



## Perbandingan QOS Dari Metode NTH ECMP dan PCC untuk Layanan Berbasis Konten

Winarno Sugeng<sup>#1</sup>, Fazza Dwi Riandy<sup>#2</sup>

<sup>#</sup>Program Studi Informatika, Institut Teknologi Nasional, Bandung

Jl. Phh. Mustofa No.23, Neglasari, Kec. Cibeunying Kaler, Kota Bandung, Jawa Barat 40124

<sup>1</sup>winarno@itenas.ac.id

<sup>2</sup>fazzariandy@mhs.itenas.ac.id

**Abstrak**— Tingginya kebutuhan *resource* untuk pengaksesan konten di internet sejalan dengan tingginya kebutuhan performa jaringan internet yang stabil dan efisien. Untuk menciptakan jaringan yang stabil dibutuhkan sebuah proses untuk meringankan beban jaringan. *Load balancing* merupakan sebuah metode yang dapat mendistribusikan beban jaringan ke dua koneksi internet atau lebih. Beban jaringan yang terdistribusi akan menciptakan kualitas jaringan yang optimal. Hal itu dipicu oleh *throughput* yang lebih baik dan *response time* yang lebih singkat. Selain itu, nilai dari parameter *Quality of Service* (QoS) dibutuhkan untuk menentukan seberapa baik kinerja jaringan tersebut, parameter *delay*, *jitter*, *packet loss*, dan *throughput*. Pada penelitian ini akan diimplementasikan tiga metode *load balancing* yaitu : ECMP, PCC, dan NTH. Metode yang direkomendasikan pada layanan publik adalah metode ECMP dan PCC, karena metode ECMP memiliki kualitas *throughput* yang baik yaitu 23.21% dan PCC memiliki kualitas *delay* yang baik yaitu 0.85 ms, lalu pada layanan *gaming*, metode yang direkomendasikan adalah metode NTH, karena memiliki hasil yang baik di setiap parameter, untuk *throughput* 6.52%, untuk nilai *delay* 0.70 ms, untuk nilai *jitter* 0.33 ms dan nilai *packet loss* sebesar 4.05%. Pada layanan *streaming* metode yang direkomendasikan adalah PCC dan NTH, PCC memiliki nilai rata – rata *throughput* yang baik, yaitu 26.6% sedangkan NTH memiliki rata – rata *delay* dan *packet loss* yang baik yaitu 0.88 ms dan 0.68 %.

**Kata kunci**— *resource*, performa jaringan, *Load Balancing*, *Quality of Service*

### I. PENDAHULUAN

Penggunaan internet memberikan banyak keuntungan dan kemudahan bagi kehidupan manusia, khususnya dalam menjalankan aktivitas sehari-hari, seperti berkomunikasi antar keluarga, teman, guru/dosen, dan kolega. Selain itu, internet juga memudahkan kita dalam mencari informasi, sarana pembelajaran, sarana menyelesaikan pekerjaan, dan sarana lainnya [1]. Maka dari itu kualitas internet dan ketersediaan internet menjadi aspek yang paling penting dalam menjalankan berbagai aktivitas pada masa ini.

Karena kemajuan pesat dalam teknologi dan informasi saat ini, diperlukan jumlah besar sumber daya (*resource*) dan data untuk mengakses konten di internet [2]. Oleh karena itu, untuk memenuhi kebutuhan ini, diperlukan metode untuk meningkatkan efisiensi dan kinerja jaringan. Dengan demikian, terciptanya jaringan yang efisien akan meningkatkan kemudahan, ketersediaan dan kenyamanan bagi pengguna untuk mengakses konten di internet.

Berdasarkan artikel yang berjudul “Kebutuhan Masyarakat Tinggi, Ketua DPD RI Minta Kualitas Koneksi Internet Ditingkatkan” yang diterbitkan oleh PikiranRakyat.com oleh [3] pada artikel tersebut disebutkan bahwa kebutuhan masyarakat terhadap ketersediaan jaringan dan kecepatan internet sangat tinggi, seiring transformasi digital di berbagai sektor.

Berdasarkan tingginya kebutuhan pengguna terhadap peningkatan kualitas internet dan tingginya kebutuhan *resource* dalam pengaksesan konten, maka diperlukan sebuah metode yang dapat memenuhi dan memaksimalkan kebutuhan *resource* pengguna dan diperlukan metode yang dapat meningkatkan efisiensi kinerja jaringan dengan melakukan perbandingan metode - metode *load balancing* untuk menentukan metode terbaik berdasarkan dari pengukuran QoS pada layanan berbasis konten

Melalui penerapan metode *load balancing*, maka beban *router* akan berkurang dan kestabilan jaringan akan tetap terjaga. Dengan terciptanya kestabilan jaringan maka akan meningkatkan kenyamanan pengguna dalam mengakses konten di internet.

Penelitian yang dilakukan merupakan lanjutan dari penelitian dari M.S. Adnan dan S. Ikhwan, dengan judul penelitian “Implementasi *Load Balancing* Metode ECMP, NTH dan PCC dengan Empat *Link* Internet Menggunakan Mikrotik”. Pada penelitian tersebut peneliti melakukan penelitian dengan melakukan pengujian *clientdownloadfile* di internet pada masing – masing metode. Hasil yang didapat adalah NTH dapat memberikan nilai yang paling baik di tiap parameter. Karena ruang lingkup penelitian yang dilakukan masih terbatas dan sederhana, oleh karena

itu diperlukan penelitian lebih lanjut dalam lingkup yang lebih kompleks.

II. METODOLOGI PENELITIAN

Pada bagian ini akan menjelaskan mengenai dasar teori yang digunakan selama proses penelitian.

A. Load Balancing

Metode *load balancing* digunakan untuk mendistribusikan beban trafik koneksi pada dua atau lebih jalur koneksi secara seimbang agar trafik berjalan dengan optimal, untuk memaksimalkan bandwidth *throughput* yang didapatkan dari *Internet Service Provider (ISP)* [5].

