

HALAMAN JUDUL

ANALISIS PERBANDINGAN KINERJA LOAD BALANCING METODE ECMP (EQUAL COST MULTI-PATH) DENGAN METODE PCC (PER CONNECTION CLASSIFIER) PADA MIKROTIK ROUTEROS

TUGAS AKHIR

**Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Informatika**



Oleh :

Nama : Hafni Candra Madewa

No. Mahasiswa : 05523195

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA**

2013

LEMBAR PENGESAHAN PEMBIMBING

**ANALISIS PERBANDINGAN KINERJA LOAD BALANCING METODE
ECMP (EQUAL COST MULTI-PATH) DENGAN METODE PCC
(PER CONNECTION CLASSIFIER) PADA MIKROTIK ROUTEROS**



Pembimbing,

Syarif Hidayat, S.Kom., MIT.

LEMBAR PENGESAHAN PENGUJI

ANALISIS PERBANDINGAN KINERJA LOAD BALANCING METODE ECMP (EQUAL COST MULTI-PATH) DENGAN METODE PCC (PER CONNECTION CLASSIFIER) PADA MIKROTIK ROUTEROS

TUGAS AKHIR

Oleh:

Nama : Hafni Candra Madewa

No. Mahasiswa : 05523195

Telah Dipertahankan di Depan Sidang Penguji Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 27 November 2013

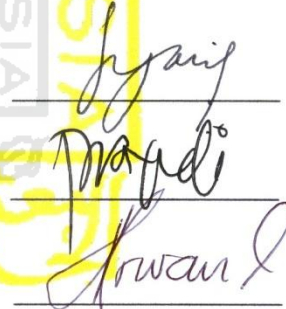
Tim Penguji

Tanda Tangan

Syarif Hidayat, S.Kom., MIT.
Ketua

Yudi Prayudi, S.Si., M.Kom.
Anggota I

Arwan Ahmad Khoiruddin, S.Kom., M.Cs.
Anggota II



Mengetahui,
Ketua Jurusan Teknik Informatika
Universitas Islam Indonesia



Yudi Prayudi, S.Si., M.Kom.

HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Kupersembahkan tugas akhir ini
dengan setulus hati.

Untuk yang tercinta dan yang terhormat,
Ayahanda (Latif Usman) dan Ibunda (Smarmi).

Dan untuk yang saya cintai dan sayangi,
Istriku (Wulan), Anakku (Hanifah)
dan Adikku (Sekar).

Yang telah memberikan dorongan semangat, do'a serta dukungan
material maupun spiritual selama masa studi.

Serta untuk Almamater ku,
Jurusan Teknik Informatika, Universitas Islam Indonesia.

HALAMAN MOTTO

“Karena sesungguhnya bersama setiap kesulitan ada kemudahan, Sesungguhnya bersama setiap kesulitan ada kemudahan.”

(Q.S. Al Insyirah 5-6)

“... Sesungguhnya Allah tidak mengubah keadaan sesuatu kaum sehingga mereka mengubah keadaan yang ada pada diri mereka sendiri...”

(Q.S. Ar Ra'd 11)

“Hai orang-orang yang beriman, jika kamu menolong (agama) Allah, niscaya Dia akan menolongmu dan meneguhkan kedudukanmu.”

(Q.S. Muhammad 7)

"Orang yang berakal bukanlah yang dapat membedakan antara kebaikan dan keburukan, Tapi orang yang berakal adalah yang tatkala melihat kebaikan ia mengikutinya, dan bila melihat keburukan ia menghindarinya".

(Ibnu Sirin Rahimahullah)

"Jika engkau bisa, jadilah seorang ulama. Jika engkau tidak mampu, maka jadilah penuntut ilmu. Bila engkau tidak bisa menjadi seorang penuntut ilmu, maka cintailah mereka. Dan jika kau tidak mencintai mereka, janganlah engkau benci mereka."

(Umar bin Abdul Aziz Rahimahullah)

“Berusahalah untuk duniamu seakan-akan kau hidup abadi, beribadahlah untuk akhiratmu seakan-akan kau mati esok hari.”

(Peribahasa Arab)

KATA PENGANTAR

Assalamu 'alaikum Warohmatullahi Wabarokatuh

Bismillaahirrahmaanirrahiim, Segala puji dan syukur senantiasa tercurah ke hadirat Allah Tabaraka wa Ta'ala. Karena hanya karena rahmat dan karunia dari Allah-lah penulis dapat menyelesaikan tugas akhir yang berjudul “ANALISIS PERBANDINGAN KINERJA LOAD BALANCING METODE ECMP (EQUAL COST MULTI-PATH) DENGAN METODE PCC (PER CONNECTION CLASSIFIER) PADA MIKROTIK ROUTEROS”. Tak lupa shalawat serta salam kita tujukan kepada Nabi Muhammad Shalallahu ‘Alaihi wa Sallam, serta kepada para keluarga, sahabat, serta para pengikut sunnah dan ajaran beliau hingga akhir zaman.

Laporan Tugas Akhir ini disusun sebagai salah satu syarat guna memperoleh gelar sarjana Teknik Informatika pada Universitas Islam Indonesia. Dan dengan proses pengerjaan Tugas Akhir ini, penulis dapat menerapkan secara langsung disiplin ilmu yang telah bertahun-tahun penulis pelajari dan alami di Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia yang tercinta ini.

Penulis menyampaikan ucapan terimakasih atas bantuan, bimbingan, dukungan dan do'a dari berbagai pihak yang membantu kelancaran pelaksanaan dan penyelesaian Tugas Akhir ini. Untuk itu dengan segala kerendahan hati penulis mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Allah Tabaraka wa Ta'ala, Satu satunya sesembahan yang berhak untuk diibadahi secara benar. Yang Maha Kuasa atas segala sesuatu di langit dan di bumi.

2. Nabi Muhammad SAW, Rasul akhir zaman, suri tauladan bagi seluruh umat islam, atas usaha dakwah tauhid Beliau dan dengan izin Allah, kita terangkat dari zaman kebodohan dan kejahiliyahan.
3. Ayahanda tercinta Latif Usman serta Ibunda tercinta Ismarmi, terimakasih atas do'a yang selalu terucap dan segala pengorbanan untuk kehidupan ku. Hanya maaf dan terimakasih yang bisa ananda ucapkan saat ini. Semoga Allah memberi kemampuan untuk anda agar bisa berbakti sampai akhir hayat dan membalas kebaikan Ayah dan Bunda sesuai kemampuan anda.
4. Istriku Diana Wulan, terimakasih atas dorongan semangat yang selalu kau berikan ketika aku hampir putus asa.
5. Bapak Yudi Prayudi, S.Si., M.Kom, selaku Ketua Jurusan Teknik Informatika. Terima kasih atas kemudahan dan dukungan yang telah diberikan.
6. Bapak Syarif Hidayat, S.Kom., MIT., selaku Dosen Pembimbing, aku belajar banyak dari semangat Bapak, terimakasih atas pengarahan, bimbingan, motivasi, sms dan telepon dari Bapak untuk mengingatkan saya, serta masukan selama pelaksanaan tugas akhir dan penulisan laporan.
7. Seluruh dosen dan staf pengajar Jurusan Teknik Informatika, terimakasih atas ilmu dan pengetahuan yang telah disampaikan kepada saya selama menempuh studi di Universitas Islam Indonesia, semoga menjadi amal jariyah Bapak-bapak dan Ibu-ibu sekalian.
8. Teman-teman mahasiswa di Universitas Islam Indonesia, terimakasih atas kerja sama, kekompakan, kekeluargaan dan persahabatan yang selama ini kalian berikan, serta seluruh teman-teman Informatika UII, jadilah ilmu yang berakhlak Al-Qur'an.
9. Semua pihak yang tidak dapat penulis sebutkan satu persatu dalam membantu sejak pengumpulan data sampai penyusunan Tugas Akhir ini.

Semoga amal ibadah dan kebaikan yang telah diberikan mendapat imbalan yang setimpal dari Allah SWT.

Terakhir penulis berharap agar Tugas Akhir ini termasuk ilmu yang bermanfaat bagi orang lain, dan dapat bernilai ibadah jariyah di sisi Allah Tabaraka wa Ta'ala. Penulis juga menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan, oleh sebab itu penulis sangat mengharapkan kritik serta saran yang bersifat membangun untuk perbaikan dimasa mendatang. Semoga Tugas Akhir ini dapat berguna dan memberikan manfaat bagi penulis serta semua pembaca. *Baarakallahu fiikum.*

Wassalamu'alaikum Warohmatullahi Wabarokatuh



Yogyakarta, 27 September 2013

Penulis

ABSTRAKSI

Salah satu solusi untuk mendapatkan kualitas layanan internet yang lebih baik adalah dengan memanfaatkan teknologi *load balancing*. Banyaknya penyedia layanan internet di Indonesia memberikan kita banyak pilihan operator mana yang akan kita gunakan layanannya. Kita dapat menggunakan lebih dari satu koneksi internet dari penyedia layanan internet yang berbeda yang kemudian di seimbangkan dengan teknologi *load balancing*.

Load balancing metode ECMP (Equal Cost Multi Path) dan metode PCC (Per Connection Classifier) merupakan contoh metode *load balancing* yang sering diterapkan pada jaringan komputer. Permasalahan yang kita hadapi adalah, diantara metode *load balancing* tersebut masing-masing mempunyai beberapa kelebihan dan kekurangan.

Tugas akhir dengan judul “Analisis Perbandingan Kinerja *Load balancing* Metode ECMP (Equal Cost Multi-Path) Dengan Metode PCC (Per Connection Classifier)” ini akan menganalisis bagaimana perbandingan kinerja antara kedua metode *load balancing* dalam sebuah sistem jaringan komputer yang memiliki dua gateway.

Rancangan yang telah dibuat kemudian akan disimulasikan dengan menggunakan sebuah aplikasi mesin virtual yang dapat membantu merealisasikan rancangan yang telah dibuat di dalam sebuah komputer dengan virtualisasi.

Kata kunci : *Load balancing*, ECMP, PCC, kinerja, *qos*

TAKARIR

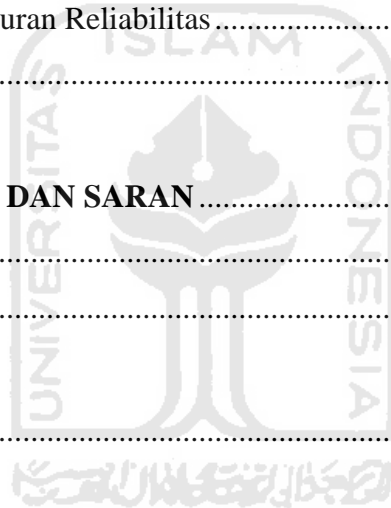
<i>address</i>	:	alamat
<i>availability</i>	:	ketersediaan
<i>bandwidth</i>	:	pita lebar
<i>client</i>	:	pelanggan
<i>connectionless</i>	:	tidak memperhitungkan koneksi
<i>command line</i>	:	baris perintah
<i>engine</i>	:	mesin
<i>gateway</i>	:	gerbang
<i>hop</i>	:	lompatan
<i>interface</i>	:	antar muka
<i>load balancing</i>	:	metode untuk menyeimbangkan beban
<i>offline</i>	:	terputus dari jaringan
<i>online</i>	:	terhubung dalam jaringan
<i>resource</i>	:	sumber daya
<i>request</i>	:	permintaan
<i>router</i>	:	pengatur rute lalu lintas data
<i>server</i>	:	penyedia layanan
<i>service</i>	:	layanan
<i>throughput</i>	:	kecepatan data aktual
<i>traffic</i>	:	lalu lintas
<i>virtual machine</i>	:	mesin virtual

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN PEMBIMBING	ii
LEMBAR PENGESAHAN PENGUJI	iii
HALAMAN PERSEMBAHAN	iv
HALAMAN MOTTO	v
KATA PENGANTAR	vi
ABSTRAKSI	ix
TAKARIR	x
DAFTAR ISI	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Metodologi Penelitian	3
1.7 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Jenis Penelitian.....	6
2.2 Kinerja Jaringan Komputer	6
a. throughput	6
b. packet loss	7
c. jitter	7
d. RTT (Round Trip Time).....	8
e. reliabilitas.....	9

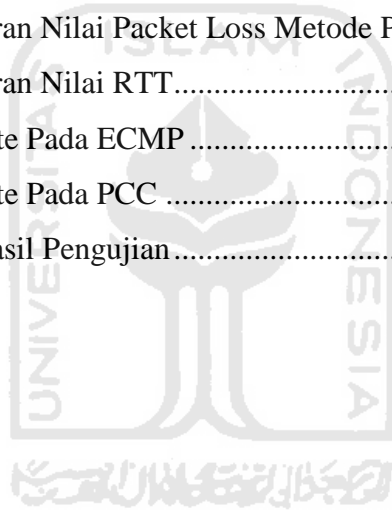
2.3	Load balancing	9
a.	ECMP (Equal Cost Multi-Path) Load Balancing.....	9
b.	PCC (Per Connection Classifier) Load Balancing	10
2.4	Mikrotik Router OS	10
2.5	Virtual Machine	10
BAB III METODOLOGI		11
3.1	Perencanaan Skenario	11
3.2	Perancangan Topologi.....	12
3.3	Analisis Kebutuhan	13
3.4	Implementasi	14
a.	Instalasi Virtual Machine	14
b.	Persiapan Jaringan Komputer pada Virtual Machine.....	14
c.	Pengaturan Konfigurasi Load Balancing PCC dan ECMP	14
d.	Pengujian.....	15
e.	Analisis Hasil Pengujian	15
BAB IV HASIL DAN PEMBAHASAN		16
4.1	Konfigurasi Virtual Machine	16
4.2	Pembangunan Jaringan Komputer	17
a.	Instalasi Linux Mint	18
b.	Instalasi MikroTik RouterOS	18
c.	Konfigurasi Jaringan Komputer.....	19
d.	Konfigurasi Router1	19
e.	Konfigurasi Router2.....	20
f.	Konfigurasi Router3.....	21
g.	Konfigurasi pada Load Balancer Metode ECMP.....	22
h.	Konfigurasi pada Load Balancer Metode PCC	23
g.	Konfigurasi Network Analyzer	25

