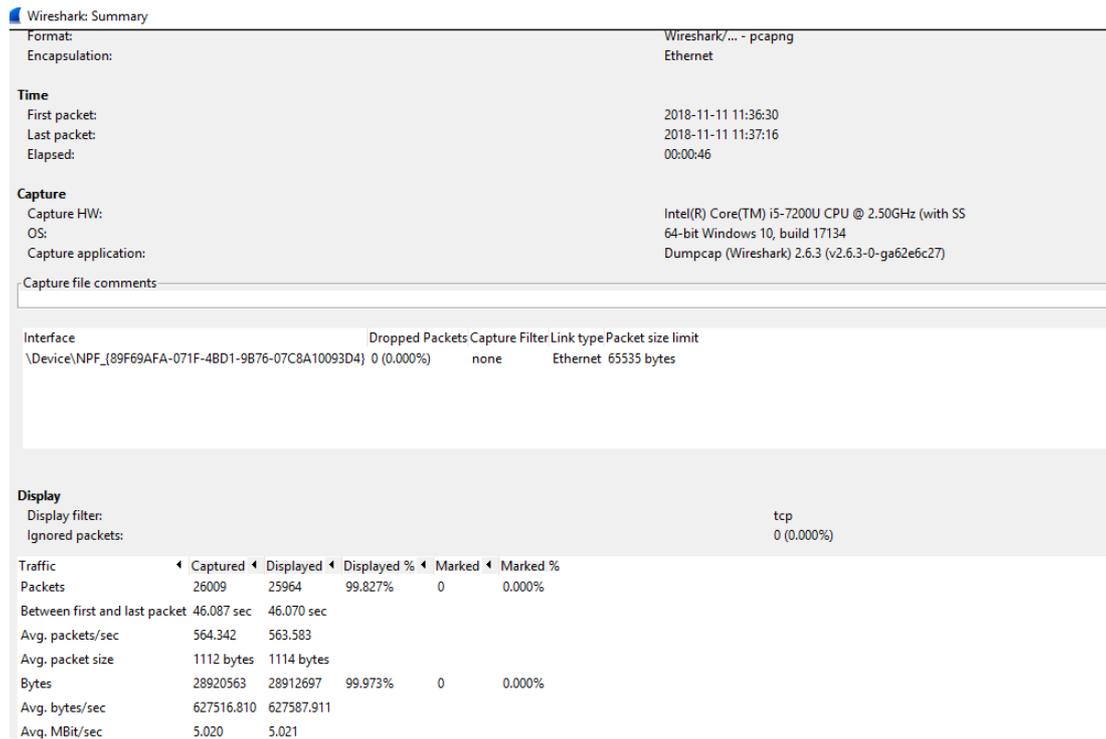
primary link terputus maka beban trafik berpindah secara otomatis ke secondary link. Sebaliknya saat secondary link terputus maka beban trafik berpindah ke primary link.

4.3. Pengukuran Quality of Service (QoS)

Di bagian ini dilakukan pengamatan sebanyak 10 kali selama proses pengujian berlangsung yaitu saat terjadi load balancing dan failover. Pengukuran QoS dilakukan dengan melakukan pengujian download file fedora workstation yang berukuran 23,501 MB menggunakan tools internet download manager dan melakukan streaming di youtube dengan kualitas 720 pixel 30 fps. Pengukuran QoS dalam penelitian ini menggunakan parameter throughput, delay, dan jitter. Pada pengukuran QoS dibutuhkan tools wireshark untuk menangkap (capture) setiap paket yang melintasi jaringan. Untuk membuktikan paket yang ditangkap oleh wireshark penulis melampirkan salah satu capture yang dapat dilihat pada Gambar 5 dan pada Gambar 6 merupakan hasil ringkasan (summary) dari paket yang ditangkap oleh wireshark.