B. Quality of Service (QoS)

*Quality of Service (QoS)* merupakan metode pengukuran seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari satu layanan. Parameter QoS yang diamati pada penelitian ini adalah parameter *delay, throughput, packet loss, jitter* [2].

1) *Throughput*: *Throughput* merupakan kecepatan (rate) transfer data efektif, yang diukur dalam *Byte per second (Bps)* [2]. Dapat dikatakan bahwa *throughput* merupakan nilai *bandwith* yang sebenarnya (aktual) yang diukur pada satuan waktu tertentu dan pada kondisi jaringan tertentu. Nilai standarisasi *throughput* ditujukan pada Tabel 1

TABEL I  
NILAI STANDARISASI THROUGHPUT

Kategori	Through	Index
Terbaik	100%	4
Baik	75%	3
Cukup	50%	2
Kurang	<25%	1

Sumber : Standarisasi Kualitas Polimetode Telkom [6]

2) *Delay*: *Delay* merupakan 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 [2]. Standarisasi nilai *delay* ditujukan pada Tabel 2.

TABEL II  
STANDARISASI DELAY

Kategori	Delay	Inde
Sangat Bagus	< 150	4
Bagus	150 – 300	3
Sedang	300 – 450	2
Buruk	>450	1

Sumber : ETSI, 1999 [7]

3) *Jitter*: *Jitter* adalah variasi dalam panjang antrian (*delay*), waktu pengolahan data, dan juga dalam waktu penggabungan kembali paket – paket . *Jitter* disebut juga sebagai variasi *delay*, berhubungan erat dengan *latency* [2]. Nilai standarisasi *jitter* ditujukan pada Tabel 3.

TABEL III

STANDARISASI JITTER

Kategori	Peak Jitter	Inde
Sangat Bagus	0	4
Bagus	0 – 75	3
Sedang	75 – 125	2
Buruk	125 – 255	1

Sumber : ETSI, 1999 [7]

4) *Packet Loss*: *Packet loss* merupakan paket yang rusak atau hilang selama proses pengiriman berlangsung. Paket yang hilang dapat terjadi karena *collision* dan *congestion* pada jaringan [2] Nilai standarisasi *packet loss* ditujukan pada Tabel 4.

TABEL IV  
STANDARISASI PACKET LOSS

Kategori	Packet Loss	Inde
Sangat Bagus	0	4
Bagus	3	3
Sedang	15	2
Buruk	25	1

Sumber : ETSI, 1999 [7]

C. Layanan (Services)

Layanan yang dimaksud pada penelitian ini adalah sebuah kebiasaan atau kebutuhan pengguna terhadap konten yang diakses di internet. Terdapat tiga layanan berbasis konten, yaitu layanan *streaming*, publik, dan *gaming*.

Layanan *streaming* merupakan sebuah proses untuk mengakses atau mengunggah berkas/konten yang berada pada *server*/komputer dan memainkannya secara *real-time* tanpa harus mengunduh ataupun menyimpannya pada komputer *client* terlebih dahulu [8]. Terdapat tiga jenis layanan *streaming* [9] :

Layanan internet publik adalah sebuah ekosistem jaringan yang dimana pengguna yang terhubung dengan jaringan hanya mengakses konten – konten umum, seperti : social media, browsing, dan lainnya.

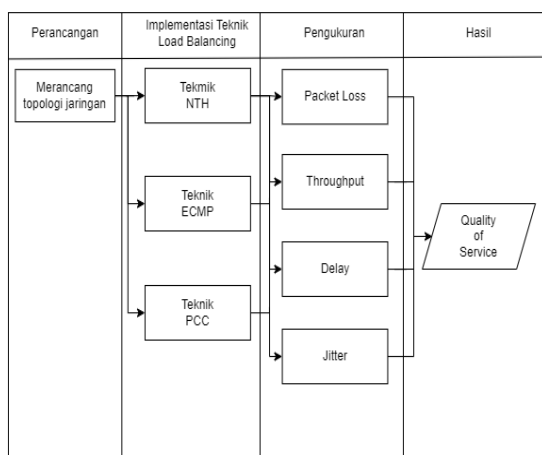
Layanan *Gaming* merupakan permainan di dunia maya yang di dalamnya terdapat interaksi *user* dengan menggunakan perangkat yang terhubung ke jaringan internet dan interaksi tersebut dapat dilakukan dengan adanya koneksi internet [10].

D. Alur Kerja Penelitian

Berdasarkan Gambar 1 akan dilakukan perancangan topologi, topologi terdiri dari router ISP, router utama, PC penguji, dan PC lainnya. Lalu akan dilakukan konfigurasi dasar pada topologi seperti : DHCP, *routes*, dan NAT. Setelah itu akan diimplementasikan metode – metode *load balancing* pada *router* utama. Adapun parameter – parameter yang diukur adalah: *throughput, packet loss, delay, dan jitter*.

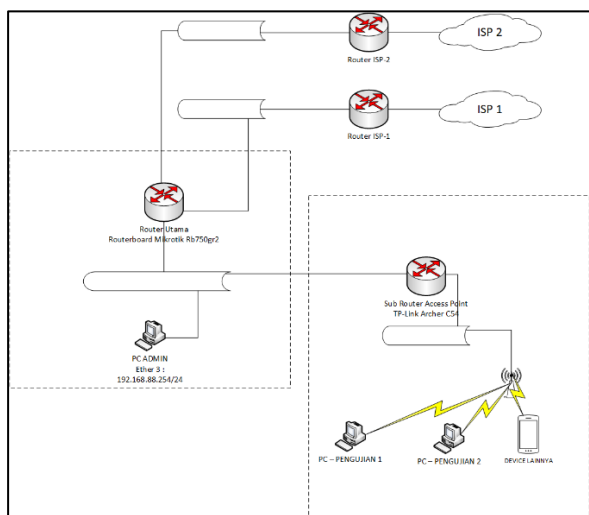
Pengujian dilakukan dengan menguji layanan *streaming*, publik dan *gaming*. Dari hasil pengukuran pada masing – masing layanan akan diklasifikasikan kualitasnya

berdasarkan hasil parameter jaringan dengan menggunakan metode *Quality of Service (QoS)*.