4.3	Pengujian.....	26
a.	Pengukuran Nilai Throughput Menggunakan Iperf	26
b.	Pengukuran Nilai Packet Loss dan Jitter Menggunakan Iperf	27
c.	Pengukuran Nilai RTT Menggunakan Ping.....	28
d.	Pengukuran Reliabilitas Jaringan Menggunakan Traceroute.....	29
4.4	Hasil Pengujian	30
a.	Hasil Pengukuran Nilai Throughput	29
b.	Hasil Pengukuran Nilai Jitter	30
c.	Hasil Pengukuran Nilai Packet Loss	32
d.	Hasil Pengukuran Nilai RTT.....	33
e.	Hasil Pengukuran Reliabilitas	34
4.5	Pembahasan.....	37
BAB V KESIMPULAN DAN SARAN		39
5.1	Kesimpulan	39
5.2	Saran.....	39
DAFTAR PUSTAKA		40



DAFTAR TABEL

Tabel 2.1 Kategori Degradasi Terhadap Tingkatan Packet Loss.....	7
Tabel 2.2 Kategori Degradasi Terhadap Tingkatan Jitter	8
Tabel 2.3 Standar Toleransi Delay.....	8
Tabel 3.1 Daftar Kebutuhan Perangkat Keras dan Perangkat Lunak.....	14
Tabel 4.1 Interface Jaringan, Alamat IP Dan Konfigurasinya pada VMware	19
Tabel 4.2 Hasil Pengukuran Nilai Jitter Metode ECMP	30
Tabel 4.3 Hasil Pengukuran Nilai Jitter Metode PCC	31
Tabel 4.4 Hasil Pengukuran Nilai Packet Loss Metode ECMP.....	32
Tabel 4.5 Hasil Pengukuran Nilai Packet Loss Metode PCC	32
Tabel 4.6 Hasil Pengukuran Nilai RTT.....	33
Tabel 4.7 Hasil Traceroute Pada ECMP	36
Tabel 4.8 Hasil Traceroute Pada PCC	37
Tabel 4.9 Rangkuman Hasil Pengujian.....	38



DAFTAR GAMBAR

Gambar 3.1 Struktur rancangan topologi jaringan.....	12
Gambar 4.1 tampilan antarmuka vmware workstation	16
Gambar 4.2 tampilan menu add hardware wizard pada vmware	17
Gambar 4.3 tampilan linux mint pada virtual machine.....	18
Gambar 4.4 tampilan mikrotik routeros pada virtual machine	18
Gambar 4.5 proses instalasi iperf pada linux mint.....	25
Gambar 4.6 iperf pada server mendengarkan koneksi protocol tcp.....	26
Gambar 4.7 iperf pada client mengirimkan koneksi protocol tcp.....	27
Gambar 4.8 iperf pada sever mendengarkan koneksi protocol udp.....	27
Gambar 4.9 iperf pada server mengirimkan koneksi protocol udp.....	28
Gambar 4.10 ping dari client ke server	28
Gambar 4.11 grafik hasil throughput pada ecmp	29
Gambar 4.12 grafik hasil throughput pada pcc	30
Gambar 4.13 grafik jitter pada ecmp dan pcc	31
Gambar 4.14 grafik packet loss pada ecmp dan pcc	33
Gambar 4.15 grafik rata-rat RTT pada ecmp dan pcc.....	34
Gambar 4.16 traceroute pertama pada ecmp.....	35
Gambar 4.17 traceroute pada ecmp setelah pemutusan salah satu gateway	35
Gambar 4.18 traceroute ke-6 dan ke-6 pada ecmp.....	36
Gambar 4.19 traceroute pertama pada pcc	36
Gambar 4.20 traceroute pada pcc setelah pemutusan salah satu gateway	37

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Teknologi informasi, khususnya internet sudah menjadi kebutuhan sehari-hari bagi sebagian orang. Keberadaannya digunakan hampir di berbagai bidang dan tidak dapat dipisahkan dari kegiatan manusia. Akan tetapi pada kenyataannya kualitas layanan internet di negara maju sangat jauh lebih baik bila dibandingkan dengan kualitas layanan internet di negara berkembang seperti di Indonesia. Hal ini disebabkan oleh berbagai macam faktor.

Kondisi geografis di Indonesia yang sangat luas dan berbukit-bukit merupakan salah satu kendala dalam membangun infrastruktur yang baik. Rendahnya kualitas infrastruktur juga berbanding terbalik dengan tingginya angka penggunaan internet di Indonesia. Selain itu persaingan antar penyedia jasa koneksi internet yang tidak sehat menyebabkan adanya perang promosi dimana operator berlomba-lomba menyediakan layanan internet yang murah, tetapi dibalik semua itu yang terjadi adalah penurunan kecepatan maupun kestabilan koneksi.

Salah satu solusi untuk mendapatkan kualitas layanan internet yang lebih baik adalah dengan memanfaatkan teknologi *load balancing*. Banyaknya penyedia layanan internet di Indonesia memberikan kita banyak pilihan operator mana yang akan kita gunakan layanannya. Kita dapat menggunakan lebih dari satu koneksi internet dari penyedia layanan internet yang berbeda yang kemudian di seimbangkan dengan teknologi *load balancing*.

Teknologi *load balancing* ini sudah banyak didukung pada berbagai sistem. Salah satunya adalah pada sistem MikroTik RouterOS. MikroTik, baik RouterBoard maupun RouterOS, banyak digunakan di Indonesia contohnya pada UKM (Usaha Kecil dan Menengah) dan juga pada pengguna perorangan, karena harganya yang lebih terjangkau tetapi dapat menyediakan kinerja yang cukup memadai.

Pada MikroTik RouterOS itu sendiri dapat dijumpai berbagai metode *load balancing* yang bisa kita pilih, diantaranya adalah metode ECMP (Equal Cost Multi-Path) dan metode PCC (Per Connection Classifier). Kedua metode diatas mempunyai karakteristik yang sama, dimana keduanya mendukung penyebaran *traffic* data pada setiap koneksi maupun setiap paket data. Diantara kedua metode *load balancing* tersebut pasti mempunyai kekurangan dan kelebihan masing-masing. Penelitian ini bertujuan untuk mengetahui metode mana yang mempunyai kinerja lebih baik.

1.2. Rumusan Masalah

Dari latar belakang yang telah ditelaah, dapat dihasilkan suatu rumusan masalah. Rumusan masalah yang akan dianalisis dalam penelitian ini yaitu bagaimana perbandingan kinerja antara *load balancing* metode ECMP (Equal Cost Multi-Path) dan metode PCC (Per Connection Classifier) pada sistem Mikrotik RouterOS.

1.3. Batasan Masalah

Load balancing dalam istilah teknologi informasi mempunyai arti yang umum. Maka agar penelitian dapat sesuai dengan objek yang dimaksud dan memudahkan dalam pengambilan data dan pengujian, perlu adanya batasan masalah. Batasan masalah dalam penelitian ini adalah:

- a. *Load balancing* pada penelitian ini adalah *load balancing* lalu lintas data dari dua *gateway* pada suatu *router*.
- b. Penelitian ini dilakukan pada *virtual machine*.
- c. Jumlah *client* hanya satu *client*.
- d. Parameter kualitas kinerja yang digunakan adalah *throughput*, *RTT* (*Round Trip Time*), *packet loss*, *jitter* dan reliabilitas sistem.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah menemukan kelebihan dan kekurangan dari kinerja *load balancing* metode ECMP dan PCC, sehingga dapat menentukan usaha selanjutnya untuk mendapatkan kualitas layanan internet yang lebih baik.

1.5. Manfaat Penelitian

Manfaat dari penelitian ini adalah, dengan mengetahui metode *load balancing* yang lebih baik, kita dapat menekan kerugian finansial maupun kerugian waktu yang terbuang percuma yang diakibatkan oleh kualitas layanan yang kurang baik karena kita bisa menentukan metode mana yang bekerja lebih optimal.

1.6. Metodologi Penelitian

Rangkaian prosedur metodologi penelitian yang penulis lakukan adalah:

a. Perencanaan Skenario

Skenario di butuhkan untuk mengetahui alur jalannya penelitian dari awal sampai akhir.

b. Analisis Kebutuhan

Untuk mengetahui apa saja yang harus dipersiapkan selama penelitian perlu dilakukannya analisis kebutuhan. Kebutuhan yang dimaksud adalah kebutuhan dari sisi perangkat keras maupun dari sisi perangkat lunak.

c. Perancangan Topologi

Setelah semua kebutuhan sistem tersedia, langkah selanjutnya adalah merancang topologi jaringan komputer. Dalam topologi tersebut dirincikan juga *IP address* dari masing-masing komponen.

d. Implementasi

Dalam implementasi penelitian, semua teori yang ada mulai diterapkan dalam praktek yang nyata. Sistem jaringan komputer dibangun berdasarkan rancangan topologi yang ada.

e. Pengujian

Pengujian dilakukan dengan berbagai cara melakukan aktivitas *request* dari *client* ke *server* melalui protocol TCP, UDP maupun ICMP. Lalu lintas data yang melalui *load balancer* di catat aktivitasnya dengan menggunakan aplikasi untuk selanjutnya dihitung nilai performansinya.

f. Analisis

Perhitungan kualitas layanan dilakukan dengan cara melakukan perhitungan terhadap parameter-parameter *QoS* seperti *throughput*, *packet loss*, *RTT (Round Trip Time)* dan *jitter* dengan menggunakan suatu aplikasi pada masing-masing *load balancer*. Kemudian hasil perhitungan dari kedua *load balancer* tersebut dianalisis dan dibandingkan untuk mengetahui hasil yang lebih baik.

1.7. Sistematika Penulisan

Rincian dari sistematika penulisan laporan penelitian ini secara keseluruhan adalah sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini diuraikan latar belakang dari masalah yang ada. Dari latar belakang tersebut akan dapat diambil suatu rumusan masalah. Akan diuraikan pula batasan masalah agar penelitian sesuai dengan objek yang dimaksud. Kemudian dijelaskan apa tujuan yang ingin dicapai, sehingga kita dapat mengetahui manfaat dari penelitian ini. Untuk mengetahui langkah-langkah dari penelitian dijelaskan pula metodologi penelitiannya. Yang terakhir adalah penjabatan dari sistematika penulisan.

BAB II : LANDASAN TEORI

Landasan teori menjelaskan dasar-dasar ilmiah dari teori yang berhubungan dengan objek penelitian yang dilakukan, sebagai dasar dari pembahasan masalah. Dasar-dasar tersebut adalah dasar-dasar ilmiah dari *kinerja* dan *Qos*, *load balancing*, PCC (Per connection classifier), ECMP (equal cost multi-path), MikroTik dan *virtual machine*.

BAB III : METODOLOGI

Dalam bab ini akan dibahas tentang perencanaan skenario penelitian, yang merupakan gambaran umum mengenai langkah-langkah penelitian. Kemudian akan dijelaskan bagaimana topologi dari jaringan yang akan dibangun. Setelah itu akan dijabarkan analisis kebutuhan sistem baik dari segi perangkat keras maupun perangkat lunaknya. Terakhir adalah pembahasan implementasi pada virtual machine secara ringkas.

BAB IV : HASIL DAN PEMBAHASAN

Bab ini menjelaskan secara detail implementasi dari penelitian ini, mulai dari proses paling awal yaitu instalasi, konfigurasi jaringan, konfigurasi dari masing-masing load balancing hingga analisis dari hasil pengujian. Kemudian akan dilakukan perhitungan untuk mengetahui bagaimana perbandingan antara kedua metode load balancing berdasarkan parameter-parameter kualitas layanan atau *QoS*.

BAB V : KESIMPULAN DAN SARAN

Pada bab terakhir akan diuraikan kesimpulan dari tugas akhir serta dikemukakan beberapa saran agar dapat digunakan untuk mengembangkan penelitian tugas akhir ini.

BAB II

LANDASAN TEORI

2.1 Jenis Penelitian

Dalam penelitian ini penulis mengumpulkan berbagai data yang berupa angka-angka dari kedua metode *load balancing*, kemudian menganalisis dan mencari bagaimana perbandingan antara keduanya.

Oleh karena itu penelitian ini merupakan penelitian kuantitatif, yaitu penelitian yang datanya berupa angka-angka atau pernyataan-pernyataan yang di angkakan dan dianalisis dengan analisis statistik untuk mencari koefisien tertentu, dan dari perhitungan-perhitungan statistik itulah kemudian ditafsirkan (Akbar, 2009).

2.2 Kinerja Jaringan Komputer

Untuk mengukur kinerja dari sebuah jaringan, maka diperlukannya sebuah kualitas layanan atau *Quality of Service (QoS)* yang baik. Nilai dari *Quality of Service* dipengaruhi oleh 5 (lima) hal, yaitu *packet loss*, *throughput*, *delay*, *jitter* dan status koneksi (Disastra, 2010).

