



IMPLEMENTASI *LOAD BALANCING VIRTUAL ROUTER REDUNDANCY PROTOKOL* DENGAN MENGGUNAKAN ALGORITMA *ROUND ROBIN* PADA LAYANAN DATA

Muh. Amsyar^{*1}, Isnawaty², Natalis Ransi³

^{1,2,3}Jurusan Teknik Informatika, Fakultas Teknik, Universitas Halu Oleo, Kendari
e-mail: ^{*1}muhammadamsyar86@gmail.com, ²isnawaty@uho.ac.id, ³natalis.ransi@uho.ac.id

Abstrak

Jaringan komputer merupakan bagian terpenting dari sistem komunikasi dalam kehidupan kita, tanpa adanya jaringan komputer kita tidak bisa berkomunikasi antara satu dengan yang lainnya. Untuk itu kita perlu mempertahankan tingkat kualitas layanan jaringan dengan cara meminimalisir terjadinya kegagalan dalam sebuah perangkat jaringan, olehnya itu perlu adanya sebuah perangkat jaringan yang dapat mengambil alih dari perangkat utama. Virtual Router Redundancy Protokoli (VRRP) merupakan protokol yang bertanggung jawab menjalankan fungsi router backup saat kondisi router master mengalami gangguan.

Mikrotik adalah salah satu router yang dapat menerapkan teknik load balancing. Load balancing mempunyai empat metode yang bisa diterapkan salah satunya yaitu metode Nth, metode Nth merupakan pengembangan dari algoritma round robin. Load balancing adalah teknik untuk mendistribusikan beban trafik pada dua atau lebih jalur koneksi secara seimbang. Sistem yang dibangun menggunakan load balancing VRRP, sudah dapat menyelesaikan masalah ketika salah satu ISP mengalami putusnya koneksi. Sistem yang dibangun menggunakan metode Nth load balancing dapat membagi beban secara seimbang dari gateway yang satu ke gateway yang lain. Berdasarkan pengujian Quality of Service pada perangkat load balancing menggunakan routerboard mikrotik 951Ui-2HnD yang telah dilakukan, diperoleh hasil nilai delay, packetloss dan troughput yang baik begitu pula dengan nilai rata-rata yang diperoleh pada kategori Telecommunications and Internet Protokol Harmonization Over Networks (TIPHON).

Kata kunci— VRRP, Load Balancing, Nth, QoS

Abstract

Computer networks are the most important part of communication systems in our lives, without a computer network we cannot communicate with one another. For that we need to maintain the quality level of network services by minimizing the occurrence of failures in a network device, by which it is necessary to have a network device that can take over from the main device. Protokoli Virtual VR Redundancy (VRRP) is a protocol that is responsible for running the backup router functions when the master router has a fault condition.

Mikrotik is a router that can implement load balancing techniques. Load balancing has four methods that can be applied, one of which is the Nth method, the Nth method is the development of the round robin algorithm. Load balancing is a technique for distributing traffic loads on two or more



connection lines in a balanced manner. The system that was built using VRRP load balancing, was able to solve the problem when one ISP experienced a connection breakup. Systems that are built using the Nth load balancing method can divide the load equally from one gateway to another gateway. Based on the Quality of Service test on load balancing devices using the microtic routerboard 951Ui-2HnD, the results of the delay, packet loss and throughput values are good as well as the average values obtained in the Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON) category.

Keywords— VRRP, Load Balancing, Nth, QoS

1. PENDAHULUAN

Jaringan komputer merupakan bagian terpenting dari sistem komunikasi dalam kehidupan kita, tanpa adanya jaringan komputer kita tidak bisa berkomunikasi antara satu dengan yang lainnya. Banyak organisasi dan perusahaan yang membutuhkan layanan jaringan demi keperluan organisasi atau keperluan bisnis mereka oleh karena itu layanan jaringan harus tersedia 24 jam dalam sehari. Untuk mempertahankan tingkat kualitas layanan jaringan maka perlu meminimalisir kegagalan dalam sebuah perangkat jaringan, olehnya itu perlu adanya sebuah perangkat jaringan *redundant* atau perangkat jaringan yang dapat mengambil alih dari perangkat utama. *Virtual Router Redundancy Protokoli* (VRRP) merupakan protokol yang bertanggung jawab menjalankan fungsi *router backup* saat kondisi *router master* mengalami gangguan. VRRP didesain untuk menyediakan proses transisi yang cepat dari *router backup* ke *router master* sehingga meminimalkan gangguan pada sebuah layanan, menjaga kestabilan komunikasi dan dapat digunakan di beberapa vendor jaringan

Mikrotik adalah salah satu *router* yang dapat menerapkan teknik *load balancing*. Dalam proses *load balancing* terdapat empat metode yang bisa diterapkan menggunakan mikrotik, salah satunya yaitu metode Nth, metode Nth dikenal dengan metode pendistribusian pada *load balancing* yang merupakan pengembangan dari algoritma *round robin*. Metode Nth dapat bekerja dengan prinsip

per packet load balance maupun *per connection load balance*, maka dapat membagi penyebaran paket data yang merata pada masing-masing *gateway* dengan adanya teknik *load balancing* dengan metode Nth menjadi suatu pilihan solusi yang sangat efektif untuk menghindari terjadi ketimpangan pada jalur koneksi. Untuk itu diterapkan teknik *load balancing* yang bisa mendistribusi beban terhadap sebuah service yang ada pada server dengan memanfaatkan metode distribusi dua line koneksi yang disebut dengan metode Nth.

Load balancing adalah teknik untuk mendistribusikan beban *trafik* pada dua atau lebih jalur koneksi secara seimbang, agar *trafik* jaringan dapat berjalan dengan optimal maka perlu memaksimalkan nilai *throughput*, memperkecil waktu tanggap dan menghindari *overload* pada salah satu jalur koneksi. Penerapan algoritma *round robin* pada perangkat *load balancing* sangat dibutuhkan, karena dengan menggunakan algoritma tersebut setiap *request* dari *client* akan diarahkan kebeberapa jalur perangkat, sehingga perangkat akan menerima beban secara merata. Dengan adanya perangkat tambahan pada *load balancing* secara tidak langsung beban perangkat akan berkurang dan hal ini pula akan meminimalisir terjadinya *overload*. Dengan demikian akses pada jaringan akan semakin lebih efisien.