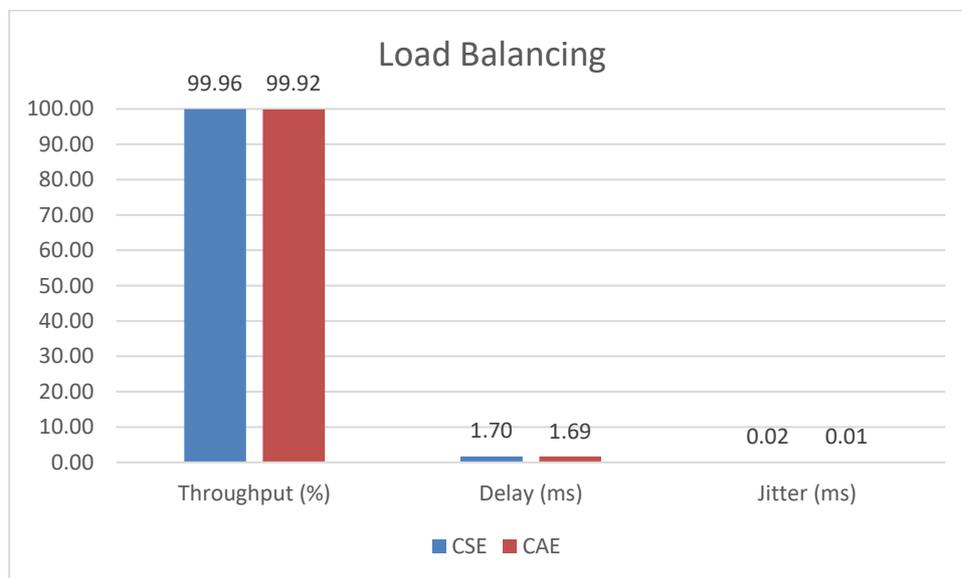


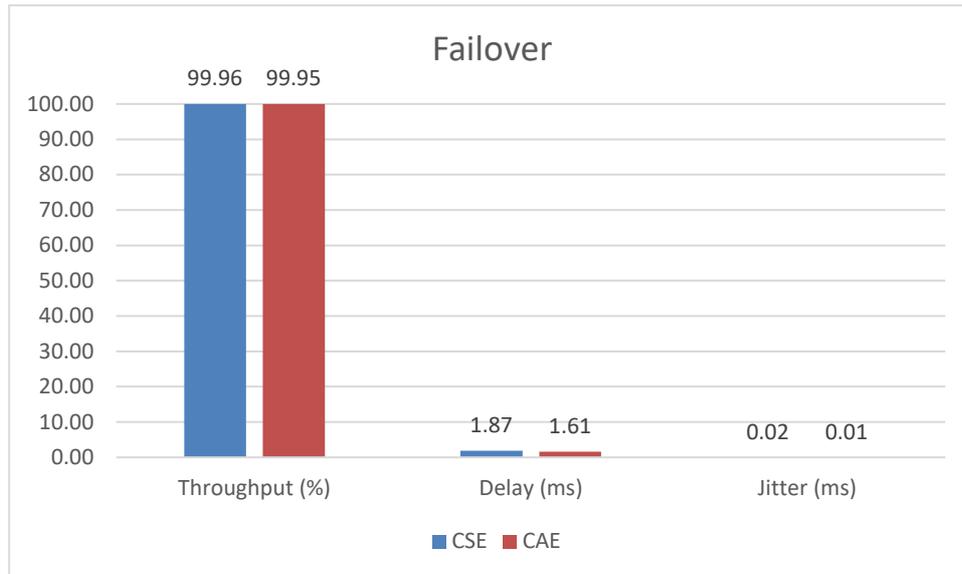
Gambar 5. Hasil capture pada salah satu pengguna



Gambar 6. Summary Wireshark

Pengukuran *QoS* dalam penelitian ini dilakukan dengan kondisi jaringan kecil dan dalam tidak padat. Pengukuran ini untuk mengetahui karakteristik dari jaringan yang sudah dibangun. Penulis melampirkan grafik rata-rata *throughput*, *delay*, dan *jitter* yang dapat dilihat pada Gambar 7 dan Gambar 8.

Gambar 7. Hasil pengukuran *QoS* ketika *load balancing*



Gambar 8. Hasil pengukuran *QoS* ketika *failover*

Berdasarkan grafik pada Gambar 5 pengamatan dilakukan saat terjadi *load balancing* diperoleh bahwa divisi *CSE* memiliki *throughput* 99.96% (*sangat bagus*), *delay* 1.70 ms (*sangat bagus*), dan *jitter* 0.02 ms (*sangat bagus*) sedangkan divisi *CAE* memiliki *throughput* 99.92% (*sangat bagus*), *delay* 1.69 ms (*sangat bagus*), dan *jitter* 0.01 ms (*sangat bagus*) serta berdasarkan grafik pada Gambar 6 pengamatan dilakukan saat terjadi *failover* diperoleh bahwa divisi *CSE* memiliki *throughput* 99.96% (*sangat bagus*), *delay* 1.87 ms (*sangat bagus*), dan *jitter* 0.02 ms (*sangat bagus*) sedangkan divisi *CAE* memiliki *throughput* 99.95% (*sangat bagus*), *delay* 1.61 ms (*sangat bagus*), dan *jitter* 0.01 ms (*sangat bagus*).