Berikut ini adalah parameter-parameter yang mempengaruhi *QoS*.

a. *Throughput*

Throughput adalah kecepatan transfer data efektif yang diukur dalam *Byte/second (Bps)* (Disastra, 2010). Atau dapat juga didefinisikan sebagai jumlah total kedatangan paket yang diamati pada tujuan paket selama waktu tertentu dibagi oleh durasi interval waktu tersebut. Semakin besar *throughput*, maka semakin baik kinerja suatu jaringan komputer.

b. *Packet Loss*

Packet loss yaitu paket gagal dikirm selama terjadinya pengiriman dan penerimaan paket pada jaringan (Disastra, 2010). Dalam penerapannya pada kehidupan sehari-hari, *packet loss* akan mempengaruhi kualitas layanan yang menggunakan protokol UDP (*User Datagram Protocol*) dan ICMP (*Internet Control Mesege Protocol*), seperti contohnya adalah pada aplikasi VoIP (*Voice over Internet Protocol*) dan game online.

Berikut adalah tabel kategori degradasi terhadap tingkatan *packet loss* (Institut Teknologi Telkom, 2009).

Packet loss	Kategori degradasi
0%	Sangat bagus
3%	Bagus
15%	Sedang
25%	Jelek

Tabel 2.1. Kategori Degradasi Terhadap Tingkatan *Packet Loss* (Institut Teknologi Telkom, 2009)

Dari tabel diatas dapat diketahui bahwa semakin besar nilai *packet loss*, maka semakin buruk kinerja suatu jaringan.

Banyak faktor yang dapat menyebabkan terjadinya *packet loss*. Contohnya adalah penurunan sinyal pada media jaringan dan terjadinya kerusakan perangkat keras jaringan.

c. *Jitter*

Jitter dapat didefinisikan sebagai variasi dari *delay* atau variasi waktu kedatangan paket. Banyak hal yang dapat menyebabkan *jitter*, diantaranya adalah peningkatan *traffic* secara tiba-tiba sehingga menyebabkan penyempitan *bandwidth* dan menimbulkan antrian. Selain itu kecepatan terima dan kirim paket dari setiap *node* juga dapat menyebabkan *jitter* (Rizaldi & Setiawan, 2010).

Berikut adalah tabel tingkatan besarnya jitter (Institut Teknologi Telkom, 2009).

Peak jitter	Kategori degradasi
0 ms	Sangat bagus
75 ms	Bagus
125 ms	Sedang
225 ms	Jelek

Tabel 2.2. Kategori Degradasi Terhadap Tingkatan *Jitter* (Institut Teknologi Telkom, 2009)

Dari tabel diatas dapat diketahui bahwa semakin besar *jitter* maka semakin buruk kinerja dari suatu jaringan.

d. RTT (*Round Trip Time*)

RTT adalah waktu yang dibutuhkan oleh suatu paket data untuk menempuh perjalanan dari sumber menuju tujuan, kemudian kembali lagi ke sumbernya (Rouse, 2007). Pada konteks ini yang disebut sumber adalah komputer yang mengirimkan sinyal, sedangkan tujuan adalah *remote* komputer atau sistem yang menerima sinyal dan meneruskannya.

Dalam beberapa penerapannya, *latency* atau *delay* diukur dengan cara mengirimkan paket yang kembali lagi, dan RTT (*Round Trip Time*) yang dihasilkan diakui sebagai *latency* (Blair, 2006).

Berikut adalah standar toleransi untuk *delay* (International Telecommunication Union, 2008).

Range in millisecond	Description
0-150 msec	<i>Acceptable for most user application</i>
150-400 msec	<i>Acceptable provided that administrator are aware of the transmission time and its impact on transmission quality of user application</i>
>400 msec	<i>Unacceptable for general network planning purpose</i>

Tabel 2.3. Standar Toleransi Delay (International Telecommunication Union, 2008)

Semakin besar RTT atau *delay*, maka semakin buruk kinerja suatu jaringan komputer.

e. Reliabilitas

Selain dipengaruhi oleh ketersediaan atas fungsionalitas sistem pada sisi *user* atau sering disebut juga dengan *availability*, kinerja jaringan juga dipengaruhi oleh ketahanan sistem atau *reliability*.

Reliabilitas adalah kemampuan dari sebuah sistem untuk menghadapi adanya kesalahan tanpa mempengaruhi *user* atau jalannya operasi (IEEE Communication Society, 1998).

2.3 Load Balancing

Load balancing dapat diartikan sebagai suatu metode untuk menyebarkan beban kerja lalu lintas data secara seimbang melalui beberapa perantara untuk mengoptimalkan penggunaan sumber daya yang ada sehingga didapatkan kinerja yang lebih baik.

Load balancing pada mikrotik adalah teknik untuk mendistribusikan beban lalu lintas data pada dua atau lebih jalur koneksi secara seimbang, agar *traffic* dapat berjalan optimal, memaksimalkan *throughput*, memperkecil waktu tanggap dan menghindari *overload* pada salah satu jalur koneksi (Dewobroto, 2012).

a. ECMP (*Equal Cost Multi-Path*) *Load Balancing*

ECMP merupakan suatu teknik routing untuk mengatur rute paket melalui beberapa jalur yang mempunyai nilai sama. *Engine* yang bertugas mengirimkan paket mengidentifikasi jalur berdasarkan *hop* berikutnya (Hopps, 2000). Semua jalur diantara tiap-tiap node mempunyai nilai *routing* yang sama, sehingga lalu lintas data akan dibagi sama rata.

b. PCC (*Per Connection Classifier*) Load Balancing

PCC adalah salah satu metode load balancing pada router mikrotik yang memungkinkan pengguna untuk membagi lalu lintas data menjadi aliran data yang sama besar dan mampu menjaga paket dengan serangkaian aturan pada aliran data tertentu (MikroTik, 2008).

2.4 MikroTik RouterOS

Perangkat router Mikrotik RouterOS merupakan sebuah sistem operasi yang fiturnya sudah dioptimalisasi untuk menyediakan fungsi-fungsi sebuah router. Mikrotik routeros adalah versi mikrotik dalam bentuk perangkat lunak yang dapat diinstall pada komputer rumahan PC (*Personal Computer*) melalui CD (*Compact Disc*) (Linto Herlambang, Moch.; Catur L, Azis, 2008).

2.5 Virtual Machine

Pada masa sekarang dimana isu mengenai energi merupakan topik yang sensitif, teknologi virtualisasi menjadi salah satu solusi untuk mengatasi tingginya konsumsi energi pada perangkat *server* maupun *desktop*. *Virtual machine* merupakan terapan teknologi virtualisasi dimana *user* dapat membangun sistem komputer atau server secara virtual, sehingga mengurangi penggunaan perangkat fisik. Selain itu *virtual machine* juga dapat digunakan untuk melakukan simulasi kerja dari suatu perangkat fisik.

Definisi dari *virtual machine* adalah implementasi perangkat lunak dari sebuah mesin yang dapat berfungsi sebagai mana mesin yang sebenarnya. *Virtual machine* pada mulanya didefinisikan sebagai sebuah duplikat yang efisien dan terisolasi dari suatu mesin asli (Popek, Gerald J; Goldberg, Robert P, 1974).

BAB III

METODOLOGI

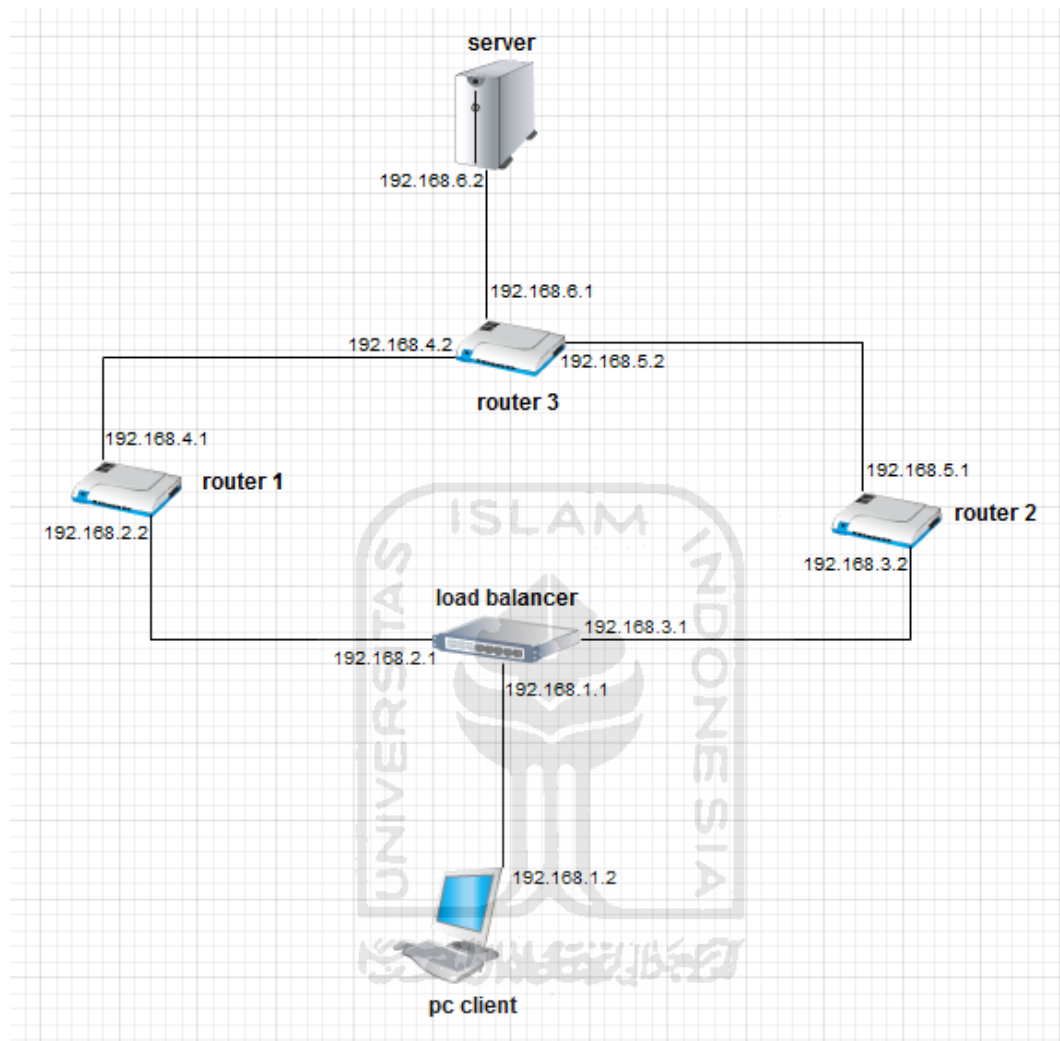
3.1. Perencanaan Skenario

Skenario dari penelitian ini yaitu akan dibangun suatu jaringan dimana *client* mempunyai dua buah *gateway*. Dua *gateway* tersebut mempunyai bandwidth yang berbeda. Agar kedua *gateway* tersebut dapat digunakan bersama maka perlu diterapkannya metode load balancing yang secara teori dapat menyeimbangkan beban lalu lintas data pada kedua *gateway* dan menyediakan fungsi fail over.

Pada kedua metode load balancing akan dilakukan aktivitas yang menggunakan protocol TCP (Transmission Control Protocol), UDP (*User Datagram Protocol*) maupun ICMP (*Internet Control Message Protocol*). Untuk melakukan hal ini akan digunakan Iperf yang mempunyai fungsi sebagai network packet generator dan juga analyzer, dan juga Ping serta Traceroute. Lalu lintas data yang melewati kedua *load balancer* akan di analisis hasilnya untuk dibandingkan satu sama lain. Parameter kinerja yang diuji adalah throughput, RTT, *jitter* dan *packet loss*.

Untuk menghemat sumber daya maka penelitian ini dilakukan dalam lingkungan *virtual machine*. Dalam *virtual machine* tersebut akan dibangun *server*, yang terkoneksi dengan *client* melalui beberapa router untuk mensimulasikan jaringan internet.

3.2. Perancangan Topologi



Gambar 3.1. Struktur Rancangan Topologi Jaringan

Seperti yang sudah dijelaskan dalam perancangan skenario, pada penelitian ini akan dibangun jaringan komputer di dalam *virtual machine* yang terdiri dari satu *client*, satu *load balancer*, dua *gateway*, satu router dan satu *server*. pada gambar 3.1, router 1 dan router 2 berfungsi sebagai *gateway 1* dan *gateway 2*. Pada *load balancer* akan diterapkan metode PCC dan ECMP secara bergantian. *Load balancer* menyeimbangkan lalu lintas data dari kedua *gateway* menuju komputer *client*. Router 3 menghubungkan *gateway* ke *server*.

3.3. Analisis Kebutuhan

Dari sisi perangkat keras, penelitian ini membutuhkan perangkat komputer yang mempunyai spesifikasi cukup untuk menjadi *host* yang menjalankan beberapa komputer *guest* dalam *virtual machine* yang akan berfungsi layaknya jaringan komputer yang sebenarnya. Dimana dalam jaringan virtual tersebut terdapat satu buah komputer *client*, satu buah PC (*Personal computer*) yang difungsikan sebagai *load balancer*, tiga buah PC yang difungsikan sebagai router sebagai perumpamaan jaringan internet dan satu buah *server*.

Sedangkan perangkat lunak yang digunakan dalam penelitian ini yang paling utama adalah VMware Workstation yang akan diinstal pada komputer *host* sebagai *virtual machine*. Pada komputer *host* juga diinstall winbox yang berfungsi mengkonfigurasi mikrotik routeros dengan lebih mudah.