2. METODE PENELITIAN

2.1 *Load balancing*

a. Pengertian *Load balancing*

load balancing adalah suatu teknik pendistribusian beban trafik berdasarkan jaringan pada dua atau lebih jalur koneksi secara seimbang, agar trafik dapat berjalan optimal, memaksimalkan throughput, memperkecil waktu tanggap dan menghindari overload pada salah satu jalur koneksi.

b. Metode *Load Balancing*

Berdasarkan metodenya *Load Balancing* dapat di bagi menjadi beberapa metode:

1. *Fail Over* definisi *failover* dalam istilah *computer networking* adalah kemampuan sebuah sistem untuk dapat berpindah secara manual maupun otomatis jika salah satu sistem mengalami kegagalan sehingga menjadi backup untuk sistem yang mengalami kegagalan.
2. *Static route* dengan *Address list Static route* dengan *address list* adalah metode *load balancing* yang mengelompokan suatu range IP *address* untuk dapat di atur untuk melewati salah satu *gateway* dengan menggunakan *static routing*. Metode ini sering digunakan pada warnet yang membedakan PC untuk *browsing* dengan PC untuk game online. Mikrotik akan menentukan jalur *gateway* yang dipakai dengan membedakan *scr-address* pada paket data.
3. *Equal Cost Multi Path (ECMP) Equal cost multi path* adalah jalur keluar secara bergantian pada *gateway*. Contohnya jika ada dua *gateway*, dia akan melewati kedua *gateway* tersebut dengan beban yang sama (*Equal cost*) pada masing-masing *gateway*.
4. Nth Nth bukanlah sebuah singkatan. Melainkan sebuah bilangan integer (bilangan ke n). Nth merupakan algoritma *round robin* yang menentukan pembagian pemecahan *connection* yang akan di *mangle* ke *rule* yang di buat untuk *load balancing*. Nth *load balancing* merupakan suatu teknik *load balancing* yang membentuk suatu deret tertentu (Nth), yang nantinya akan digunakan sebagai suatu sistem antrian di dalam *mangle rule* yang dibentuk. Nth diimplementasikan dalam suatu deret yang terdiri dari *every* dan *packet* yang akan direalisasikan dalam suatu deret interger. Pada metode *load balancing* ini, paket data yang masuk akan ditandai sebagai suatu variabel n dalam tipe data integer .

5. *Per Connection Classifier (PCC)* merupakan metode yang menspesifikasikan suatu paket menuju *gateway* suatu koneksi tertentu. PCC mengelompokan trafik koneksi yang keluar masuk *router* menjadi beberapa kelompok. Pengelompokan ini bisa di bedakan berdasarkan *scr-address*, *scr-port* dan *dst-port*. Mikrotik akan mengingat jalur *gateway* yang telah dilewati diawal trafik koneksi. Sehingga pada paket-paket data selanjutnya yang masih berkaitan akan dilewatkan pada jalur *gateway* yang sama dengan paket data sebelumnya yang sudah dikirim atau istilahnya *PCC Matcher*. [1]

2.2 Firewall

Firewall Adalah suatu sistem yang dirancang untuk mencegah akses yang tidak diinginkan dari atau ke dalam suatu jaringan internal. *firewall* bekerja dengan cara melacak dan mengendalikan jalannya data serta memutuskan aksi untuk melewatkan (*pass*), menjatuhkan (*drop*), menolak (*reject*), mengenkripsi atau melakukan pencatatan aktivitas (*log*) data. *Firewall* menjamin agar data sesuai dengan aturan (*rule*) yang terdapat di dalam kebijakan keamanannya (*security policy*) yaitu seperangkat aturan yang telah didefinisikan di dalam keamanan jaringan internal. Umumnya, sebuah *firewall* digunakan untuk membatasi atau mengontrol akses terhadap siapa saja yang memiliki akses terhadap jaringan pribadi dari pihak luar. Saat ini, istilah *firewall* menjadi istilah lazim yang merujuk pada sistem yang mengatur komunikasi antar dua macam jaringan yang berbeda. [2]

2.3 Network Address Translation (NAT)

Network Address Translation (NAT) adalah suatu metode untuk menghubungkan lebih dari satu komputer ke jaringan internet dengan menggunakan satu alamat IP *publik* (Grang and Gupta, 2013).

Metode NAT banyak digunakan di seluruh dunia termasuk di Indonesia. Pada dasarnya semua jenis NAT beroperasi dengan cara *client – server*. Dalam hal ini, *client* di zona internal yang memulai permintaan untuk memperoleh sumber daya dari *server* di zona internet *publik* (Masoud, 2013). Di sini semua *client* akan mendapatkan alamat IP lokal yang diberikan oleh komputer *server*. Dengan

mekanisme NAT terbatasnya IP *publik* tidak menjadi masalah. Salah satu syarat untuk menghubungkan komputer ke jaringan internet adalah dengan menggunakan IP *publik*. Melalui NAT memungkinkan beberapa node untuk berbagi satu atau lebih alamat IP *publik*. Gateway NAT berada pada batas jaringan lokal dan *publik* dan memodifikasi alamat IP lokal dan *port* dari paket yang diperuntukkan untuk jaringan publik. Paket IP yang dibundel dengan IPSec, seperti AH dan ESP secara intrinsik dimaksudkan untuk melindungi integritas dari paket IP (termasuk sumber dan tujuan alamat) dari perubahan atau gangguan karena peran fundamental NAT gateway untuk memodifikasi alamat IP dalam header paket, IPSec, dan NAT memiliki ketidakcocokan intrinsik (Ahmad and Yaacob, 2012).