Gambar 1 Alur Kerja Penelitian

E. Perancangan Topologi Jaringan



Gambar 2 Rancangan Topologi

Router Utama berfungsi untuk implementasi metode *load balancing*, pada *network router* utama terdapat dua perangkat yang terhubung yaitu (lihat Gambar 2) :

PC-Admin berfungsi sebagai perantara untuk mengimplementasikan *load balancing* pada router utama.

Sub-Router Access Point berfungsi untuk memberi akses internet pada perangkat yang berada pada terhubung secara *wireless*.

Rancangan topologi ini diterapkan pada semua layanan pengujian (*publik, streaming, dan gaming*), pengujian dilakukan pada *network Sub Router Access Point* dengan rincian sebagai berikut :

Pengambilan data dan pengujian dilakukan dengan menggunakan PC-PENGUJIAN-1, perangkat tersebut terhubung dengan *router* secara *wireless* dan pengambilan data menggunakan sistem aplikasi yang telah dibangun.

PC-PENGUJIAN-2 berfungsi untuk mengakses layanan yang sedang diuji selama periode pengujian,

PC-PENGUJIAN-2 juga berfungsi untuk membebani jaringan untuk memaksimalkan performa metode *load balancing*.

DEVICE-PENGUJIAN LAINNYA berfungsi untuk melakukan pengambilan data dan mengakses konten internet serta sebagai beban jaringan untuk memaksimalkan kinerja *load balancing*.

F. Implementasi Teknik Load Balancing

Tahap selanjutnya akan diimplementasikan metode – metode *load balancing*, yaitu metode NTH, ECMP, dan PCC.

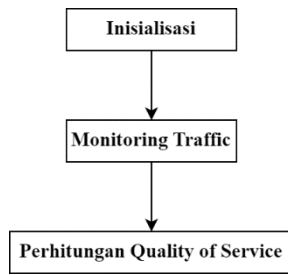
1) *Nth*: *Nth* adalah metode *load balancing* yang membentuk suatu deret tertentu (*Nth*), yang akan digunakan sebagai sistem antrian di dalam *firewall mangle*. *Nth* digunakan dalam suatu deret yang terdiri dari *every* dan *packet* yang akan direalisasikan dalam suatu deret integer. Metode *load balancing* ini akan menunjukkan paket data yang masuk sebagai variabel *n* dalam tipe data integer. Lalu paket tersebut akan diproses berdasarkan nilai variabel *n* yang didapat. Berdasarkan penelitian yang dilakukan oleh Hidayatullah, M. G. [11], ada beberapa variabel penting dalam *Nth* yaitu:

- *Every* - Nilai *every* adalah jumlah kelompok yang akan dihasilkan. Dengan demikian, jika terdapat dua jalur koneksi, paket akan dibagi menjadi dua kelompok yang masing-masing akan dibagi ke dua koneksi tersebut, sehingga nilai *every* adalah dua.
- *Packet* - Paket terdiri dari jumlah koneksi yang akan dilabelkan. Jika ada dua kelompok, tentunya harus ada dua *rule mangle*. Dalam satu aturan, angka untuk *every* harus sama, tetapi angka paket harus berbeda.

2) *Equal Cost Multiple Path (ECMP)*: *Equal Cost Multi Path (ECMP)* adalah pemilihan jalur secara bergantian pada *gateway* dan merupakan salah satu metode *load balancing* yang paling cepat membagi beban trafik menuju internet [12]. Metode ECMP memiliki kemampuan untuk membagi trafik jaringan di antara beberapa jalur yang memiliki nilai yang sama, dengan membagi trafik di antara jalur-jalur yang tersedia secara merata [13].

3) *Per Connection Classifier (PCC)*: *Per Connection Classifier (PCC)* adalah metode yang cara kerjanya menspesifikasikan suatu paket menuju *gateway* suatu jaringan tertentu. PCC membagi paket yang keluar dan masuk *router* menjadi berbagai kelompok. Paket ini diklasifikasikan berdasarkan *set-address, scr-port, dan port* [14]. Sistem akan mengingat jalur *gateway* yang dilewati pada awal trafik koneksi, sehingga paket-paket data berikutnya yang masih berhubungan dengan paket data sebelumnya akan dilewatkan pada jalur *gateway* yang sama [15]. Apabila paket tidak memiliki *gateway*, maka akan dilakukan proses *hashing*, proses *hashing* bertujuan untuk memberikan informasi *gateway* yang akan dilalui bagi paket yang tidak memiliki asal *gateway*.

G. Pengukuran Parameter



Gambar 3 Flowchart Pengukuran Parameter

Pada Gambar 3 merupakan tahapan proses sistem, dijelaskan sebagai berikut :

1) *Inisialisasi*: Sistem akan mengumpulkan informasi mengenai perangkat keras jaringan yang terpasang pada perangkat *user*. Setelah informasi didapatkan sistem akan menampilkan informasi tersebut pada sebuah daftar *interface* yang dapat user pilih untuk proses monitoring.

Lalu sistem akan menyimpan informasi mengenai *interface* yang *user* pilih untuk digunakan pada proses selanjutnya.

2) *Monitoring Trafik*: Pada proses ini sistem akan mengumpulkan informasi mengenai perangkat keras jaringan yang terpasang pada perangkat *user*.

Setelah informasi didapatkan sistem akan menampilkan informasi tersebut pada sebuah daftar *interface* yang dapat *user* pilih untuk proses monitoring.