Pada jaringan komputer yang dibangun dalam *virtual machine* tersebut, komputer *client* menggunakan Linux Mint sebagai sistem operasinya. Pada komputer *client* akan diinstall aplikasi Iperf dan Jperf yang berfungsi mengirimkan koneksi TCP maupun UDP ke *server* dan sekaligus dapat digunakan untuk mengukur nilai *Throughput*, *packet loss* dan *jitter*. Kemudian untuk aplikasi Ping dan Traceroute sudah terinstall secara *default* pada sistem operasi Linux Mint. Aplikasi ping tersebut dimanfaatkan untuk mengukur respons

Load balancer dan router menggunakan MikroTik RouterOS yang diinstallkan pada PC sehingga berfungsi layaknya *load balancer* dan router yang sebenarnya.

Kemudian untuk *server* juga menggunakan Linux Mint sebagai sistem operasinya. Tool Iperf ini juga harus diinstall pada komputer *server* ini, karena berfungsi menangkap sinyal yang dikirimkan Iperf dari sisi *client*. Demikianlah skenario dari penelitian yang akan penulis lakukan.

Hardware	Software
<i>Host Computer</i>	VMware Workstation, winbox
<i>PC client (guest)</i>	Linux Mint, Iperf, Jperf, Ping, Traceroute
<i>Load balancer (guest)</i>	MikroTik RouterOS
<i>Router (guest)</i>	Mikrotik RouterOS
<i>Server (guest)</i>	Linux Mint, Iperf

Tabel 3.1. Daftar Kebutuhan Perangkat Keras dan Perangkat Lunak

3.4. Implementasi

Setelah rancangan dari topologi jaringan dan semua kebutuhan sistem telah tersedia, maka langkah selanjutnya adalah implementasi dari penelitian ini.

Berikut ini adalah langkah-langkah implementasi pembangunan jaringan yang sesuai dengan topologi yang telah dirancang :

a. Instalasi *virtual machine*

Virtual machine diinstal pada komputer *host* yang digunakan untuk melakukan penelitian.

b. Persiapan jaringan komputer pada *virtual machine*

Persiapan ini meliputi:

- 1) Instalasi sistem operasi masing-masing komputer baik *client*, *load balancer*, router maupun *server*,
- 2) Instalasi sistem MikroTik RouterOS pada *load balancer* dan pada router,
- 3) Serta konfigurasi jaringan komputer sampai semua sistem saling terhubung.

c. Pengaturan konfigurasi *load balancing* PCC dan ECMP

Pada tahap ini akan dilakukan konfigurasi pada masing-masing *load balancer* .

d. Pengujian

Pengujian dilakukan dengan berbagai cara melakukan aktivitas *request* dari *client* ke *server* melalui protocol TCP maupun UDP menggunakan aplikasi *network analyzer* yang telah ditentukan.

Setiap parameter *quality of service* diuji satu demi satu.

1. Throughput

Pengujian throughput dilakukan dengan menggunakan tool iperf dengan modus TCP yang dilakukan selama 30 (tiga puluh) detik dan dilaporkan setiap detik.

2. Packet loss

Pengujian packet loss dilakukan dengan menggunakan tool iperf dengan modus UDP, dengan cara melakukan pengujian sebanyak 6 (enam) kali kemudian diambil rata-rata packet loss nya.

3. Jitter

Pengujian jitter dilakukan dengan menggunakan tool iperf dengan modus UDP, dengan cara melakukan pengujian sebanyak 6 (enam) kali kemudian diambil rata-rata jitter nya..

4. RTT

Pengujian RTT dilakukan dengan menggunakan tool ping, dengan cara melakukan pengujian sebanyak 6 (enam) kali kemudian diambil rata-rata RTT nya..

5. Reliability

Pengujian reliability dilakukan dengan menggunakan tool traceroute. Pada saat pengujian reliability jaringan, akan dilakukan pemutusan koneksi pada salah satu dari dua gateway yang ada. Kemudian akan dipantau bagaimana load balancer menghadapi kondisi tersebut.

e. Analisis hasil pengujian

Analisis hasil pengujian dilakukan dengan cara membandingkan data-data yang dihasilkan kedua metode load balancing pada saat pengujian sebelumnya pada setiap parameter kualitas layanan.

BAB IV

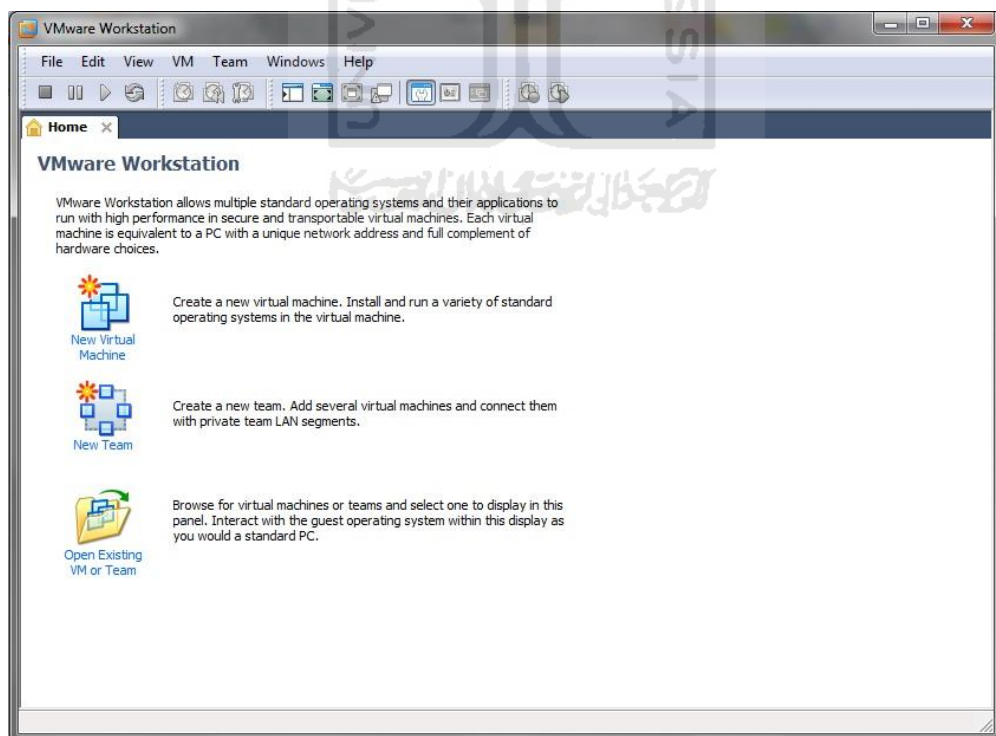
HASIL DAN PEMBAHASAN

4.1. Konfigurasi Virtual Machine

Seperti yang sudah dijelaskan pada batasan masalah, agar lebih mudah dalam pelaksanaan dan pengambilan data, maka penelitian ini dilakukan di dalam virtual machine. Jadi perangkat keras yang dibutuhkan hanyalah satu buah komputer yang mempunyai spesifikasi yang cukup untuk menjalankan beberapa guest operating system.

VMware mendukung fungsi-fungsi jaringan sehingga kita dapat membangun jaringan komputer virtual di dalam VMware.

Virtual machine yang digunakan dalam penelitian ini adalah VMware Workstation versi 7. Berikut adalah tampilan antar muka dari VMware Workstation setelah proses instalasi selesai.

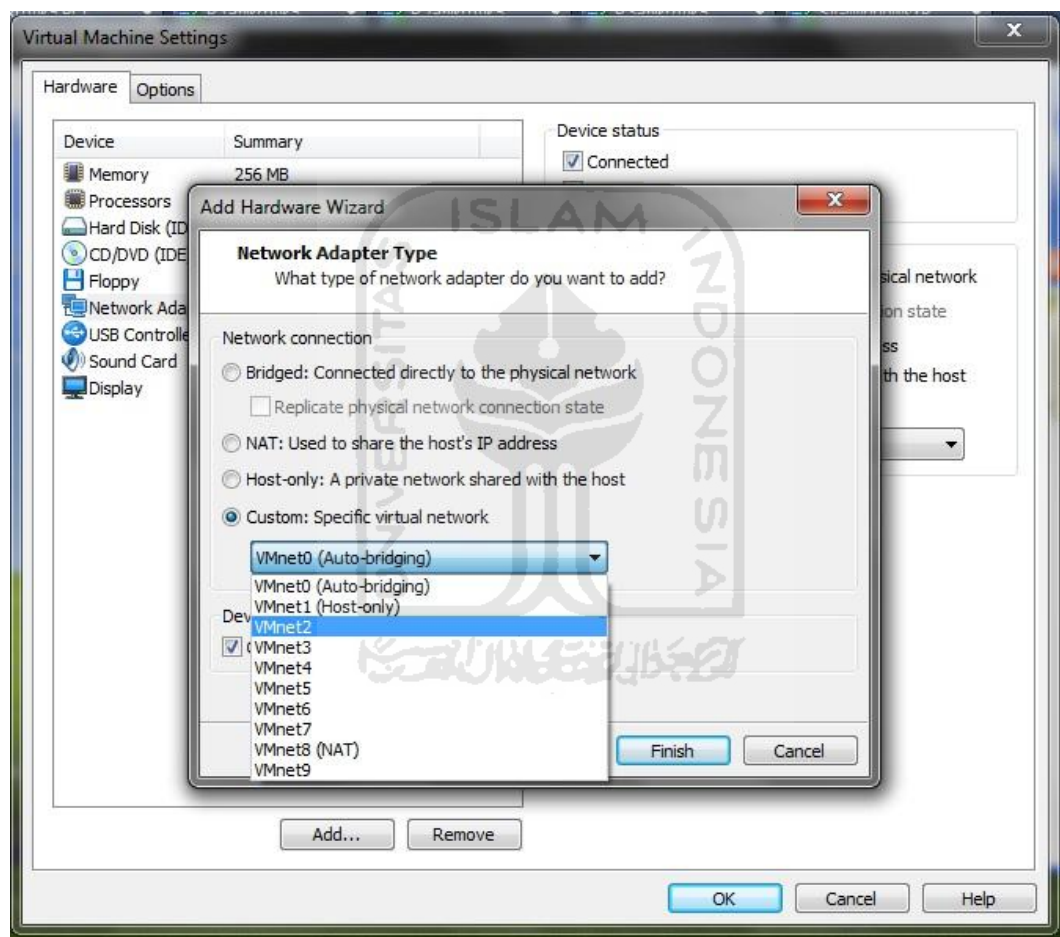


Gambar 4.1. Tampilan Antarmuka VMware Workstation

Untuk melakukan penambahan interface jaringan pada VMware, dapat dilakukan dengan memilih menu **VM**, kemudian pilih sub menu **setting**.

Kemudian pada tab **hardware** tekan tombol **add...** untuk menampilkan antarmuka *add hardware wizard*. Pada antarmuka *hardware type* pilih **network adapter** dan tekan tombol **next**.

Setelah itu akan muncul jendela *Network Adapter Type*. Pilih **Custom: Specific virtual network**.



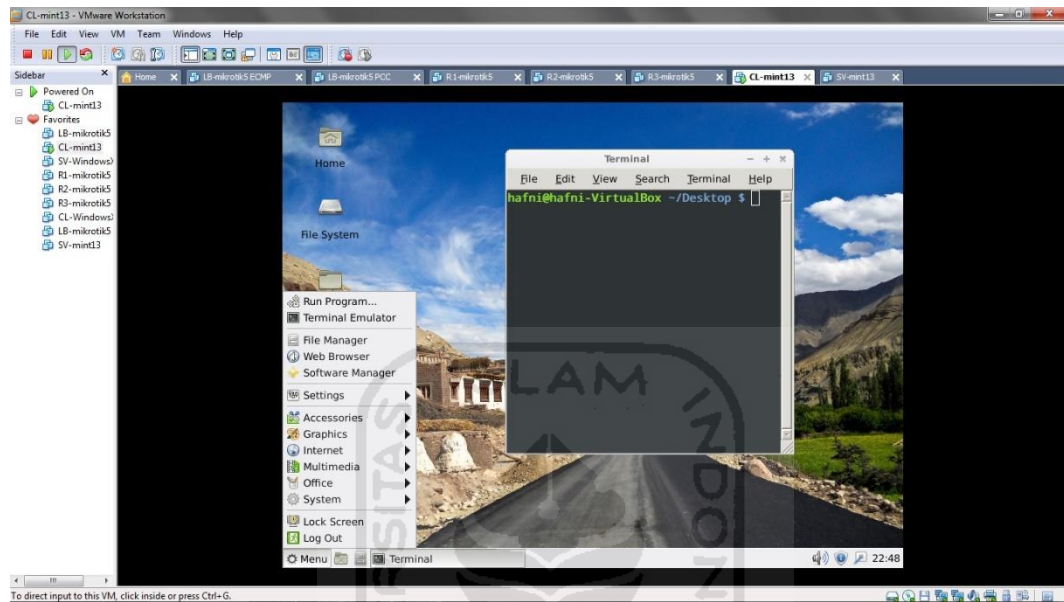
Gambar 4.2. Tampilan add hardware wizard pada VMware

4.2. Pembangunan Jaringan Komputer

Setelah Virtual Machine terinstal, langkah selanjutnya adalah pembangunan jaringan komputer di dalam virtual machine.

a. Instalasi Linux Mint

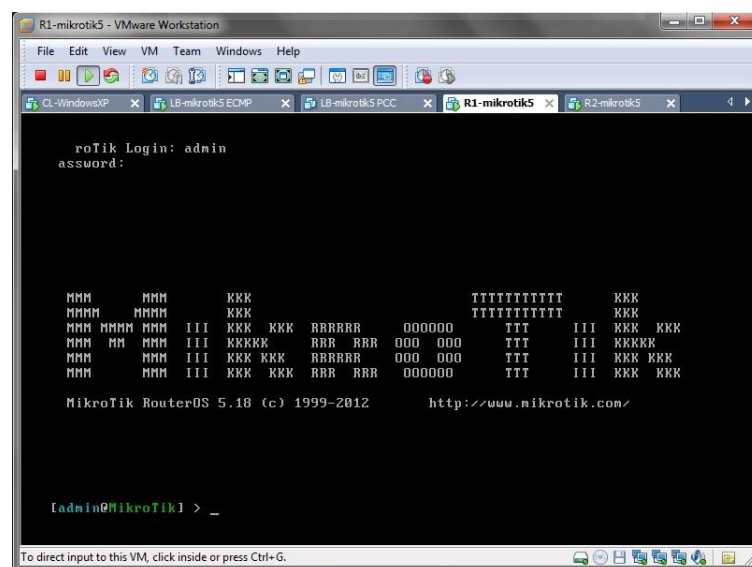
Sistem operasi Linux Mint akan digunakan pada komputer client dan server. Berikut adalah tampilan antar muka Linux Mint setelah selesai instalasi pada virtual machine.