NAT bekerja dengan mengalihkan suatu paket data dari suatu alamat IP ke alamat IP lainnya. Ketika suatu paket dialihkan, NAT akan mengingat dari mana asal paket dan kemana tujuan paket tersebut. Apabila paket kembali, NAT akan mengirimkannya ke alamat asal atau dengan kata lain host hanya akan menerima paket yang dikirim atau yang dimintanya sehingga komunikasi dapat berjalan dengan baik. Jaringan komputer LAN yang menggunakan NAT disebut dengan NATted *Network*. Sebagai contoh, di MikroTik NAT digunakan untuk komunikasi internal dan komunikasi eksternal maksudnya pengalihan data dapat dilakukan untuk paket yang berasal dari jaringan NATted (internal) ke jaringan luar eksternal atau dari jaringan luar menuju jaringan NATted. Hal tersebut sering disebut dengan komunikasi dua arah dari dan ke jaringan NATted atau internal.[3]

Network Address Translation (NAT) merupakan metode untuk menghubungkan lebih dari satu komputer ke jaringan internet dengan menggunakan satu alamat IP (Zamzami, N. F, 2013). Ada 3 jenis utama dari NAT, yaitu:

1. *Source* NAT atau SNAT Dipergunakan untuk merubah *source address* dari suatu paket data. Sebagai contoh penggunaan SNAT adalah pada *gateway* internet, dimana ketika suatu *host* (pc) dalam LAN melakukan koneksi internet, yang terlihat dari internet adalah IP publiknya bukan IP lokalnya.
2. *Destination* NAT atau DNAT Dipergunakan

jika ingin meneruskan (*redirect*) paket dari IP public melalui *firewall* kedalam suatu *host*.

3. *Bidirectional* NAT atau *Masquerade* sama dengan SNAT, akan tetapi *Masquerade* biasanya dipergunakan jika menggunakan IP publik yang dinamis semacam koneksi ADSL, sedangkan SNAT biasa digunakan untuk IP statik seperti layanan *dedicated*. [4]

2.4 Routing

Sebuah mekanisme yang digunakan untuk mengarahkan dan menentukan jalur yang akan dilewati paket dari satu *device* ke *device* yang berada di jaringan lain (Wijaya, A.,2016).

Proses *routing* yang dilakukan oleh *host* cukup sederhana. Jika *host* tujuan terletak di jaringan yang sama atau terhubung langsung. IP datagram dikirim langsung ke tujuan. Apabila *routing host* menuju jaringan yang berbeda (internet), IP datagram dikirim ke *default router*. Router ini yang akan mengatur pengiriman IP selanjutnya, hingga sampai ke tujuannya, dalam suatu tabel routing terhadap: IP *address* tujuan, IP *address next hop router (gateway)*, *flag* yang menyatakan jenis *routing*. Dalam proses meneruskan paket ke tujuan, IP *router* akan melakukan hal-hal berikut :

1. Mencari di *table routing, entry* yang cocok dengan IP *address* tujuan. Jika ditemukan, paket akan dikirim ke *next hop router* atau *interface* yang terhubung langsung dengannya.
2. Mencari di *table routing, entry* yang cocok dengan alamat jaringan dari tujuan jaringan. Jika ditemukan, paket dikirim ke *next hop router* tersebut.
3. Mencari *table routing, entry* data yang bertanda *default*, jika ditemukan paket dikirim ke *router* tersebut. [5]

2.5 Quality of Service (QoS)

Quality of Service (QoS) adalah kemampuan suatu jaringan untuk menyediakan layanan yang baik dengan menyediakan *bandwidth*, mengatasi *jitter* dan *delay*. Parameter QoS adalah *latency, jitter, packet loss, throughput, MOS*. QoS sangat ditentukan oleh kualitas jaringan yang digunakan. Terdapat beberapa faktor yang dapat menurunkan nilai QoS, seperti: *Redaman, Distorsi, dan Noise*. Performa jaringan komputer dapat bervariasi akibat dari beberapa masalah, seperti halnya masalah *bandwidth, latency* dan *jitter*, yang dapat membuat efek yang cukup besar bagi banyak

aplikasi. Sebagai contoh, *video streaming* dapat membuat pengguna kesal ketika paket data aplikasi tersebut berjalan dengan *bandwidth* yang tidak cukup, dengan *latency* yang tidak dapat diprediksi, atau *jitter* yang berlebih. Beberapa fitur *Quality of Service* (QoS) dapat menangani masalah diatas, dapat menurunkan *latency* dengan mengendalikan pengiriman paket data dan membatasi paket data tertentu, *jitter* yang dapat diprediksi dan dicocokkan dengan kebutuhan aplikasi yang digunakan di dalam jaringan tersebut.[6]

Beberapa fitur *Quality of Service* (QoS) dapat menangani masalah diatas, dapat menurunkan *latency* dengan mengendalikan pengiriman paket data dan membatasi paket data tertentu, *jitter* yang dapat diprediksi dan dicocokkan dengan kebutuhan aplikasi yang digunakan di dalam jaringan tersebut. Performansi mengacu ke tingkat kecepatan dan keandalan penyampaian berbagai jenis beban data di dalam suatu komunikasi. Performansi merupakan kumpulan dari beberapa parameter teknis yaitu :

1. Delay (Waktu Tunda)

Delay (Waktu Tunda) *Delay* adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama. Untuk mencari nilai *Delay*, dapat menggunakan Persamaan (1).

$$Delay = \frac{Total\ delay}{Jumlah\ total\ packet}$$

Delay versi *Telecommunications and Internet Protocol Harmonization Over Networks* (TIPHON) dikelompokkan menjadi empat kategori seperti terlihat pada Tabel 1.

Tabel 1. Kategori *Delay*

Kategori Degradasi	<i>Delay</i>
Sangat Bagus	0 ms
Bagus	0 s/d 75 ms
Sedang	75 s/d 125 ms
Buruk	>125 ms

2. Packet Loss (Paket Hilang)

Packet Loss (Paket Hilang) Merupakan suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang, dapat terjadi karena *collision* dan *congestion* pada jaringan dan hal ini berpengaruh pada semua aplikasi karena

retransmisi akan mengurangi efisiensi jaringan secara keseluruhan meskipun jumlah *bandwidth* cukup tersedia untuk aplikasi-aplikasi tersebut. Umumnya perangkat jaringan memiliki *buffer* untuk menampung data yang diterima. Jika terjadi *kongesti* yang cukup lama, *buffer* akan penuh, dan data baru tidak akan diterima. Untuk mencari nilai *Packet Loss*, dapat menggunakan Persamaan 2.

$$Packet\ Loss = \frac{Packet\ tercapture - paket\ terkirim}{Packet\ total\ tercapture} \times 100\%$$

Packet loss Versi *Telecommunications and Internet Protocol Harmonization Over Networks* (TIPHON) dikelompokkan menjadi empat kategori seperti terlihat pada Tabel 2.