Lalu sistem akan menyimpan informasi mengenai *interface* yang *user* pilih untuk digunakan pada proses selanjutnya

3) *Perhitungan Quality of Service*: Pada proses ini akan menjelaskan mengenai perhitungan variabel paket, variabel paket ini bertujuan untuk menghitung parameter – parameter QoS. Variabel paket yang akan dihitung adalah :

- *Total Delay*  
Variabel ini didapatkan dari informasi waktu kedatangan dari paket satu ke paket selanjutnya.
- *Time Span*  
Time span merupakan lama waktu pengamatan monitoring.
- *Received Packets*  
*Received packet* merupakan variabel yang didapatkan dari jumlah paket yang didapatkan oleh *interface* (upload).
- *Sent Packets*  
*Sent packets* merupakan variabel yang didapatkan dari jumlah paket yang dikirim dari sisi *user* (upload).
- *Total Data*  
Total data merupakan variabel yang didapatkan dari akumulasi besar paket selama monitoring.
- *Variasi Delay*  
Variasi *delay* didapatkan dari selisih kedatangan paket selanjutnya dan paket sebelumnya.

Hal yang pertama dilakukan pada proses perhitungan QoS adalah menentukan *log* hasil monitoring yang akan dianalisis. Lalu sistem akan melakukan analisis dengan

melakukan proses *parsing* pada *log* yang telah ditentukan, proses dilakukan dengan cara mengurai informasi pada *log*, lalu melakukan *parsing* pada bagian yang dibutuhkan untuk perhitungan variabel – variabel, mencakup : *total packet loss*, *process time (time span)*, *packet received*, *variable delays*, *total packets*, dan *total delays*.

Setelah nilai dari variabel – variabel paket telah diketahui maka proses selanjutnya adalah perhitungan parameter QoS yang mencakup : *throughput*, *delay*, *jitter*, dan *packet loss*.

*Packet loss* dapat dihitung dengan menggunakan variabel paket *total packet loss*, dan *total packets*. Variabel *Total packet loss* merupakan perhitungan dari pengurangan paket yang dikirim dengan paket yang diterima. Dari dua variabel paket tersebut maka dapat dihitung nilai *packet loss* menggunakan rumus berikut :

$$Packet\ Loss = \frac{(Paket\ data\ dikirim - paket\ data\ diterima)}{paket\ data\ yang\ dikirim} \times 100\% \quad (1)$$

Parameter *throughput* dapat dihasilkan dengan menggunakan variabel paket *total data*. Variabel *total data* merupakan variabel yang menghitung besar data selama monitoring. Adapun rumus untuk menghitung *throughput* adalah sebagai berikut :

$$Throughput\ (\%) = \left( \frac{Bytes\ yang\ dikirim}{lama\ pengamatan} : Bandwith \right) \times 100 \quad (2)$$

Dalam menghitung parameter *delay* dapat dihasilkan dengan menggunakan variabel paket *total delay* dan *total data packets* (received packets). Berikut merupakan perhitungan *delay* :

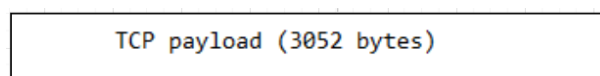
$$Delay = \frac{Total\ Delay}{Paket\ yang\ diterima} \quad (3)$$

Terakhir, untuk menghitung *jitter* dapat dihasilkan dengan menggunakan variabel paket variabel *delay*, dan *total packets* (received packets). Berikut merupakan perhitungan *jitter* :

$$Jitter = \frac{Total\ rata-rata\ variasi\ delay}{total\ paket\ yang\ diterima} \quad (4)$$

#### H. Parsing

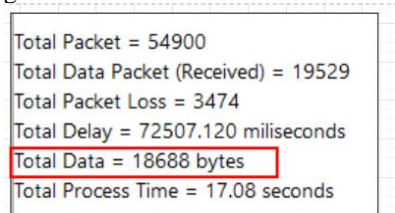
Setelah dilakukannya monitoring jaringan, maka didapatkan hasil monitoring berupa *log*. Karena *log* yang didapatkan masih “mentah” maka harus dilakukan suatu proses agar *log* tersebut dapat bermanfaat, proses yang dibutuhkan merupakan proses *parsing*. *Parsing* merupakan proses untuk memecah kata atau kalimat yang bertujuan untuk mengekstraksi sebuah informasi yang dibutuhkan. Pada penelitian ini proses *parsing* dilakukan untuk mendapatkan nilai – nilai variabel paket. Nilai tersebut didapatkan dengan melakukan proses *parsing* pada *log* hasil monitoring. Pada Gambar 3 merupakan contoh bagian *log* yang akan di-*parsing*



Gambar 4 Potongan log TCP payload

Gambar 4 merupakan potongan *log* dari bagian TCP *payload* yang akan digunakan untuk proses *parsing* dan

perhitungan variabel paket, potongan tersebut berfungsi untuk menghitung banyaknya data yang diterima dan dikirim selama proses monitoring (*Bytes*). Nilai dari TCP *payload* akan ditambahkan/*append* dengan nilai TCP *payload* pada *dataframe* yang lainnya. Pada Gambar 5 adalah *output* dari hasil perhitungan TCP *payload* untuk menghitung *total data*.



Gambar 5 Hasil parsing TCP payload

Hasil dari variabel total data akan digunakan untuk menghitung nilai dari parameter QoS yaitu *throughput*.

### III. HASIL DAN PEMBAHASAN

#### A. Hasil Pengujian

Hasil pengujian akan mencakup pengujian pada layanan publik, layanan *gaming*, dan layanan *streaming*. Hasil pada pengujian ini mencakup pengujian data pada total rentang waktu 1.15 jam lalu pengambilan data dilakukan setiap 15 menit yang dilakukan pada 5 sesi setiap layanannya, hal ini bertujuan untuk melihat trendline dari perubahan parameter pada rentang waktu yang lebih singkat.