Gambar 4.3. Tampilan Linux Mint pada Virtual Machine

b. Instalasi mikrotik routers

Sistem operasi mikrotik routeros akan digunakan pada load balancer dan semua router. Berikut adalah tampilan antar muka mikrotik routeros.



Gambar 4.4. Tampilan mikrotik routeros pada Virtual Machine

c. Konfigurasi jaringan komputer

Setelah melakukan instalasi sistem operasi dengan benar, hal pertama yang harus dilakukan adalah menentukan jumlah interface jaringan pada setiap komponen jaringan. Kemudian dilakukan konfigurasi alamat IP bagi semua komputer. Setelah itu kita harus melakukan pengaturan pada VMware agar setiap komponen jaringan dapat terhubung satu sama lain.

Berikut ini adalah tabel interface jaringan, alamat IP dan konfigurasi pada VMware.

Hardware	Interface name	IP address	VM Network Connection Type
PC Client	LAN 2	192.168.1.2	Custom: VMnet2
Load Balancer	lb_ke_cl	192.168.1.1	Custom: VMnet2
Load balancer	lb_ke_r1	192.168.2.1	Custom: VMnet3
Load balancer	lb_ke_r2	192.168.3.1	Custom: VMnet4
Router 1	R1_ke_lb	192.168.2.2	Custom: VMnet3
Router 1	R1_ke_r3	192.168.4.1	Custom: VMnet5
Router 2	R2_ke_lb	192.168.3.2	Custom: VMnet4
Router 2	R2_ke_r3	192.168.5.1	Custom: VMnet6
Router 3	R3_ke_r1	192.168.4.2	Custom: VMnet5
Router 3	R3_ke_r2	192.168.5.2	Custom: VMnet6
Router 3	R3_ke_sv	192.168.6.1	Custom: VMnet7
PC server	LAN 2	192.168.6.2	Custom: VMnet7

Tabel 4.1. Interface jaringan, alamat IP dan konfigurasi pada VMware

d. Konfigurasi Router1

Berikut ini adalah tampilan dari perintah-perintah *command line* yang kita inputkan pada MikroTik RouterOS yang berfungsi sebagai router1.

```
/interface ethernet
set 0 name=r1_ke_lb
set 1 name=r1_ke_r3
```

Perintah diatas adalah perintah untuk menentukan *interface* jaringan yang terpasang pada router1 beserta penamaannya. Router1 mempunyai 2 (dua) *interface* yang menghubungkan ke *load balancer* dan ke router3.

```
/ip address
add address=192.168.2.2/24 interface=r1_ke_lb
add address=192.168.4.1/24 interface=r1_ke_r3
```

Perintah diatas adalah perintah untuk menentukan *ip address* jaringan yang terpasang pada masing-masing *interface* jaringan pada router1

```
/routing rip interface
add interface=r1_ke_r3 receive=v1 send=v1
add interface=r1_ke_lb receive=v1 send=v1

/routing rip network
add network=192.168.4.0/24
add network=192.168.2.0/24
```

Perintah diatas adalah perintah untuk konfigurasi RIP (*Routing Information Protocol*) yang merupakan *dynamic routing* pada router1, agar router1 dapat dikenali dan saling berkomunikasi dengan *router* yang lain.

```
/queue simple
Add max-limit=1M/1M name=limit-r1 target-addresses=192.168.2.0/24
```

Perintah diatas adalah perintah untuk membatasi *bandwidth* yang melalui router1.

e. Konfigurasi pada router2

Berikut ini adalah tampilan dari perintah-perintah *command line* yang kita inputkan pada MikroTik RouterOS yang berfungsi sebagai router2.

```
/interface ethernet
set 0 name=r2_ke_lb
set 1 name=r2_ke_r3
```

Perintah diatas adalah perintah untuk menentukan interface jaringan yang terpasang pada router2 beserta penamaannya. Seperti router1, Router2 mempunyai 2 interface jaringan yang menghubungkan dengan load balancer dan router3.

```
/ip address
add address=192.168.3.2/24 interface=r2_ke_lb
add address=192.168.5.1/24 interface=r2_ke_r3
```

Perintah diatas adalah perintah untuk menentukan *ip address* jaringan yang terpasang pada masing-masing *interface* jaringan pada router2.

```
/routing rip interface
add interface=r2_ke_r3 receive=v1 send=v1
add interface=r2_ke_lb receive=v1 send=v1

/routing rip network
add network=192.168.3.0/24
add network=192.168.5.0/24
```

Perintah diatas adalah perintah untuk konfigurasi RIP (*Routing Information Protocol*) yang merupakan *dynamic routing* pada router2, agar router2 dapat dikenali dan saling berkomunikasi dengan *router* yang lain.

```
/queue simple
add max-limit=2M/2M name=limit-r1 target-addresses=192.168.2.0/24
```

Perintah diatas adalah perintah untuk membatasi *bandwidth* yang melalui router2.

f. Konfigurasi pada router3

Berikut ini adalah tampilan dari perintah-perintah *command line* yang kita inputkan pada MikroTik RouterOS yang berfungsi sebagai router3.

```
/interface ethernet
set 0 name=r3_ke_r1
set 1 name=r3_ke_r2
set 2 name=r3_ke_sv
```

Perintah diatas adalah perintah untuk menentukan interface jaringan yang terpasang pada router3 beserta penamaannya. Router3 mempunyai 3 (tiga) interface jaringan. Masing-masing menghubungkan ke router1, router2 dan server.

```
/ip address
add address=192.168.4.2/24 interface=r3_ke_r1
add address=192.168.5.2/24 interface=r3_ke_r2
add address=192.168.6.1/24 interface=r3_ke_sv
```

Perintah diatas adalah perintah untuk menentukan *ip address* jaringan yang terpasang pada masing-masing *interface* jaringan pada router3.

```
/routing rip interface
add interface=r3_ke_r1 receive=v1 send=v1
add interface=r3_ke_r2 receive=v1 send=v1
add interface=r3_ke_sv receive=v1 send=v1

/routing rip network
add network=192.168.6.0/24
add network=192.168.4.0/24
add network=192.168.5.0/24
```

Perintah diatas adalah perintah untuk konfigurasi RIP (*Routing Information Protocol*) yang merupakan *dynamic routing* pada router3, agar router2 dapat dikenali dan saling berkomunikasi dengan *router* yang lain.

g. Konfigurasi pada load balancer metode ECMP

Berikut ini adalah tampilan dari perintah-perintah *command line* yang kita inputkan pada Mikrotik RouterOS yang berfungsi sebagai load balancer metode ECMP.

```
/interface ethernet
set 0 name=lb_ke_cl
set 1 name=lb_ke_r1
set 2 name=lb_ke_r2
```

Perintah diatas adalah perintah untuk menentukan interface jaringan yang terpasang pada load balancer beserta penamaannya. Load balancer mempunyai 3

(tiga) interface jaringan. Masing-masing menghubungkan ke router1, router2 dan client.

```
/ip address
add address=192.168.1.1/24 interface=lb_ke_c1
add address=192.168.2.1/24 interface=lb_ke_r1
add address=192.168.3.1/24 interface=lb_ke_r2
```

Perintah diatas adalah perintah untuk menentukan *ip address* jaringan yang terpasang pada masing-masing *interface* jaringan pada load balancer.

```
/ip firewall nat
add action=masquerade chain=srcnat out-interface=lb_ke_r1
add action=masquerade chain=srcnat out-interface=lb_ke_r2

/ip route
add dst-address=0.0.0.0/0 gateway=192.168.2.2,192.168.3.2, 192.168.3.2 check-gateway=ping

/ip firewall mangle
add chain=input action=mark-connection in-interface=lb_ke_r1 new-connection-mark=r1-conn
add chain=input action=mark-connection in-interface=lb_ke_r2 new-connection-mark=r2-conn
add chain=output action=mark-routing connection-mark=r1-conn new-routing-mark=to-r1
add chain=output action=mark-routing connection-mark=r1-conn new-routing-mark=to-r2

/ip route
add dst-address=0.0.0.0/0 gateway=192.168.2.2 routing-mark=to-r1
add dst-address=0.0.0.0/0 gateway=192.168.3.2 routing-mark=to-r2
```

Perintah-perintah diatas adalah kumpulan konfigurasi untuk *load balancing* metode ECMP.

h. Konfigurasi pada load balancer metode PCC

Berikut ini adalah tampilan dari perintah-perintah *command line* yang kita inputkan pada MikroTik RouterOS yang berfungsi sebagai load balancer metode PCC.

```

/interface ethernet
set 0 name=lb_ke_cl
set 1 name=lb_ke_r1
set 2 name=lb_ke_r2

```

Perintah diatas adalah perintah untuk menentukan interface jaringan yang terpasang pada load balancer beserta penamaannya. Load balancer mempunyai 3 (tiga) interface jaringan. Masing-masing menghubungkan ke router1, router2 dan client.

```

/ip address
add address=192.168.1.1/24 interface=lb_ke_cl
add address=192.168.2.1/24 interface=lb_ke_r1
add address=192.168.3.1/24 interface=lb_ke_r2

```

Perintah diatas adalah perintah untuk menentukan *ip address* jaringan yang terpasang pada masing-masing *interface* jaringan pada load balancer.

```

/ip firewall mangle
add chain=prerouting dst-address=192.168.2.0/24 in-interface=lb_ke_cl
add chain=prerouting dst-address=192.168.3.0/24 in-interface=lb_ke_r1
add action=mark-connection chain=prerouting connection-mark=no-mark in-
interface=lb_ke_r1 new-connection-mark=r1-conn
add action=mark-connection chain=prerouting connection-mark=no-mark in-
interface=lb_ke_r2 new-connection-mark=r2-conn
add action=mark-connection chain=prerouting connection-mark=no-mark dst-
address-type=!local in-interface=lb_ke_cl new-connection-mark=r1-conn per-
connection-classifier=both-addresses:2/0
add action=mark-connection chain=prerouting connection-mark=no-mark in-
interface=lb_ke_cl new-connection-mark=r2-conn per-connection-classifier=both-
addresses:2/1 src-address-type=!local
add action=mark-routing chain=prerouting connection-mark=r1-conn in-
interface=lb_ke_cl new-routing-mark=ke-r1
add action=mark-routing chain=prerouting connection-mark=r2-conn in-
interface=lb_ke_cl new-routing-mark=ke-r2
add action=mark-routing chain=output connection-mark=r1-conn new-routing-
mark=ke-r1
add action=mark-routing chain=output connection-mark=r2-conn new-routing-
mark=ke-r2

/ip route
add check-gateway=ping distance=1 gateway=192.168.2.2 routing-mark=ke-r1
add check-gateway=ping distance=1 gateway=192.168.3.2 routing-mark=ke-r2

```

```

add check-gateway=ping distance=1 gateway=192.168.2.2
add check-gateway=ping distance=2 gateway=192.168.3.2

/ip firewall nat
add action=masquerade chain=srcnat out-interface=lb_ke_r1
add action=masquerade chain=srcnat out-interface=lb_ke_r2

```

Perintah-perintah diatas adalah kumpulan konfigurasi untuk *load balancing* metode PCC.

i. Konfigurasi Network Analyzer

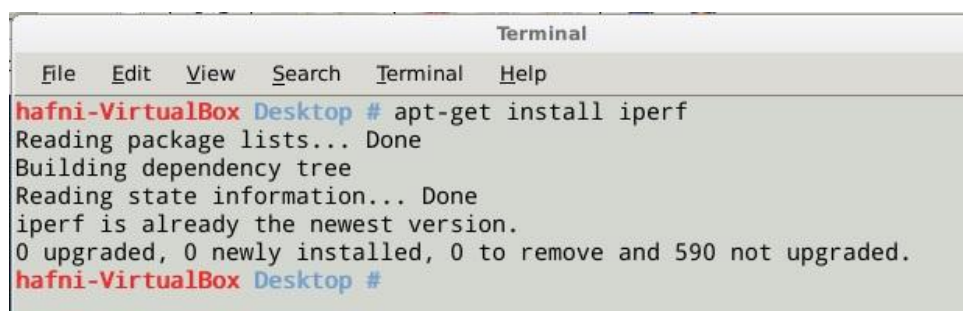
Tool yang digunakan untuk menganalisis lalu lintas data pada jaringan yang digunakan untuk penelitian ini adalah Iperf. Iperf dapat mengirimkan aliran data menggunakan protocol TCP maupun UDP, kemudian mengukur besarnya throughput yang ada. Iperf dipasang pada client dan server, dan dijalankan berbasis text command prompt.