Tabel 2. Kategori *Packet Loss*

Kategori Degradasi	<i>Packet Loss</i>
Sangat Bagus	0 %
Bagus	3%
Sedang	15 %
Buruk	25 %

3. Throughput

Throughput adalah kecepatan (*rate*) *transfer* data efektif, yang diukur dalam bps. *Troughput* merupakan jumlah total kedatangan paket yang sukses yang diamati pada destination selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. Nilai *Throughput* dapat dihitung menggunakan Persamaan (3).

$$Throughput = \frac{jumlah\ data\ yang\ dikirim}{waktu\ pengiriman\ data}$$

Sedangkan nilai *Throughput* Versi *Telecommunications and Internet Protocol Harmonization Over Networks* (TIPHON) dikelompokkan menjadi empat kategori seperti terlihat pada Tabel (3).

Tabel 3 Kategori *Throughput*

Kategori Degradasi	<i>Throughput</i>
Sangat Bagus	76 - 100 %
Bagus	51 - 75%
Sedang	26 - 50 %
Buruk	< 25 %

Dalam usaha menjaga dan meningkatkan nilai QoS, dibutuhkan teknik untuk menyediakan utilitas jaringan, yaitu dengan mengklasifikasikan dan

memprioritaskan setiap informasi sesuai dengan karakteristiknya masing-masing. Contohnya, terdapat paket data yang bersifat sensitif terhadap *delay* tetapi tidak sensitif terhadap *packet loss* seperti VoIP, ada juga paket yang bersifat sensitif terhadap *packet loss* tetapi tidak sensitif terhadap *delay* seperti transfer data.[7]

3. HASIL DAN PEMBAHASAN

3.1 Implementasi

Tahap implementasi merupakan tahap penerapan sistem agar dapat berjalan sesuai dengan kebutuhan dan perancangan yang telah dibuat. Selain itu juga pada implementasi ini akan dijelaskan bagaimana sistem ini akan bekerja.

Penelitian mengenai *Load Balancing* menggunakan algoritma *round robin* untuk layanan internet berbasis WLAN ini digunakan beberapa perangkat keras (*hardware*) dan perangkat lunak (*software*) yang nantinya akan dilakukan pengukuran *Quality of Service (QoS)*.

Adapun *hardware* yang digunakan antara lain laptop yang berfungsi untuk melakukan konfigurasi pada *RouterBoard Mikrotik 951Ui-2HnD* dan *capture data*. Sedangkan *software* yang digunakan adalah *Winbox* untuk melakukan konfigurasi *RouterBoard Mikrotik 951Ui-2HnD* dan *Wireshark* untuk melakukan *capture data* untuk menghitung nilai QoS secara manual.

Berikut ini adalah spesifikasi beberapa perangkat yang digunakan dalam penelitian ini, diantaranya yaitu :

1. RouterBoard Mikrotik 951Ui-2HnD

Spesifikasi dari *RouterBoard Mikrotik 951Ui-2HnD* yang digunakan dapat dilihat pada Tabel 1.

Tabel 1 Spesifikasi dari Router Board Mikrotik 951Ui-2HnD

No	Nama Perangkat	Spesifikasi
1	Size RAM	128 MB
2	Wireless Standard	802.11b/g/n
3	Operating System	RouterOS
4	10/100 Ethernet Ports	5
5	Storage Size	128

2. Personal Computer

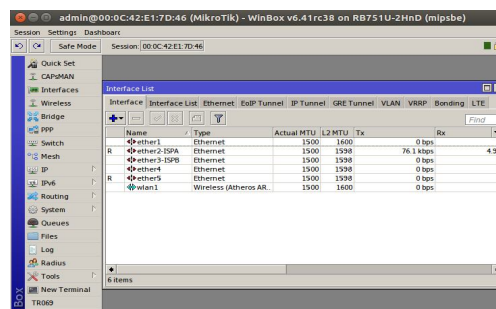
Adapun personal komputer yang digunakan untuk merancang konfigurasi *load balancing* adalah sebagai berikut :

Tabel 2 Spesifikasi dari personal computer yang digunakan

No	Nama Perangkat	Spesifikasi
1	Prosesor	Intel Core i3
2	Memori	Monitor 15,6 inch
3	Monitor	RAM 4 GB DDR3
4	Hardisk	1TB HDD

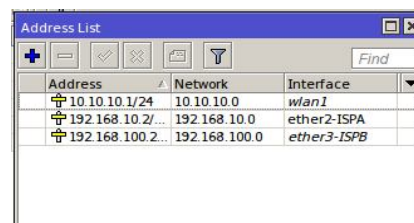
3.2 Konfigurasi Load Balancing

Dalam tahapan konfigurasi *load balancing*, hal pertama yang dilakukan adalah melakukan konfigurasi *hardware*. Langkah awalnya yaitu memasang LAN internet ISPA dan LAN internet ISPB dari sumber ke port *Ethernet* pada mikrotik. Selanjutnya untuk memudahkan dalam mengembangkan sistem, maka perlu dilakukannya inisialisasi *interface* dengan cara memberikan nama pada masing-masing *interface*. yang ditujukan pada gambar 1.



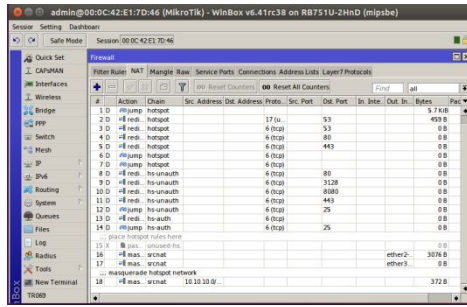
Gambar 1 Konfigurasi Ethernet Interface

Berikutnya pemberian *IP address* agar setiap *client* bisa saling terhubung dengan *router*. *Router* yang digunakan memiliki tiga *interface* yaitu *interface ether2-ISP*, *ether3-ISP* yang merupakan *interface* untuk melewatkan koneksi internet dan *interface wlan* yang digunakan untuk tempat koneksi *client* atau *interface* yang menghubungkan antara *router* dengan *client*.