#### B. Skenario Monitoring pada Layanan Publik

Pengujian pada layanan publik dilakukan dengan cara sebagai berikut :

- Dilakukan pengambilan data sebanyak 5 sesi pengujian, dimana lama sesi adalah 1.15 jam dan pengambilan data setiap 15 menit
- Konten yang diuji pada platform Youtube Shorts selama waktu pengujian selama 5 sesi.
- *User* yang mengakses layanan yang sama adalah 10 *user*, dengan *total user* 15 *user*
- Parameter yang diprioritaskan pada penelitian ini adalah nilai dan konsistensi dari *throughput* dan *delay* [16].

#### C. Skenario Monitoring pada Layanan Gaming

Pengujian pada layanan *gaming* dilakukan dengan cara sebagai berikut :

- Dilakukan pengambilan data sebanyak 5 sesi pengujian, dimana lama sesi adalah 1.15 jam dan pengambilan data setiap 15 menit.
- Aplikasi yang digunakan untuk pengujian pada *game* Genshin Impact.
- Variabel yang diprioritaskan pada pengujian ini adalah nilai dan konsistensi data *throughput*, *delay*, *jitter*, dan *packet loss* [10].
- *User* yang mengakses layanan yang sama adalah 10 *user*, dengan *total user* 15 *user*.

#### D. Hasil Monitoring pada Layanan Streaming

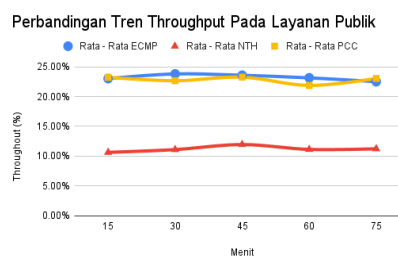
Pengujian pada layanan *streaming* dilakukan dengan cara sebagai berikut :

- Dilakukan pengambilan data sebanyak 5 sesi pengujian, dimana lama sesi adalah 1.15 jam dan pengambilan data setiap 15 menit
- Pengujian dilakukan dengan melakukan *live streaming*, *live streaming* dilakukan pada platform Youtube
- *Live streaming* dilakukan dengan pada kualitas 720p
- Variabel yang diprioritaskan adalah *throughput*, *delay*, dan *packet loss* [9] .
- *User* yang mengakses layanan yang sama adalah 10 *user*, dengan *total user* 15 *user*.

#### E. Pembahasan

Pada bagian ini akan menjelaskan mengenai analisis pada masing – masing metode *load balancing* berdasarkan layanan yang diuji. Analisis dilakukan dengan membandingkan hasil dari parameter – parameter QoS dari masing – masing pengujian.

##### 1. Perbandingan Trend Monitoring pada Layanan Publik

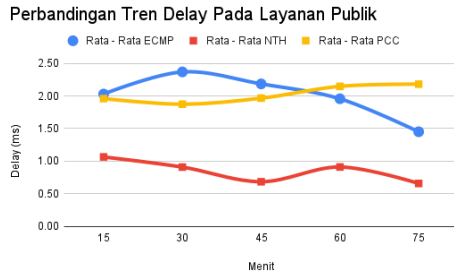


Gambar 6 Perbandingan Tren *Throughput* pada Layanan Publik

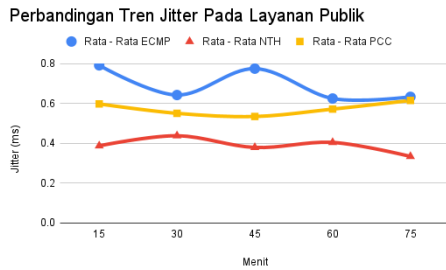
Berdasarkan Gambar 6 metode ECMP mampu memberikan *throughput* rata-rata yang signifikan, yakni sekitar 23.21%. Keberhasilan ini adalah hasil dari pendekatan ECMP yang efisien dalam membagi beban *trafik* ke berbagai jalur koneksi, dengan tujuan untuk mengoptimalkan *throughput*. Meskipun demikian, penting untuk dicatat bahwa penggunaan metode ECMP juga menghasilkan tingkat rata-rata *delay* dan *packet loss* yang cenderung lebih tinggi. Hal ini disebabkan oleh perubahan konstan dalam pemilihan jalur koneksi, yang memaksa terjadinya transmisi ulang paket dan berdampak pada peningkatan *delay* dan *packet loss*. Sedangkan rendahnya rata – rata metode NTH terjadi dikarenakan algoritma NTH yang berbasis antrian tidak dapat memaksimalkan *throughput* yang dibutuhkan selama pengujian berlangsung.

Pada Gambar 7 terlihat bahwa metode NTH menunjukkan kinerja yang lebih baik dibandingkan dengan dua metode lainnya dengan rata – rata 0.86 ms. Keunggulan ini karena algoritma NTH mampu menyusun alokasi *trafik* secara sistematis melalui algoritma antrian. Penerapan antrian ini berperan penting dalam mengurangi risiko *collision* dan *congestions*, yang kerap menjadi penyebab utama dari *delay* dan *packet loss*. Sedangkan pada metode PCC, metode PCC memiliki rata – rata *delay*

yang paling tinggi dibandingkan dua metode lainnya. Hal ini disebabkan karena

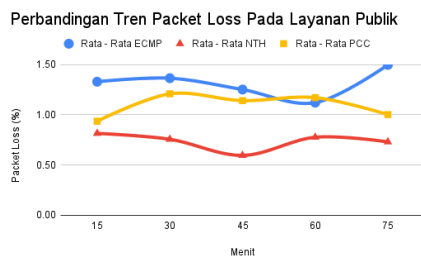


Gambar 7 Perbandingan Tren Delay pada Layanan Publik



Gambar 8 Perbandingan Tren Jitter pada Layanan Publik

Pada Gambar 8 metode NTH berhasil mencapai rata-rata jitter yang lebih baik yaitu 0.338 ms dibandingkan dengan metode lainnya. Hal ini menunjukkan bahwa NTH memberikan konsistensi dalam kualitas jaringan, yang sangat penting dalam layanan yang memerlukan stabilitas. Sedangkan metode ECMP memiliki rata – rata jitter yang tinggi dikarenakan memiliki rata – rata delay yang tinggi [17].