Iperf dapat digunakan bersama-sama dengan Jperf. Jperf adalah versi GUI (graphical user interface) dari Iperf yang berbasis Java. Oleh karena itu sistem operasi dari komputer yang menggunakannya harus sudah terinstall JRE(java runtime environment) terlebih dahulu.

Untuk melakukan instalasi Iperf pada Linux Mint, baik di komputer client maupun server, adalah dengan mengetikkan perintah berikut pada terminal.

```
Sudo apt-get install Iperf
```

Berikut ini adalah tampilan dari proses instalasi Iperf pada Linux Mint menggunakan perintah seperti diatas.



```

Terminal
File Edit View Search Terminal Help
hafni-VirtualBox Desktop # apt-get install iperf
Reading package lists... Done
Building dependency tree
Reading state information... Done
iperf is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 590 not upgraded.
hafni-VirtualBox Desktop #

```

Gambar 4.5. Proses Instalasi Iperf Pada Linux Mint

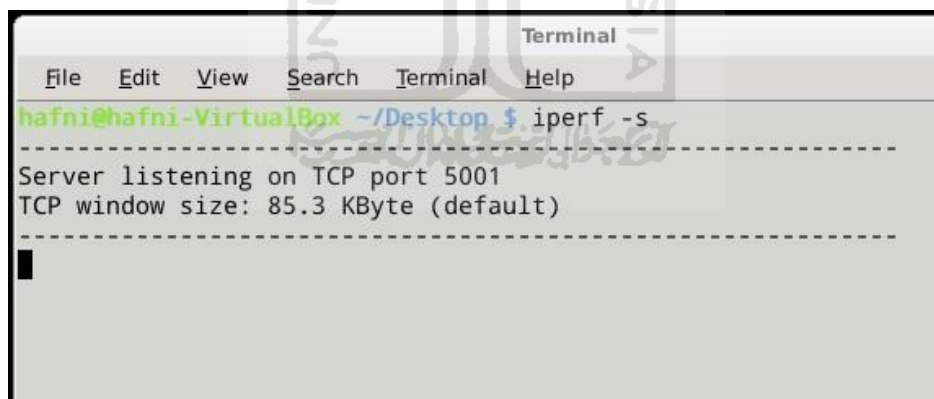
Sedangkan untuk menggunakan Jperf, seperti yang sudah disebutkan sebelumnya, kita harus menginstall JRE pada komputer terlebih dahulu. Setelah JRE terinstall, yang harus dilakukan adalah mendownload aplikasi Jperf dan mengekstraknya pada direktori yang diinginkan. Kemudian cari dan temukan file ‘**jperf.sh**’, ganti permission dari file tersebut agar bisa dieksekusi.

4.3. Pengujian

Pengujian akan dilakukan untuk mendapatkan nilai throughput, delay, jitter dan packet loss. Pengujian akan dilakukan pada kedua load balancer satu demi satu.

- a. Pengukuran nilai throughput dilakukan menggunakan protocol TCP pada Iperf. Berikut ini adalah langkah-langkahnya:
 1. Menjalankan iperf pada sisi server agar bisa menangkap paket yang dikirimkan dari client.

```
Iperf -s
```



```

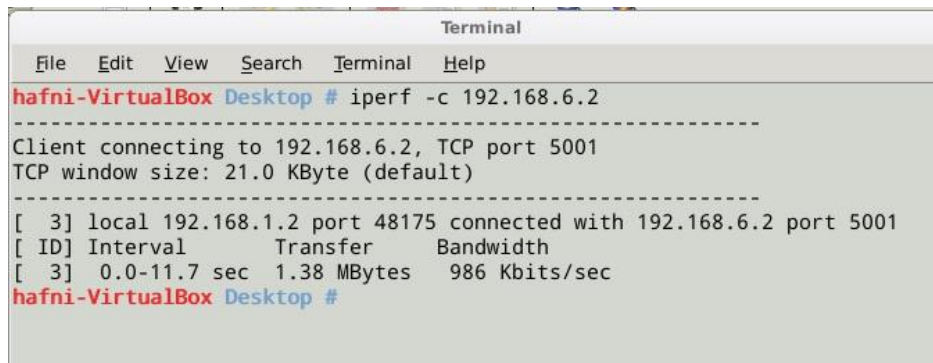
Terminal
File Edit View Search Terminal Help
hafni@hafni-VirtualBox ~/Desktop $ iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----

```

Gambar 4.6. Iperf Pada Server Mendengarkan Koneksi Protocol TCP

2. Menjalankan iperf dari client untuk mengirimkan paket menggunakan protokol TCP ke server dengan alamat 192.168.6.2, dan memantau throughputnya.

```
Iperf -c 192.168.6.2
```



```

Terminal
File Edit View Search Terminal Help
hafni-VirtualBox Desktop # iperf -c 192.168.6.2
-----
Client connecting to 192.168.6.2, TCP port 5001
TCP window size: 21.0 KByte (default)
-----
[ 3] local 192.168.1.2 port 48175 connected with 192.168.6.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0-11.7 sec  1.38 MBytes  986 Kbits/sec
hafni-VirtualBox Desktop #

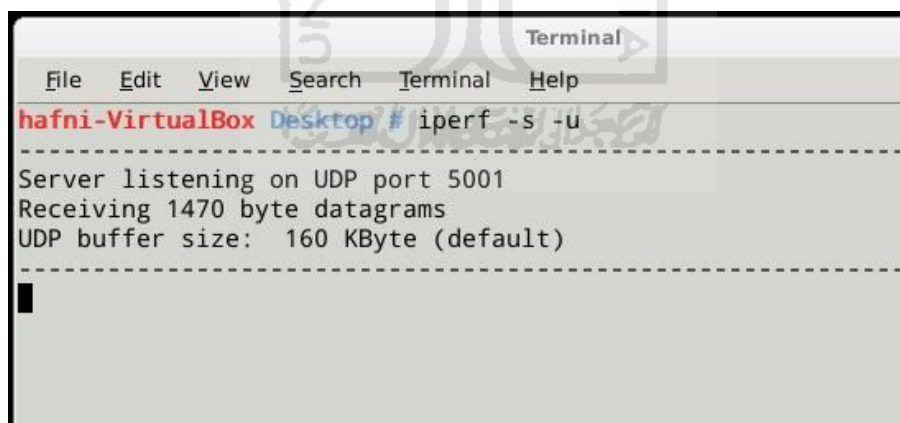
```

Gambar 4.7. Iperf Pada Client Mengirimkan Koneksi Protocol TCP

Dapat dilihat dari gambar bahwa throughput langsung ditampilkan setelah pengujian.

- b. Pengukuran nilai packet loss dan jitter menggunakan iperf pada protokol UDP.
 1. Mengatur iperf pada sisi server agar mendengarkan koneksi pada protokol UDP

```
Iperf -s -u
```



```

Terminal
File Edit View Search Terminal Help
hafni-VirtualBox Desktop # iperf -s -u
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 160 KByte (default)
-----
█

```

Gambar 4.8. Iperf Pada Server Mendengarkan Koneksi Protocol UDP

2. Menjalankan iperf dari client untuk mengirimkan paket menggunakan protokol UDP ke server dengan alamat 192.168.6.2, dan memantau jitter dan packet lossnya.

```
Iperf -c 192.168.6.2 -u
```

```

Terminal
File Edit View Search Terminal Help
hafni-VirtualBox Desktop # iperf -c 192.168.6.2 -u
-----
Client connecting to 192.168.6.2, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 160 KByte (default)
-----
[ 3] local 192.168.1.2 port 36007 connected with 192.168.6.2 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0-10.0 sec  1.25 MBytes 1.05 Mbits/sec
[ 3] Sent 893 datagrams
[ 3] Server Report:
[ 3] 0.0-10.1 sec  1.19 MBytes 985 Kbits/sec 1.927 ms 45/ 893 (5%)
hafni-VirtualBox Desktop #

```

Gambar 4.9. Iperf Pada Server Mengirimkan Koneksi Protocol UDP

Dari gambar diatas dapat dilihat pada dua kolom terakhir baris terakhir tampilan terminal adalah jitter dan packet loss.

c. Mengukur RTT (Round Trip Time) menggunakan Ping

Dengan menggunakan tool ping dari sisi client yang sudah terinstall pada Linux Mint kita dapat melihat besarnya RTT.

Ping 192.168.6.2

```

Terminal
File Edit View Search Terminal Help
hafni-VirtualBox Desktop # ping 192.168.6.2 -c 15
PING 192.168.6.2 (192.168.6.2) 56(84) bytes of data:
64 bytes from 192.168.6.2: icmp_req=1 ttl=61 time=3.89 ms
64 bytes from 192.168.6.2: icmp_req=2 ttl=61 time=5.37 ms
64 bytes from 192.168.6.2: icmp_req=3 ttl=61 time=1.97 ms
64 bytes from 192.168.6.2: icmp_req=4 ttl=61 time=3.06 ms
64 bytes from 192.168.6.2: icmp_req=5 ttl=61 time=3.03 ms
64 bytes from 192.168.6.2: icmp_req=6 ttl=61 time=2.98 ms
64 bytes from 192.168.6.2: icmp_req=7 ttl=61 time=2.24 ms
64 bytes from 192.168.6.2: icmp_req=8 ttl=61 time=9.13 ms
64 bytes from 192.168.6.2: icmp_req=9 ttl=61 time=3.63 ms
64 bytes from 192.168.6.2: icmp_req=10 ttl=61 time=2.41 ms
64 bytes from 192.168.6.2: icmp_req=11 ttl=61 time=4.22 ms
64 bytes from 192.168.6.2: icmp_req=12 ttl=61 time=3.36 ms
64 bytes from 192.168.6.2: icmp_req=13 ttl=61 time=3.24 ms
64 bytes from 192.168.6.2: icmp_req=14 ttl=61 time=2.99 ms
64 bytes from 192.168.6.2: icmp_req=15 ttl=61 time=3.38 ms

--- 192.168.6.2 ping statistics ---
15 packets transmitted, 15 received, 0% packet loss, time 14045ms
rtt min/avg/max/mdev = 1.971/3.664/9.137/1.669 ms
hafni-VirtualBox Desktop #

```

Gambar 4.10. Ping Dari Client ke Server

Dari gambar diatas dapat dilihat bahwa tool ping dapat mengukur packet loss dan RTT.

d. Mengukur reliabilitas jaringan menggunakan Traceroute

Traceroute merupakan aplikasi bawaan linux mint yang biasa digunakan untuk mengetahui rute paket dari sumber ke tujuan. Traceroute dapat dijalankan menggunakan perintah:

```
traceroute 192.168.6.2
```

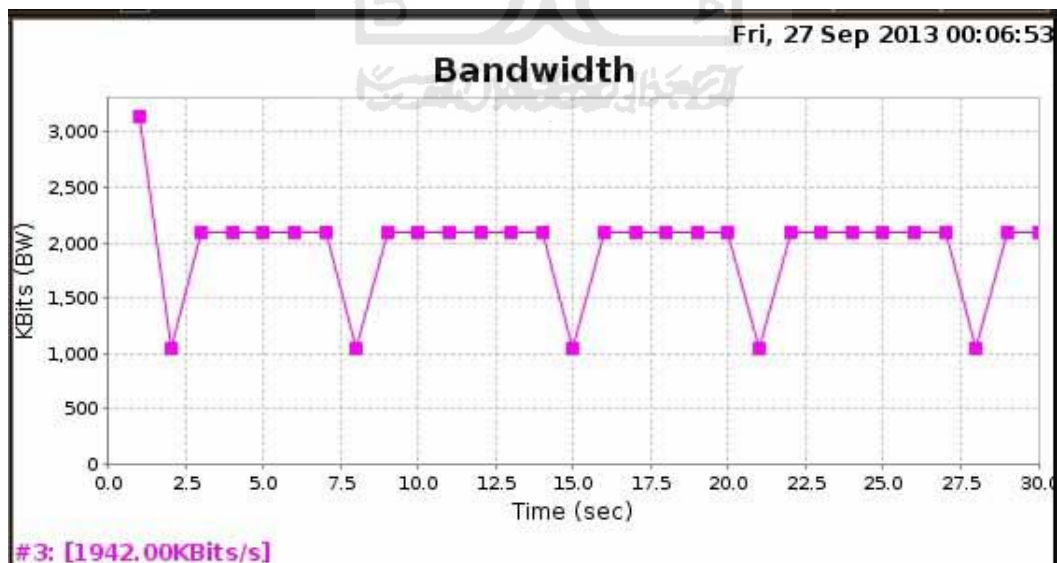
4.4. Hasil Pengujian

Dari pengujian yang sudah dilakukan telah didapatkan hasil sebagai berikut

a. Pengukuran nilai throughput

Berikut ini adalah hasil pengukuran nilai throughput pada masing-masing load balancing, dimana pengujian dilakukan selama tiga puluh detik dan dilaporkan setiap detik.