Gambar 2 IP Address pada Interface

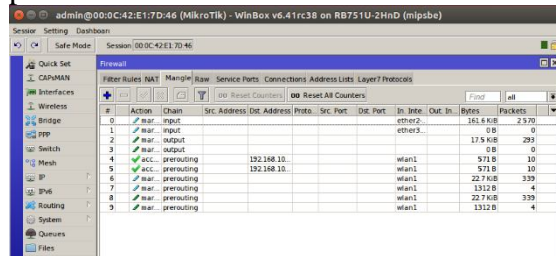
Agar komputer *client* dapat terhubung dengan internet, maka perlu dilakukan translasi dari *IP private* yang dimiliki *client* ke *IP public*.



Gambar 3 Konfigurasi NAT

Mangle adalah tahapan dimana paket yang datang dari suatu *interface* tertentu akan diproses. Fungsi dari aturan yang ada di *mangle* adalah untuk menandai paket agar diarahkan sesuai dengan *rule routing* yang ada. Di tahap ini, penulis akan menerapkan aturan *rule mangle* dari *load balancing* menggunakan metode *Nth*.

Setiap paket data akan diberi tanda oleh *Nth* secara beruntun dan berulang-ulang. Ini menyebabkan metode *Nth load balancing* menggunakan algoritma *round robin*. Algoritma *round robin* ini dapat membagi trafik jaringan secara merata agar tidak terjadi *overload* pada saat pengiriman paket.



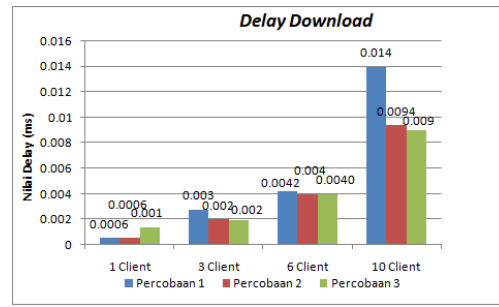
Gambar 4 Konfigurasi Mangle Meode Nth

3.3 Grafik Pengujian *Quality of Service*

Berdasarkan hasil pengujian *Download, Upload, Streaming, Download Upload, Download Streaming, Download Upload Streaming* dan *Upload Streaming* dari beberapa percobaan yang dilakukan.

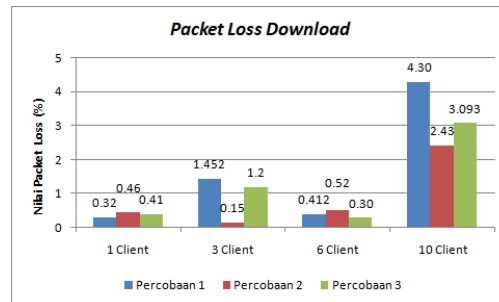
1. *Download*

Grafik yang ditunjukkan pada Gambar 5 menerangkan nilai *Delay mode Download* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *delay* berkisar dari 0,0006 ms sampai 0,001 ms yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Delay*.



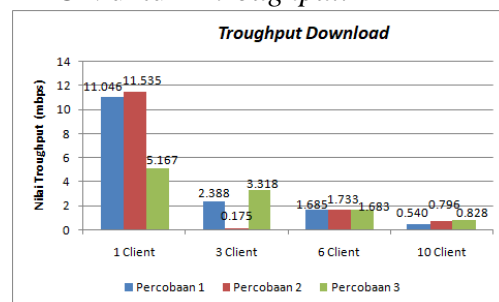
Gambar 5 Delay Download

Grafik yang ditunjukkan pada Gambar 6 menerangkan nilai *Packet Loss mode Download* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Packet Loss* berkisar dari 0,32 % sampai 0,41 % yang termasuk dalam kategori sangat baik untuk setiap percobaan yang dilakukan.



Gambar 6 Packet Loss mode Download

Grafik yang ditunjukkan pada Gambar 7 menerangkan nilai *Throughput mode Download* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 5,167 mbps sampai 11,046 mbps yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Throughput*.

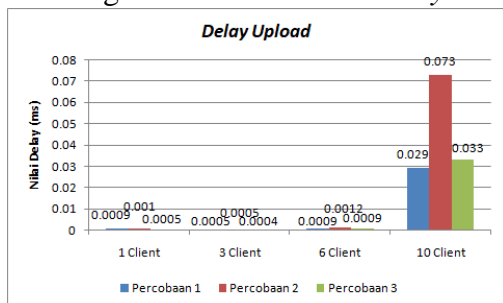


Gambar 7 Throughput mode Download

2. *Upload*

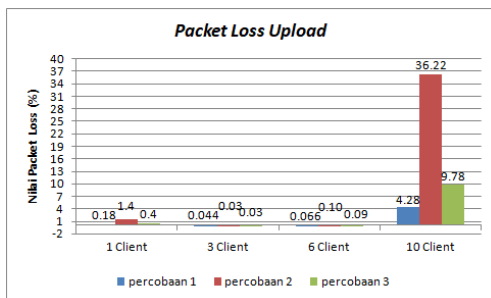
Grafik yang ditunjukkan pada Gambar 8 menerangkan nilai *Delay mode*

Upload pada 3 kali percobaan dengan 1 client, 3 client, 6 client dan 10 client. Dari grafik bisa dilihat bahwa nilai Delay berkisar dari 0,0009 ms sampai 0,0005 ms yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk Delay.