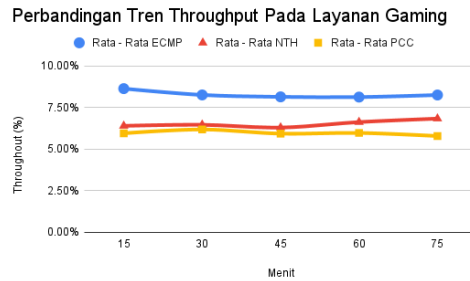
Gambar 9 Perbandingan Tren Packet Loss pada Layanan Publik

Pada Gambar 9 dapat diketahui dalam hal packet loss, metode NTH mampu memberikan tingkat packet loss yang baik dengan rata-rata 0.73 %. Kemampuan algoritma NTH untuk mengelola pengiriman paket dengan efisien berkontribusi pada tingkat packet loss yang lebih rendah, yang sangat penting dalam mempertahankan integritas data dalam layanan ini.

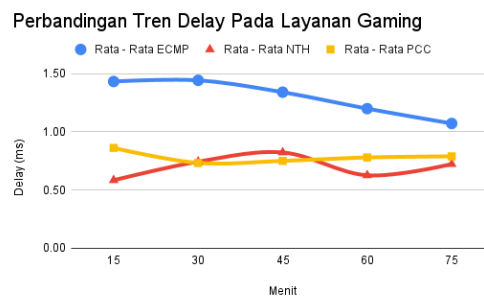
## 2. Perbandingan Trend Monitoring pada Layanan Gaming

Dalam analisis throughput pada Gambar 10, metode ECMP memberikan rata-rata throughput sekitar 8.28%. Keunggulan ini terjadi karena pendekatan ECMP yang efisien dalam pembagian beban ke berbagai jalur koneksi, dengan tujuan memaksimalkan throughput. Namun, perlu

dicatat bahwa meskipun throughput tinggi, metode ECMP kurang efektif dalam menjaga nilai parameter lainnya, seperti delay, jitter, dan packet loss. Hal ini disebabkan oleh perubahan jalur koneksi yang sering terjadi dalam ECMP, yang mengharuskan transmisi ulang paket dan mengakibatkan peningkatan delay serta packet loss.

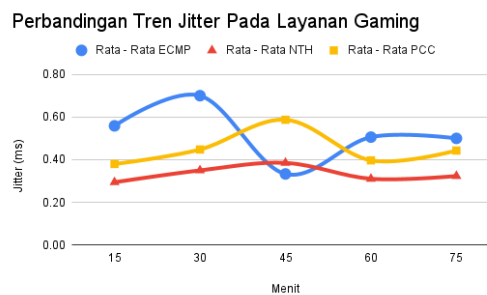


Gambar 10 Perbandingan Tren Throughput pada Layanan Gaming



Gambar 11 Perbandingan Tren Delay pada Layanan Gaming

Pada Gambar 11, dapat dilihat bahwa ECMP memiliki rata – rata grafik delay yang lebih tinggi dibandingkan dua metode lainnya, hal ini disebabkan karena pada algoritma ECMP terdapat proses perubahan jalur koneksi yang dapat menyebabkan koneksi internet terputus dan menyebabkan kenaikan nilai delay dan packet loss. Nilai delay yang tinggi juga dapat disebabkan oleh kenaikan nilai packet loss yang dihasilkan, karena berdasarkan hubungan antar parameter QoS [17] saat nilai delay atau packet loss tinggi maka akan mempengaruhi nilai throughput menjadi rendah, sedangkan saat nilai packet loss yang tinggi maka akan menyebabkan nilai delay tinggi dan nilai throughput yang rendah.

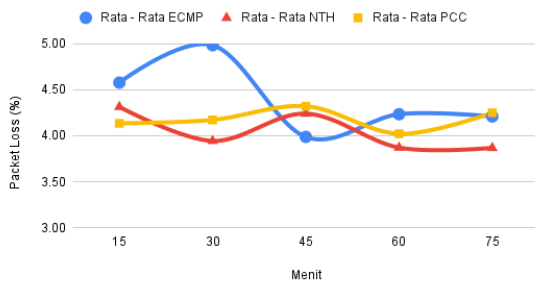


Gambar 12 Perbandingan Tren Jitter pada Layanan Gaming

Pada Gambar 12, dapat dilihat bahwa grafik NTH memiliki rata – rata delay yang lebih baik dibandingkan dengan dua metode lainnya yaitu dengan rata – rata 0.338 ms. Hal ini dapat disebabkan karena pada metode NTH,

metode NTH memiliki rata – rata *delay* yang rendah. Berdasarkan hubungan antar parameter QoS [17], nilai *delay* akan berbanding lurus dengan nilai *delay*. Maka dari itu rata – rata nilai *jitter* tidak akan berbeda jauh dengan rata – rata nilai *delay*.

Perbandingan Tren Packet Loss Pada Layanan Gaming

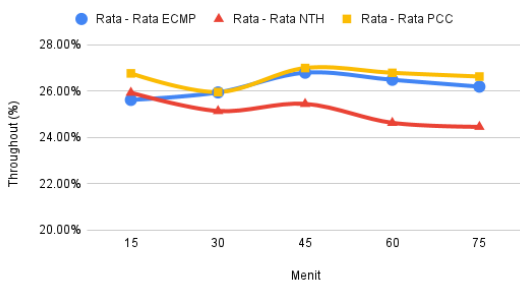


Gambar 13 Perbandingan Tren *Packet Loss* pada Layanan *Gaming*

Pada Gambar 13, metode yang memiliki rata – rata nilai *packet loss* yang baik adalah NTH dengan nilai 0.73 %. Hal ini dapat disebabkan karena metode NTH memiliki rata – rata nilai *delay* yang rendah. Apabila nilai *delay* rendah maka akan memperkecil kemungkinan terjadinya *packet loss*. Dapat dilihat pada metode ECMP saat pengujian menit ke-30, terjadi kenaikan nilai *packet loss* menjadi 4.58 % yang dimana menyebabkan kenaikan nilai *delay* pada waktu pengujian yang sama yaitu sebesar 1.44 ms. Pada grafik ECMP saat grafik mengalami penurunan di menit ke-45 menjadi 3.99 %, maka nilai *delay* juga menurun menjadi 1.34 ms