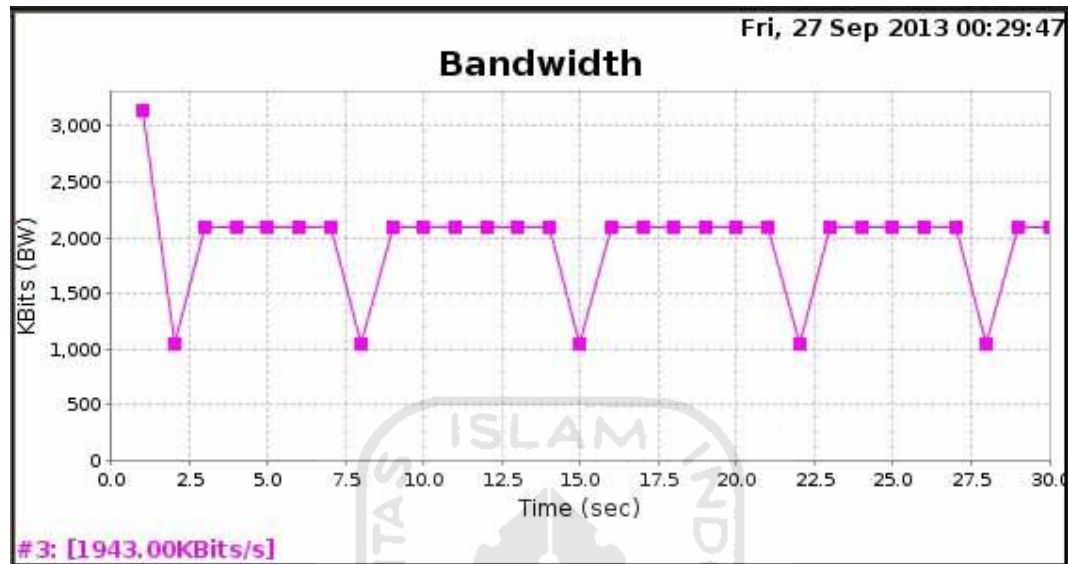
1. Throughput pada metode ECMP



Gambar 4.11. Grafik Hasil Throughput pada ECMP

Dari grafik diatas dapat diketahui bahwa throughput berkisar antara 2000 Kbits/s sampai dengan 1000 Kbits/s, dengan rata-rata **1942 Kbits/s**.

2. Throughput pada metode PCC



Gambar 4.12. Grafik Hasil Throughput pada PCC

Sedangkan pada metode PCC, meskipun kisaran throughput hampir sama dengan pada metode ECMP, namun rata-rata throughput yang dihasilkan ternyata sedikit lebih tinggi yaitu pada **1943 Kbits/s**.

b. Pengukuran nilai jitter.

1. Pengukuran pada metode ECMP

Tes ke nilai	I	II	III	IV	V	VI	Rata-rata
Jitter (ms)	2,039	1,929	1,827	2,584	2,298	1,948	2.104

Tabel 4.2. Hasil Pengukuran Nilai Jitter Metode ECMP

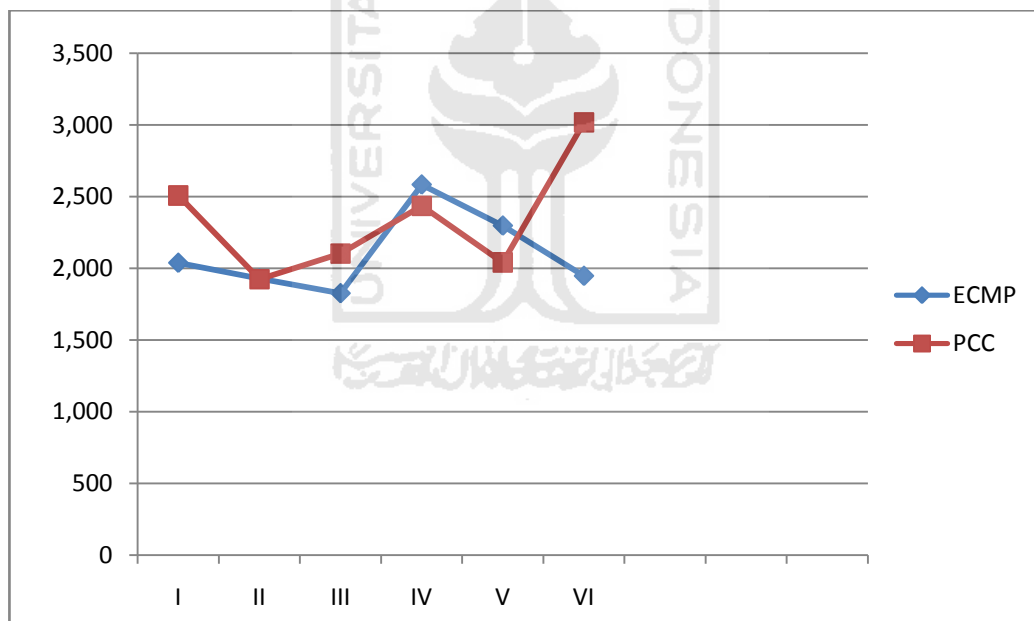
Pengujian dilakukan sebanyak 6 (enam) kali. Kemudian dicari nilai rata-rata jitter yang dihasilkan. Rata-rata yang didapatkan dari pengujian jitter adalah **2,104 ms**.

2. Pengukuran pada metode PCC

Tes ke nilai	I	II	III	IV	V	VI	Rata-rata
Jitter (ms)	2,509	1,925	2,104	2,437	2,042	3,019	2,339

Tabel 4.3. Hasil Pengukuran Nilai Jitter Metode PCC

Setelah dilakukan pada metode ECMP, selanjutnya adalah melakukan pengujian dengan cara yang sama pada metode PCC. Yaitu dengan cara melakukan 6 (enam) kali pengujian kemudian dicari rata-ratanya. Dari pengujian yang dilakukan didapatkan rata-rata jitter yang dihasilkan adalah **2,339 ms**.



Gambar 4.13. Grafik Jitter pada ECMP dan PCC

Gambar diatas adalah hasil ilustrasi nilai pengujian yang dituangkan dalam bentuk grafik.

c. Pengukuran nilai packet loss

1. Pada metode ECMP

Tes ke nilai	I	II	III	IV	V	VI	Rata-rata
Pkt. Loss (%)	0,47	0,47	0,59	0,47	0,47	0,47	0.49

Tabel 4.4. Hasil Pengukuran Nilai Packet Loss Metode ECMP

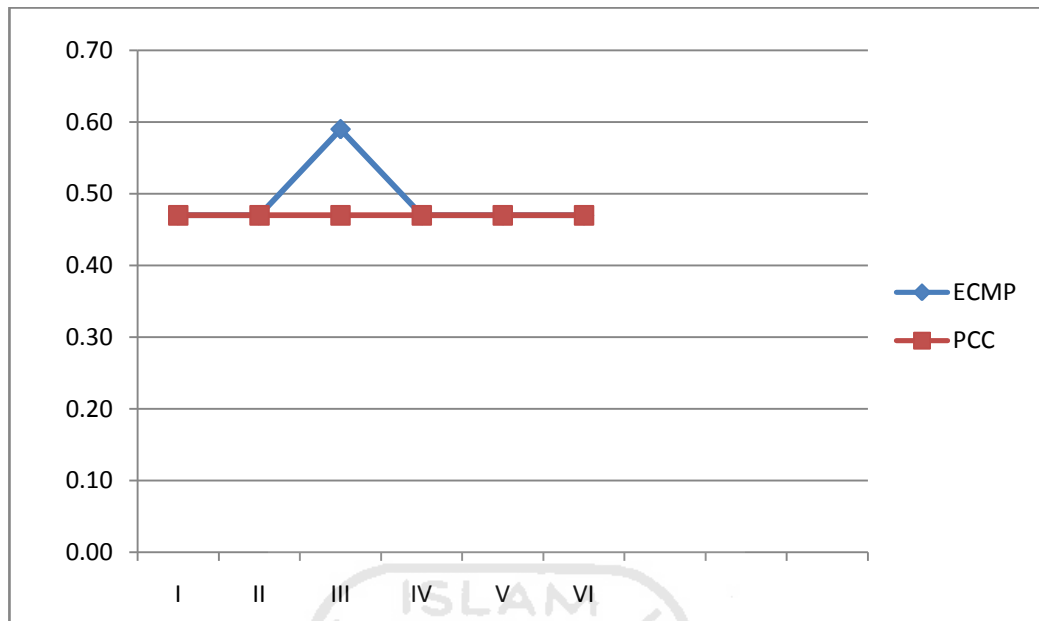
Dari percobaan didapatkan rata-rata packet loss pada jaringan yang menggunakan metode ECMP adalah **0,49 %**.

2. Pada metode PCC

Tes ke nilai	I	II	III	IV	V	VI	Rata-rata
Pkt. Loss (%)	0,47	0,47	0,47	0,47	0,47	0,47	0.47

Tabel 4.5. Hasil Pengukuran Nilai Packet Loss Metode PCC

Sedangkan percobaan pada metode PCC menghasilkan rata-rat packet loss sebesar **0,47 %**.



Gambar 4.14. Grafik Packet Loss pada ECMP dan PCC

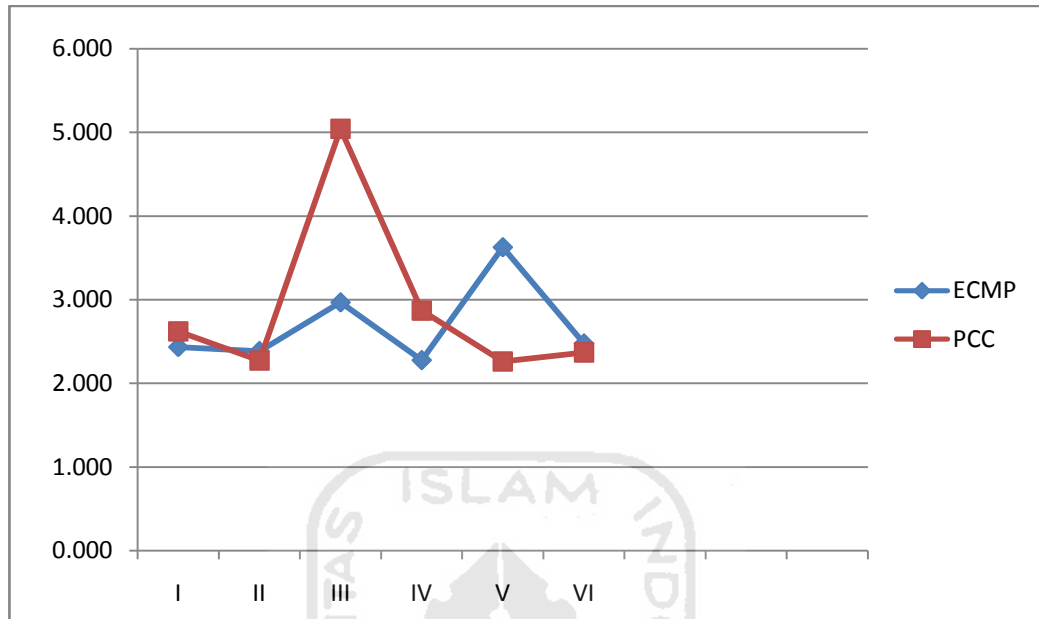
Ilustrasi diatas adalah hasil percobaan pengujian packet loss yang dituangkan dalam bentuk grafik.

d. Pengukuran nilai RTT

test	Paket	RTT pada ECMP (ms)				RTT pada PCC (ms)			
		Min	Avg	Max	Mdev	Min	Avg	Max	Mdev
I	30	1,843	2,435	3,333	0,481	2,007	2,623	5,328	0,940
II	30	1,760	2,387	3,203	0,346	1,827	2,275	3,202	0,372
III	30	1,927	2,969	6,473	1,378	1,913	5,045	11,896	3,357
IV	30	1,956	2,278	2,928	0,293	1,999	2,872	6,991	1,433
V	30	1,926	3,629	7,409	2,036	1,884	2,263	2,956	0,313
VI	30	2,006	2,477	2,863	0,265	1,672	2,372	4,098	0,621
Rata-rata		1,903	2,283	4,368	0,799	1,884	2,908	5,745	1,173

Tabel 4.6. Hasil Pengukuran Nilai RTT

Hasil pengujian RTT menghasilkan RTT sebesar **2,283 ms** pada metode ECMP, dan **2,908 ms** pada metode PCC.



Gambar 4.15. Grafik Rata-rata RTT pada ECMP dan PCC

Ilustrasi diatas adalah hasil dari pengujian RTT pada kedua metode load balancing yang dituangkan dalam bentuk grafik.

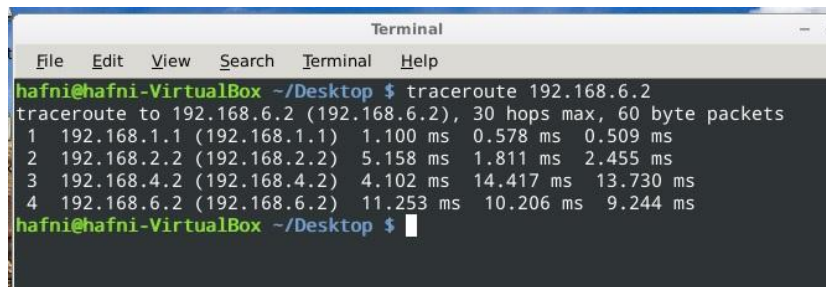
e. Pengujian Reliability

Untuk menguji reliabilitas dari sistem jaringan komputer, akan dilakukan suatu tindakan pada jaringan berupa pemutusan salah satu dari gateway internet untuk menguji kemampuan load balancer dalam menyediakan fungsi fail over.

Tool yang digunakan adalah traceroute. Traceroute dilakukan sebelum dan sesudah pemutusan salah satu gateway.

1. Pengujian pada metode ECMP

Berikut ini adalah hasil traceroute sebelum dilakukan pemutusan salah satu gateway.