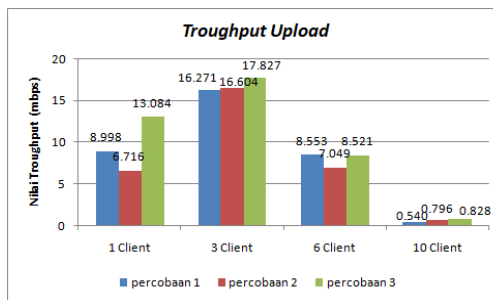
Gambar 8 Delay mode Upload

Grafik yang ditunjukkan pada Gambar 9 menerangkan nilai Packet Loss mode Upload pada 3 kali percobaan dengan 1 client, 3 client, 6 client dan 10 client. Dari grafik bisa dilihat bahwa nilai Packet Loss berkisar dari 0,18 % sampai 0,4% yang termasuk dalam kategori sangat baik untuk 1 client sampai 6 client sedangkan pada 10 client nilai Packet Loss percobaan 2 masuk dalam kategori cukup dalam kategori TIPHON untuk Packet Loss.



Gambar 9 Packet Loss mode Upload

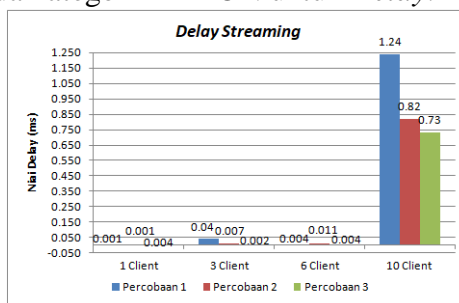
Grafik yang ditunjukkan pada Gambar 10 menerangkan nilai Throughput mode Upload pada 3 kali percobaan dengan 1 client, 3 client, 6 client dan 10 client. Dari grafik bisa dilihat bahwa nilai Throughput berkisar dari 8,998 mbps sampai 13,084 mbps yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk Throughput.



Gambar 10 Throughput mode Upload

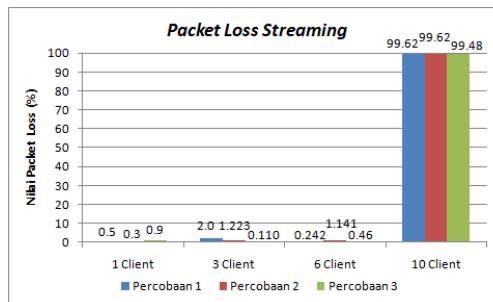
3. Streaming

Grafik yang ditunjukkan pada Gambar 11 menerangkan nilai Delay mode Streaming pada 3 kali percobaan dengan 1 client, 3 client, 6 client dan 10 client. Dari grafik bisa dilihat bahwa nilai delay berkisar dari 0,001 ms sampai 0,004 ms yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk Delay.



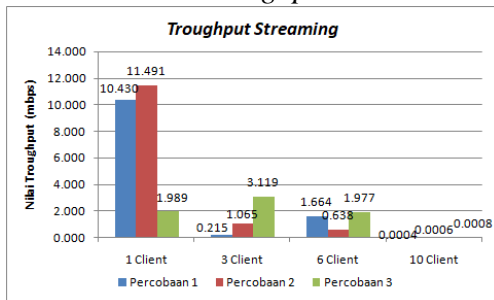
Gambar 11 Delay mode Streaming

Grafik yang ditunjukkan pada Gambar 12 menerangkan nilai Packet Loss mode Streaming pada 3 kali percobaan dengan 1 client, 3 client, 6 client dan 10 client. Dari grafik bisa dilihat bahwa nilai Packet Loss berkisar dari 0,5 % sampai 0,9% yang termasuk dalam kategori sangat baik untuk 1 client sampai 6 client sedangkan pada 10 client nilai Packet Loss masuk dalam kategori cukup pada kategori TIPHON untuk Packet Loss.



Gambar 12 Packet Loss mode Streaming

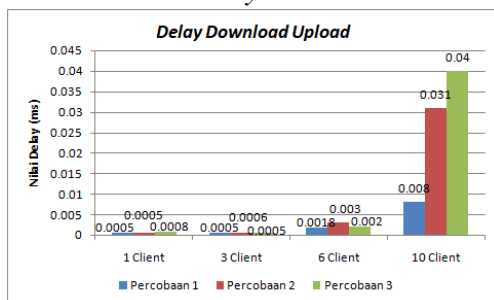
Grafik yang ditunjukkan pada Gambar 13 menerangkan nilai *Throughput mode Streaming* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 1,989 mbps sampai 10,430 mbps yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Throughput*.



Gambar 13 *Throughput mode Streaming*

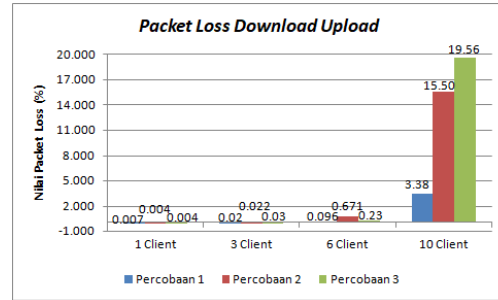
4. *Download Upload*

Grafik yang ditunjukkan pada Gambar 14 menerangkan nilai *Delay mode Download Upload* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 0,0005 ms sampai 0,0008 ms yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Delay*.



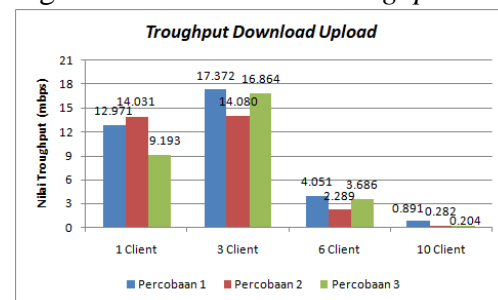
Gambar 14 *Delay mode Download Upload*

Grafik yang ditunjukkan pada Gambar 15 menerangkan nilai *Packet Loss mode Download Upload* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Packet Loss* berkisar dari 0,007% sampai 0,004% yang termasuk dalam kategori sangat baik untuk 1 *client* dan 6 *client* sedangkan 10 *client* pada nilai *Packet Loss* masuk dalam kategori bagus pada kategori TIPHON untuk *Packet Loss*.