### 3. Perbandingan Trend Monitoring pada Layanan Streaming

Perbandingan Tren Throughput Pada Layanan Streaming



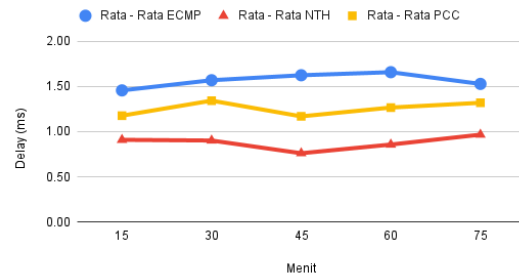
Gambar 14 Perbandingan Tren *Throughput* pada Layanan *Streaming*

Berdasarkan grafik pada Gambar 14, metode PCC dapat memberikan rata – rata *throughput* yang tinggi dibandingkan dua metode lainnya dengan nilai *throughput* 26.62 %, hal ini dikarenakan cara kerja metode PCC yang dapat secara langsung menyalurkan paket secara cepat ke jalur asalnya. Namun metode PCC memiliki kelemahan, dimana dapat terjadi overload pada salah satu jalur koneksi yang menyebabkan collision dan congestion pada paket, dimana hal tersebut akan mengakibatkan tingginya nilai *delay*, dan *packet loss* apabila menggunakan metode pada waktu yang lama. Lalu pada grafik *delay*, *jitter*, dan *packet loss*.

Pada Gambar 15, ECMP memiliki nilai rata – rata yang lebih besar dibandingkan dua metode lainnya dengan nilai 1.57 ms, hal ini dapat disebabkan karena algoritma ECMP

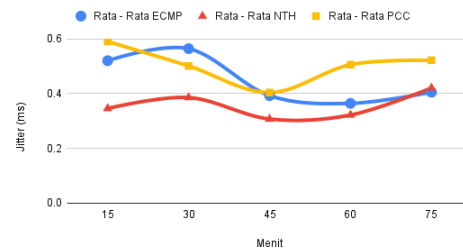
yang memungkinkan perubahan jalur antar koneksi yang berdampak meningkatnya nilai *packet loss* dan *delay*, karena saat terjadinya perubahan jalur koneksi, lalu lintas pada *router* akan *down* dalam sementara waktu yang menyebabkan meningkatnya nilai *delay*, dan *packet loss*. Sebaliknya, metode NTH memiliki rata – rata nilai *delay* yang rendah yaitu 0.88 ms, karena rata – rata *packet loss* yang dihasilkan metode NTH rendah.

Perbandingan Tren Delay Pada Layanan Streaming



Gambar 15 Perbandingan Tren *Delay* pada Layanan *Streaming*

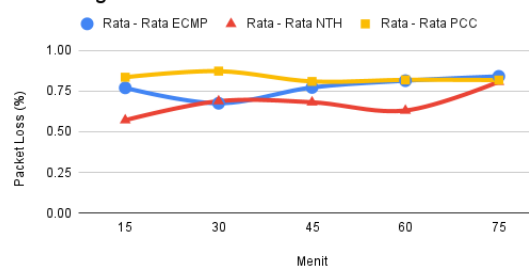
Perbandingan Tren Jitter Pada Layanan Streaming



Gambar 16 Perbandingan Tren *Jitter* pada Layanan *Streaming*

Pada Gambar 16, dapat dilihat bahwa metode NTH memiliki rata – rata *jitter* yang lebih baik dibandingkan dua metode lainnya yaitu 0.356 ms. Hal ini disebabkan karena metode NTH memiliki rata – rata nilai *delay* yang rendah juga maka akan berdampak kepada nilai *jitter* [17].

Perbandingan Tren Packet Loss Pada Layanan Streaming



Gambar 17 Perbandingan Tren *Packet Loss* pada Layanan *Streaming*

Pada Gambar 17, metode yang memiliki rata – rata *packet loss* yang baik adalah NTH dengan rata – rata 0.68 %. Hal ini dapat disebabkan karena kemampuan algoritma NTH yang dapat membagi paket secara sistematis yang dapat menghasilkan nilai *delay* dan *packet loss* yang rendah.

### F. Penilaian Quality of Service

Pada bagian ini akan dilakukan penilaian parameter Quality of Service (QoS) berdasarkan nilai dari rata – rata

yang telah didapat pada masing – masing pengujian metode *load balancing* yang dilakukan pada setiap layanan. Penilaian akan dilakukan untuk setiap parameter QoS, yaitu *throughput*, *delay*, *jitter*, dan *packet loss*.

G. Penilaian Parameter Quality of Service pada Layanan Publik

1. Penilaian Quality of Service Metode ECMP

TABEL V  
PENILAIAN QOS METODE ECMP PADA LAYANAN PUBLIK

Parameter	ECMP	Quality of Service
Throughput (%)	23.21	1 (Kurang Baik)
Delay (ms)	2.00	4 (Sangat Bagus)
Jitter (ms)	0.69	4 (Sangat Bagus)
Packet Loss (%)	1.31	4 (Sangat Bagus)

Berdasarkan Tabel 5 dapat disimpulkan bahwa pada layanan publik, ECMP mendapatkan nilai QoS 1 (Kurang Baik) untuk nilai *throughput* dengan nilai sebesar 23.21%, pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 2.00 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.69 ms. Lalu pada parameter *packet loss* mendapatkan nilai 4 (Sangat Bagus) dengan nilai sebesar 1.31 %.