5. KESIMPULAN

Berdasarkan eksperimen yang telah dilakukan dalam penelitian ini, maka dapat memberikan kesimpulan sebagai berikut :

1. *Load Balancing* dengan metode *ECMP* yang diterapkan dapat membagi beban trafik pada dua *link* koneksi secara seimbang.
2. Berdasarkan pengamatan saat terjadi *load balancing* diperoleh bahwa divisi *CSE* memiliki *throughput* 99.96% (*sangat bagus*), *delay* 1.70 ms (*sangat bagus*), dan *jitter* 0.02 ms (*sangat bagus*) sedangkan divisi *CAE* memiliki *throughput* 99.92% (*sangat bagus*), *delay* 1.69 ms (*sangat bagus*), dan *jitter* 0.01 ms (*sangat bagus*).
3. Efek *failover* yang dihasilkan dari penerapan *load balancing* metode *ECMP* berjalan dengan baik.
4. Berdasarkan pengamatan saat terjadi *failover* diperoleh bahwa divisi *CSE* memiliki *throughput* 99.96% (*sangat bagus*), *delay* 1.87 ms (*sangat bagus*), dan *jitter* 0.02 ms (*sangat bagus*) sedangkan divisi *CAE* memiliki *throughput* 99.95% (*sangat bagus*), *delay* 1.61 ms (*sangat bagus*), dan *jitter* 0.01 ms (*sangat bagus*).

REFERENCES

- [1] F. Utami, Lindawati, and Suzanzeffi, Optimalisasi Load Balancing Dua Isp Untuk Manajemen Bandwidth Berbasis Mikrotik, *Pros. Snatif*, pp. 153–160,

- 2015.
- [2] R. Pambudi and M. A. Muslim, Implementasi Policy Base Routing dan Failover Menggunakan Router Mikrotik untuk Membagi Jalur Akses Internet di FMIPA Unnes, *J. Teknol. dan Sist. Komput.*, vol. 5, no. 2, p. 57, 2017.
 - [3] T. I. Darmawan, Analisa Link Balancing dan Failover 2 Provider Menggunakan Border Gateway Protocol (BGP) Pada Router Cisco 7606s, *J. Teknol. dan Sist. Inf.*, vol. 03, pp. 326–333, 2017.
 - [4] D. jalu N. BP, Implementasi Load Balancing dan Failover Dua ISP Menggunakan Mikrotik di PT. XYZ, Universitas Mercu Buana, 2017.
 - [5] A. Frayogi, W. Yahya, and R. A. Setiawan, Perbandingan Kinerja RouterOS Mikrotik dan Zeroshell pada Mekanisme Load Balancing Serta Failover, *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 7, pp. 2689–2697, 2018.
 - [6] P. Citraweb Solusi Teknologi, Load Balance Metode ECMP, <http://www.mikrotik.co.id/>. [Online]. Available: http://mikrotik.co.id/artikel_lihat.php?id=76.
 - [7] M. Anif *et al.*, Implementasi Teknologi Load Balancing Dua Jalur Internet Service Provider (ISP) menggunakan Metode Per Connection Classifier (PCC) di Pondok Pesantren Yasin Kudus, *Jati Emas (Jurnal Apl. Tek. dan Pengabd. Masyarakat)*, vol. 2, no. 1, pp. 26–34, 2018.
 - [8] R. Dani and F. Suryawan, Perancangan dan Pengujian Load Balancing dan Failover menggunakan NGINX, *J. Ilmu Komput. dan Inform. Peranc.*, vol. 3, no. 1, pp. 43–50, 2017.
 - [9] M. F. Zurkarnaen and M. I. Isnaini, Implementasi Load Balancing dengan Metode Equal Cost Multi-Path, vol. 1, no. 1, pp. 13–17, 2018.
 - [10] M. I. Firdaus, Analisis Perbandingan Kinerja Load Balancing Metode ECMP (Equal Cost Multi-Path) Dengan Metode PCC (Per Connection Classifier) Pada Mikrotik RouterOS, vol. 8, no. 3, pp. 165–170, 2017.
 - [11] Jinchuan Yan and Kang Wang, QoS analysis based on ACO in WMSNs, *2015 IEEE 16th Int. Conf. Commun. Technol.*, pp. 525–528, 2015.
 - [12] Y. A. Pranata, I. Fibriani, and S. B. Utomo, Analisis Optimasi Kinerja Quality Of Service Pada Layanan Komunikasi Data Menggunakan NS - 2 di PT . PLN (PERSERO) Jember, *Sinergi*, Vol. 20, No. 2, pp. 149–156, 2016.
 - [13] R. Wulandari, Analisis QoS (Quality Of Service) Pada Jaringan Internet (Studi Kasus : UPT Loka Uji Teknik Penambangan Jampang Kulon – LIPI), *J. Tek. Inform. dan Sist. Inf.*, vol. 2, pp. 162–172, 2016.
 - [14] Andriatmoko, M. Sadikin, and S. Riyadi, Redundancy Protocol (VRRP) dengan Load Sharing pada Router Gateway berbasis Multi Vendor, 2018.