Gambar 15 *Packet Loss mode Download Upload*

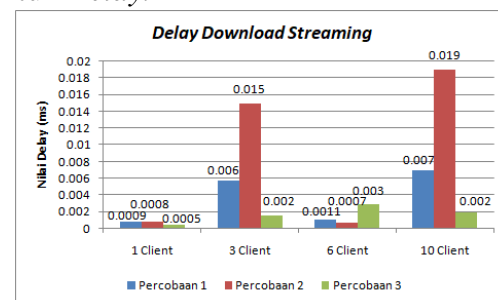
Grafik yang ditunjukkan pada Gambar 16 menerangkan nilai *Throughput mode Download Upload* pada 3 kali percobaan dengan 1 *client*, 5 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 9,193 mbps sampai 12,971 mbps yang termasuk dalam kategori bagus pada kategori TIPHON untuk *Throughput*.



Gambar 16 *Throughput Download Upload*

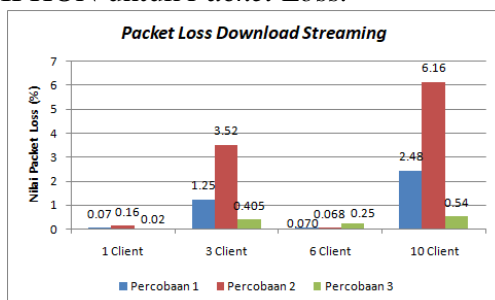
5. *Download Streaming*

Grafik yang ditunjukkan pada Gambar 17 menerangkan nilai *Delay mode Download Streaming* pada 3 kali percobaan dengan 1 *client*, 5 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Delay* berkisar dari 0,0009 ms sampai 0,0005 ms yang termasuk dalam kategori bagus pada kategori TIPHON untuk *Delay*.



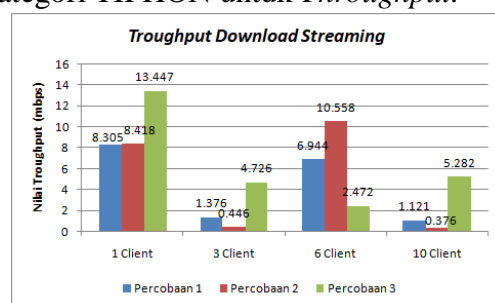
Gambar 17 *Delay Download Streaming*

Grafik yang ditunjukkan pada Gambar 18 menerangkan nilai *packet loss mode Download Streaming* pada 3 kali percobaan dengan 1 *client*, 5 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *packet loss* berkisar dari 0,07 % sampai 0,02 % yang termasuk dalam kategori sangat baik untuk 1 *client* sampai 6 *client* sedangkan 10 *client* pada percobaan 2 nilai *Packet Loss* masuk dalam kategori bagus pada kategori TIPHON untuk *Packet Loss*.



Gambar 18 *packet loss mode Download Streaming*

Grafik yang ditunjukkan pada Gambar 19 menerangkan nilai *Throughput mode Download Streaming* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 8,305 mbps sampai 13,447 mbps yang termasuk dalam kategori bagus pada kategori TIPHON untuk *Throughput*.

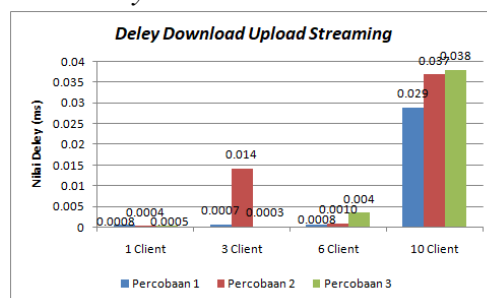


Gambar 19 *Throughput mode Download Streaming*

6. Download Upload Streaming

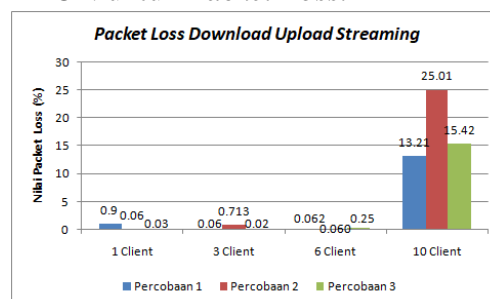
Grafik yang ditunjukkan pada Gambar 20 menerangkan nilai *Delay mode Download Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat

bahwa nilai *Delay* berkisar dari 0,0008 ms sampai 0,0005 ms yang termasuk Dalam kategori baik pada kategori TIPHON untuk *Delay*.



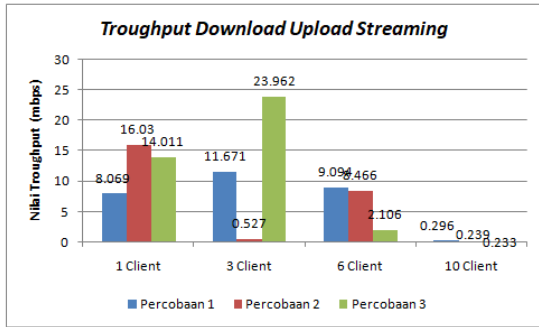
Gambar 20 *Delay mode Download Upload Streaming*

Grafik yang ditunjukkan pada Gambar 21 menerangkan nilai *Packet Loss mode Download Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Packet Loss* berkisar dari 0,09 % sampai 0,03 % yang termasuk dalam kategori sangat baik kecuali pada percobaan 2 untuk 10 *client* termasuk dalam kategori cukup pada kategori TIPHON untuk *Packet Loss*.

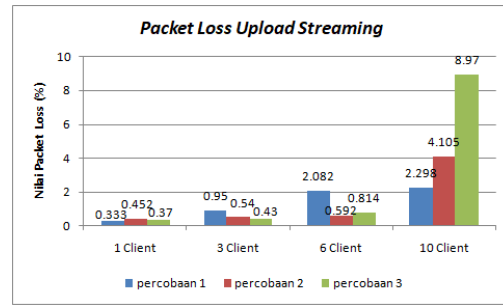


Gambar 21 *Packet Loss mode Download Upload Streaming*

Grafik yang ditunjukkan pada Gambar 22 menerangkan nilai *Throughput mode Download Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 8,069 mbps sampai 14,011 mbps yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Throughput*.