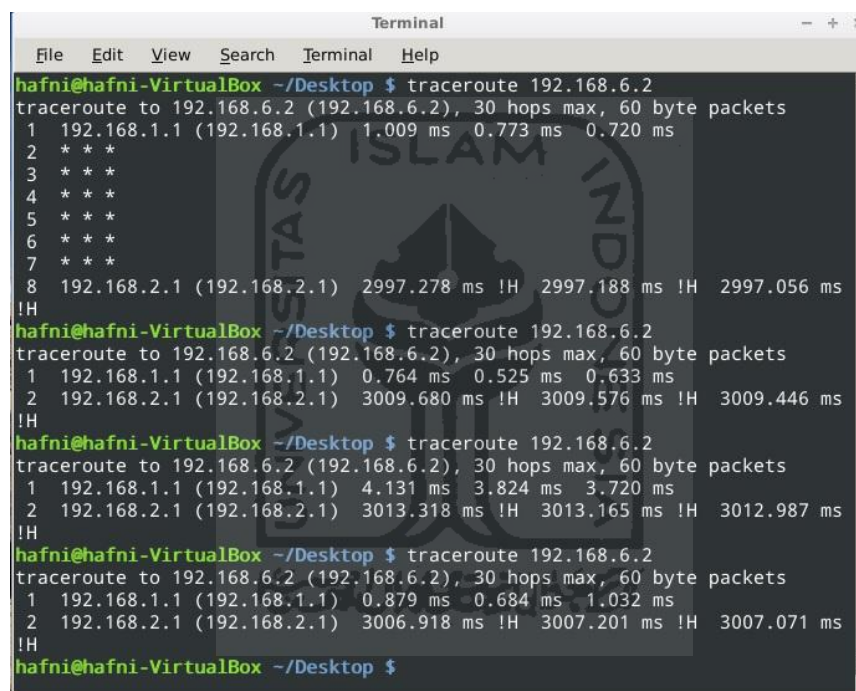
```

Terminal
File Edit View Search Terminal Help
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 1.100 ms 0.578 ms 0.509 ms
 2 192.168.2.2 (192.168.2.2) 5.158 ms 1.811 ms 2.455 ms
 3 192.168.4.2 (192.168.4.2) 4.102 ms 14.417 ms 13.730 ms
 4 192.168.6.2 (192.168.6.2) 11.253 ms 10.206 ms 9.244 ms
hafni@hafni-VirtualBox ~/Desktop $

```

Gambar 4.16. Traceroute Pertama Pada ECMP

Sedangkan gambar dibawah adalah hasil traceroute setelah dilakukan pemutusan salah satu gateway.



```

Terminal
File Edit View Search Terminal Help
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 1.009 ms 0.773 ms 0.720 ms
 2 * * *
 3 * * *
 4 * * *
 5 * * *
 6 * * *
 7 * * *
 8 192.168.2.1 (192.168.2.1) 2997.278 ms !H 2997.188 ms !H 2997.056 ms
!H
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 0.764 ms 0.525 ms 0.633 ms
 2 192.168.2.1 (192.168.2.1) 3009.680 ms !H 3009.576 ms !H 3009.446 ms
!H
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 4.131 ms 3.824 ms 3.720 ms
 2 192.168.2.1 (192.168.2.1) 3013.318 ms !H 3013.165 ms !H 3012.987 ms
!H
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 0.879 ms 0.684 ms 1.032 ms
 2 192.168.2.1 (192.168.2.1) 3006.918 ms !H 3007.201 ms !H 3007.071 ms
!H
hafni@hafni-VirtualBox ~/Desktop $

```

Gambar 4.17. Traceroute Pada ECMP Setelah Pemutusan salah satu Gateway

Dari gambar diatas dapat dilihat bahwa pada metode ECMP tidak dapat segera menyediakan fail over.

Dari percobaan yang penulis lakukan, fail over baru dapat didapatkan pada percobaan traceroute ke-5 setelah dilakukan pemutusan salah satu gateway, yang sebelumnya menggunakan gateway 192.168.2.2 menjadi 192.168.3.2. Berikut ini adalah gambar yang menunjukkan fungsi fail over pada metode ECMP.

```

Terminal
File Edit View Search Terminal Help
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 1.465 ms 1.315 ms 1.125 ms
 2 * * *
 3 * * *
 4 192.168.6.2 (192.168.6.2) 15.972 ms 15.877 ms 15.739 ms
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 0.705 ms 0.547 ms 0.488 ms
 2 192.168.3.2 (192.168.3.2) 5.881 ms 5.737 ms 5.602 ms
 3 192.168.5.2 (192.168.5.2) 7.180 ms 8.072 ms 8.844 ms
 4 192.168.6.2 (192.168.6.2) 12.135 ms 12.166 ms 12.030 ms
hafni@hafni-VirtualBox ~/Desktop $

```

Gambar 4.18. Traceroute ke-5 dan ke-6 Pada ECMP Setelah Pemutusan salah satu Gateway

Tes ke-	status
I	Tidak terkoneksi ke server
II	Tidak terkoneksi ke server
III	Tidak terkoneksi ke server
IV	Tidak terkoneksi ke server
V	Terkoneksi ke server

Tabel 4.7. Laporan hasil traceroute Pada ECMP setelah pemutusan salah satu gateway

2. Pengujian pada metode PCC

Berikut ini adalah hasil traceroute sebelum dilakukan pemutusan salah satu gateway.

```

Terminal
File Edit View Search Terminal Help
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 * * *
 2 192.168.2.2 (192.168.2.2) 3.663 ms 3.671 ms 3.536 ms
 3 192.168.4.2 (192.168.4.2) 5.881 ms 7.196 ms 9.028 ms
 4 192.168.6.2 (192.168.6.2) 11.895 ms 12.665 ms 13.422 ms
hafni@hafni-VirtualBox ~/Desktop $

```

Gambar 4.19. Traceroute Pertama Pada PCC

Sedangkan di bawah ini adalah gambar yang menunjukkan hasil traceroute setelah pemutusan salah satu gateway.

```

11 * * *
12 * * *
13 * * *
14 * * *
15 * * *
16 * * *
17 * * *
18 * * *
19 * * *
20 * * *
21 * * *
22 * * *
23 * * *
24 * * *
25 * * *
26 * * *
27 * * *
28 192.168.6.2 (192.168.6.2) 12.187 ms 12.050 ms 11.914 ms
hafni@hafni-VirtualBox ~/Desktop $ traceroute 192.168.6.2
traceroute to 192.168.6.2 (192.168.6.2), 30 hops max, 60 byte packets
 1 192.168.1.1 (192.168.1.1) 0.853 ms 0.863 ms 0.823 ms
 2 192.168.3.2 (192.168.3.2) 8.195 ms 8.109 ms 7.921 ms
 3 192.168.5.2 (192.168.5.2) 9.262 ms 10.068 ms 10.749 ms
 4 192.168.6.2 (192.168.6.2) 14.136 ms 17.455 ms 17.086 ms
hafni@hafni-VirtualBox ~/Desktop $

```

Gambar 4.20. Traceroute Pada PCC Setelah Pemutusan Salah Satu Gateway

Dari gambar percobaan diatas dapat dilihat bahwa metode PCC berhasil melakukan fail over setelah pemutusan salah satu gateway.

Tes ke-	status
I	Terkoneksi ke server

Tabel 4.8. Laporan hasil traceroute Pada PCC setelah pemutusan salah satu gateway

4.5. Pembahasan

Dari pengamatan pada grafik dan table hasil pengujian masing-masing parameter kinerja, kita dapat mengetahui bahwa:

- Nilai throughput pada metode PCC sedikit lebih besar daripada metode ECMP. Pada metode ECMP rata-rata nilai throughput yang dihasilkan adalah **1942 Kbits/s**, sedangkan pada metode PCC rata-rata nilai throughput yang dihasilkan adalah **1943 Kbits/s**.
- Nilai jitter pada metode PCC lebih tinggi daripada nilai jitter pada metode ECMP. Rata-rata nilai jitter pada metode ECMP adalah **2,104 ms**, sedangkan rata-rata nilai jitter pada metode PCC adalah **2,339 ms**.

- c. Nilai packet loss pada kedua metode load balancing relatif hampir sama. Hanya pada satu dari enam pengetesan persentase packet loss metode PCC lebih sedikit daripada metode ECMP. Rata-rata packet loss pada metode ECMP adalah **0,49 %**, sedangkan rata-rata packet loss pada metode PCC adalah **0,47 %**.
- d. Nilai response time atau RTT pada metode ECMP sedikit lebih cepat daripada metode PCC. Pada metode ECMP rata-rata RTT adalah **2,283 ms**. Sedangkan pada metode PCC rata-rata RTT adalah **2,908 ms**.
- e. Metode ECMP membutuhkan **5 (lima)** kali percobaan untuk dapat melakukan fail over, sedangkan metode PCC lebih cepat dalam melakukan failover karena langsung dapat melakukan fail over pada percobaan pertama setelah salah satu gateway diputus.

Untuk lebih jelasnya, hasil dari pengujian dapat dilihat pada tabel di bawah ini.

Parameter	ECMP	PCC
Throughput	1942 kbits/s	1943 kbits/s
Jitter	1,204 ms	2,339 ms
packet loss	0,49 %	0,47 %
RTT	2,283 ms	2,908 ms
reliability	Worse	Better

Tabel 4.9. Rangkuman Hasil Pengujian

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil dari penelitian yang telah dilakukan, dapat ditarik kesimpulan yang memuat kelebihan dan kekurangan dari masing-masing metode load balancing. Kesimpulan tersebut adalah sebagai berikut :

- a. Metode PCC menghasilkan throughput lebih baik daripada metode ECMP.
- b. Metode PCC memiliki ketahanan atau reliabilitas yang lebih baik ketika terjadi gangguan pada jaringan.
- c. Metode ECMP menghasilkan RTT yang lebih baik daripada metode PCC.
- d. Perbandingan nilai jitter dan packet loss pada kedua metode load balancing tidak signifikan perbedaannya. Selain itu tingkat degradasi pada keduanya masih dapat ditoleransi dan termasuk dalam kategori sangat bagus.

5.2. Saran

Saran penulis untuk pengembangan penelitian selanjutnya di masa mendatang berdasarkan kesimpulan yang diperoleh antara lain :

- a. Diharapkan penelitian mengenai load balancing metode ECMP dan PCC di masa yang akan datang menggunakan batasan masalah yang lebih luas, seperti jumlah user yang mengakses jaringan tidak dibatasi hanya 1 (satu), atau mengenai penggunaan simulator bisa menggunakan simulator selain VMware, sehingga dapat diketahui pengaruhnya terhadap kinerja.
- b. Diharapkan adanya analisis load balancing yang menggunakan metode selain metode ECMP dan PCC.

DAFTAR PUSTAKA

- [AKB09] Akbar, S., (2009), *Penelitian Tindakan Kelas : filosofi, metodologi & Implementasi*, Yogyakarta: Cipta Media Aksara.
- [IST05] Istijanto., (2005), *Riset SDM, Cara Praktis Mendeteksi Dimensi-Dimensi Kerja Karyawan*, Jakarta: PT. Gramedia Pustaka Utama.
- [LIN08] Linto Herlambang, Moch.; Catur L, Azis, (2008), *Panduan Lengkap Menguasai Router Masa Depan Menggunakan MikroTik RouterOS*, Yogyakarta: Andi Offset.
- [POP74] Popek, Gerald J; Goldberg, Robert P., (1974), *Formal Requirements for Virtualizable Third Generation Architectures*, New York: Association for Computing Machinery.
- [DIS10] Disastra, I., (2010), *Perbandingan Kinerja dan Keamanan PC Router Zebra dan Mikrotik RouterOS*, Skripsi, FTI, UII, Yogyakarta.
- [WUL11] Wulandari, S.; Affandi, A., (2011), Pengukuran Kinerja Layanan Jaringan Komputer Untuk Manajemen Ketersediaan, *SESINDO*, (p. 2), Surabaya.
- [INS09] Institut Teknologi Telkom, (2009), QoS dan Pengukurannya, In *Diktat Telkom* (p. 312), Bandung.
- [RIZ10] Rizaldi, H.; dkk., (2010), *Qos (Quality of Service)*, Makalah, FST, UINSK, Yogyakarta.
- [INT08] International Telecommunication Union, (2008), *One Way Transmission Time*, Tersedia di [http://www.cisco.com/en/US/tech/tk652/tk698/technologies_white_aper09186a00800a8993.shtml](http://www.cisco.com/en/US/tech/tk652/tk698/technologies_white_paper09186a00800a8993.shtml).
- [HOP00] Hopps, C., (2000), *Analysis of an Equal-Cost Multi-Path Algorithm*. Tersedia di <http://tools.ietf.org/html/rfc2992>.
- [MIK08] MikroTik, (2008), *Manual: PCC*, Tersedia di <http://wiki.mikrotik.com/wiki/Manual:PCC>.
- [IEE98] IEEE Communication Society, (1998), *Network Reliability*, Tersedia di

http://committees.comsoc.org/cqr/FAE_Docs/B1_Net_Rel/nw4reliab.html.

- [ROU07] Rouse, M., (2007), *Round Trip Time (RTT)*, Tersedia di <http://searchnetworking.techtarget.com/definition/round-trip-time>.
- [BLA07] Blair, E., (2006), *Latency*, Tersedia di <http://searchcio-midmarket.techtarget.com/definition/latency>.
- [DEW12] Dewobroto, P., (2012), *Load Balance Menggunakan Metode PCC*, Tersedia di http://www.mikrotik.co.id/artikel_lihat.php?id=34.