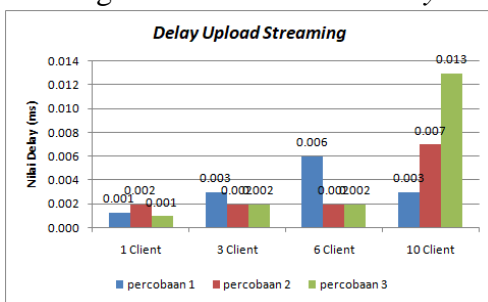
Gambar 22 Troughput mode Download Upload Streaming



Gambar 24 Packet Loss mode Download Upload Streaming

7. Upload Streaming

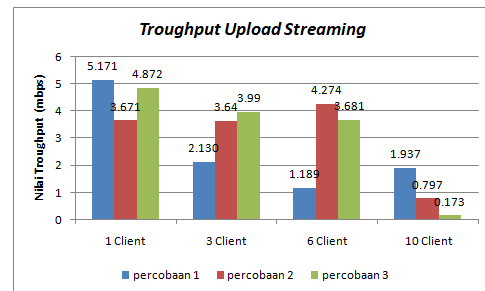
Grafik yang ditunjukkan pada Gambar 23 menerangkan nilai *Delay* mode *Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 5 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Delay* berkisar dari 0,001 ms sampai 0,001 ms yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Delay*.



Gambar 23 Delay mode Upload Streaming

Grafik yang ditunjukkan pada Gambar 24 menerangkan nilai *Packet Loss* mode *Download Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 5 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Packet Loss* berkisar dari 0,333 % sampai 0,37 % yang termasuk dalam kategori sangat baik kecuali pada percobaan 3 untuk 10 *client* termasuk dalam kategori bagus pada kategori TIPHON untuk *Packet Loss*.

Grafik yang ditunjukkan pada Gambar 25 menerangkan nilai *Throughput* mode *Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 5 *client* dan 10 *client*. Dari grafik bisa dilihat bahwa nilai *Throughput* berkisar dari 4,872 mbps sampai 5,171 mbps yang termasuk dalam kategori sangat baik pada kategori TIPHON untuk *Throughput*.



Gambar 25 Troughput mode Upload Streaming

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan selama perancangan sampai analisis QoS dapat disimpulkan.

1. Sistem yang dibangun menggunakan *load balancing VRRP*, sudah dapat menyelesaikan masalah ketika salah satu ISP mengalami terputusnya koneksi. Hal ini dapat dilihat dari pengalihan koneksi secara otomatis ke *gateway* dari ISP yang masih aktif, sehingga kinerja jaringan tetap berjalan normal.
2. Sistem yang dibangun menggunakan metode *Nth load balancing* dapat membagi beban secara seimbang berdasarkan urutan dan akan bergiliran dari *gateway* yang satu ke *gateway*

yang lain. Oleh karena itu *gateway* yang digunakan selalu bergantian dan tidak tetap.

3. Berdasarkan pengujian *Quality of Service* pada perangkat *load balancing* menggunakan *routerboard mikrotik 951Ui-2HnD* yang telah dilakukan, nilai *delay*, *packetloss* dan *troughput* pada *mode Download, Upload, Streaming, Download Upload, Download Streaming, Download Upload Streaming* dan *Upload Streaming* pada 3 kali percobaan dengan 1 *client*, 3 *client*, 6 *client* dan 10 *client* diperoleh hasil yang baik begitu pula dengan nilai rata-rata yang diperoleh pada kategori *Telecommunications and Internet Protokol Harmonization Over Networks (TIPHON)*.

5. Saran

Berdasarkan kesimpulan diatas, penulis mengajukan beberapa saran untuk pengembangan lebih lanjut dari penelitian yang telah dilakukan, antara lain:

1. Memfungsikan fitur lain dari mikrotik yaitu *bandwith management* agar dapat membagi *bandwith* secara merata sesuai dengan jumlah *client* yang aktif.
2. *Load balancing* pada ISP dapat dikembangkan menggunakan lebih dari dua koneksi.
3. Dapat dikembangkan dengan menambahkan parameter QoS yang lain. Untuk pengujian QoS dibutuhkan jaringan yang stabil untuk mendapatkan nilai yang maksimal.

Blok Situs Pada Jaringan LAN,” *Tek. DAN Sist. Inf.*, Vol. 3, No. 2, pp. 56–62, 2017.

- [3] Tutang, A. Tutuko, and K. Rosyidin, “Implementasi Network Address Translation (NAT) Menggunakan Kerio Control Versi 7.4.1 di Pusat Penelitian Bioteknologi – LIPI,” *Dokumentasi dan Inf.*, Vol. 6, No. 1, pp. 97–108, 2015.
- [4] F. Ardianto, B. Alfaresi, and A. Darmadi, “Rancang Bangun Load Balancing Dua Internet Service Provider (ISP) Berbasis Mikrotik,” *Surya Energy*, Vol. 3, No. 1, pp. 198–202, 2018.
- [5] Armanto, “Perancangan Pengelolaan Jaringan Load Balancing Dan Fileover Menggunakan Router Mikrotik RB 951 Series Pada STKIP PGRI Lubuklinggau,” *Jusikom*, Vol. 2, No. 2, pp. 83–90, 2017.
- [6] A. Emrul, M. Yamin, and L. Surimi, “ANALISIS QUALITY of SERVICE (QoS) Kinerja Sistem Hotspot Pada Routerboard Mikrotik 951Ui-2HnD Pada Jaringan Teknik Informatika,” *semanTIK*, Vol. 3, No. 2, pp. 87–94, 2017.
- [7] A. N. W. Wardhana, M. Yamin, and L. F. Aksara, “ANALISIS QUALITY OF SERVICE (QoS) Jaringan Internet Berbasis Wireless Lan Pada Layanan Indihome,” *semanTIK*, Vol. 3, No. 3, pp. 201–248, 2017.

DAFTAR PUSTAKA

- [1] T. Sukendar, “Keseimbangan Bandwidth Dengan Menggunakan Dua ISP Melalui Metode Nth Load Balancing Berbasis Mikrotik,” *Tek. Komput. AMIK BSI*, Vol. 3, No. 1, pp. 86–92, 2017.
- [2] M. Siddik, “Implementasi Mikrotik Router Board 750 Sebagai Firewall