2. Penilaian Quality of Service Metode NTH

TABEL VI  
PENILAIAN QOS METODE NTH PADA LAYANAN PUBLIK

Parameter	NTH	Quality of Service
Throughput (%)	11.21	1 (Kurang Baik)
Delay (ms)	0.85	4 (Sangat Bagus)
Jitter (ms)	0.388	4 (Sangat Bagus)
Packet Loss (%)	0.73	4 (Sangat Bagus)

Berdasarkan Tabel 6 dapat disimpulkan bahwa pada layanan publik, NTH mendapatkan nilai QoS 1 (Kurang Baik) untuk nilai *throughput* dengan nilai sebesar 11.21% , pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 0.85 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.388 ms. Lalu pada parameter *packet loss* mendapatkan nilai 4 (Sangat Bagus) dengan nilai sebesar 0.73 %.

3. Penilaian Quality of Service Metode PCC

Berdasarkan Tabel 7 dapat disimpulkan bahwa pada layanan publik, PCC mendapatkan nilai QoS 2 (Cukup) untuk nilai *throughput* dengan nilai sebesar 22.81%, pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 2.03 ms, pada parameter *jitter* mendapatkan nilai 3 (Sangat Bagus) dengan nilai *jitter*

0.573 ms. Lalu pada parameter *packet loss* mendapatkan nilai 4 (Sangat Bagus) dengan nilai sebesar 1.09 %.

TABEL VII  
PENILAIAN QOS METODE PCC PADA LAYANAN PUBLIK

Parameter	PCC	Quality of Service
Throughput (%)	22.81	1 (Kurang Baik)
Delay (ms)	2.03	4 (Sangat Bagus)
Jitter (ms)	0.573	3 (Sangat Bagus)
Packet Loss (%)	1.09	4 (Sangat Bagus)

H. Penilaian Quality of Service pada Layanan Gaming

1. Penilaian Quality of Service Metode ECMP

TABEL VIII  
PENILAIAN QOS METODE ECMP PADA LAYANAN GAMING

Parameter	ECMP	Quality of Service
Throughput (%)	8.28	1 (Kurang Baik)
Delay (ms)	1.30	4 (Sangat Bagus)
Jitter (ms)	0.52	4 (Sangat Bagus)
Packet Loss (%)	4.40	3 (Bagus)

Berdasarkan Tabel 8 dapat disimpulkan bahwa pada layanan *gaming*, ECMP mendapatkan nilai QoS 1 (Kurang Baik) untuk nilai *throughput* dengan nilai sebesar 8.28 % , pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 1.30 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.52 ms. Lalu pada parameter *packet loss* mendapatkan nilai 3 (Bagus) dengan nilai sebesar 4.40 %.

2. Penilaian Quality of Service Metode NTH

TABEL IX  
PENILAIAN QOS METODE NTH PADA LAYANAN GAMING

Parameter	NTH	Quality of Service
Throughput (%)	6.52	1 (Kurang Baik)
Delay (ms)	0.70	4 (Sangat Bagus)
Jitter (ms)	0.33	4 (Sangat Bagus)
Packet Loss (%)	4.05	3 (Bagus)

Berdasarkan Tabel 9 dapat disimpulkan bahwa pada layanan *gaming*, NTH mendapatkan nilai QoS 1 (Kurang Baik) untuk nilai *throughput* dengan nilai sebesar 6.52 % , pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 0.70 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.33 ms. Lalu pada parameter *packet loss* mendapatkan nilai 3 (Bagus) dengan nilai sebesar 4.05 %.

3. Penilaian Quality of Service Metode PCC



Berdasarkan Tabel 10 dapat disimpulkan bahwa pada layanan *gaming*, PCC mendapatkan nilai QoS 1 (Kurang Baik) untuk nilai *throughput* dengan nilai sebesar 5.96%, pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 0.78 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.45 ms. Lalu pada parameter *packet loss* mendapatkan nilai 3 (Bagus) dengan nilai sebesar 4.18 %.

TABEL X  
PENILAIAN QOS METODE PCC PADA LAYANAN GAMING

Parameter	PCC	Quality of Service
Throughput (%)	5.96	1 (Kurang Baik)
Delay (ms)	0.78	4 (Sangat Bagus)
Jitter (ms)	0.45	4 (Sangat Bagus)
Packet Loss (%)	4.18	3 (Bagus)

1. Penilaian Quality of Service pada Layanan Streaming

1. Penilaian Quality of Service Metode ECMP

TABEL XI  
PENILAIAN QOS METODE ECMP PADA LAYANAN STREAMING

Parameter	ECMP	Quality of Service
Throughput (%)	26.20	2 (Baik)
Delay (ms)	1.57	4 (Sangat Bagus)
Jitter (ms)	0.45	4 (Sangat Bagus)
Packet Loss (%)	0.77	4 (Sangat Bagus)

Berdasarkan Tabel 11 dapat disimpulkan bahwa pada layanan *streaming*, ECMP mendapatkan nilai QoS 2 (Baik) untuk nilai *throughput* dengan nilai sebesar 26.20%, pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 1.57 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.45 ms. Lalu pada parameter *packet loss* mendapatkan nilai 3 (Bagus) dengan nilai sebesar 0.77 %.

2. Penilaian Quality of Service Metode NTH

TABEL XII  
PENILAIAN QOS METODE NTH PADA LAYANAN STREAMING

Parameter	NTH	Quality of Service
Throughput (%)	25.12	2 (Baik)
Delay (ms)	0.88	4 (Sangat Bagus)
Jitter (ms)	0.356	4 (Sangat Bagus)
Packet Loss (%)	0.68	4 (Sangat Bagus)

Berdasarkan Tabel 12 dapat disimpulkan bahwa pada layanan *streaming*, NTH mendapatkan nilai QoS 2 (Baik) untuk nilai *throughput* dengan nilai sebesar 25.12%, pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 0.88 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter*

0.356 ms. Lalu pada parameter *packet loss* mendapatkan nilai 3 (Bagus) dengan nilai sebesar 0.68 %.

3. Penilaian Quality of Service Metode PCC

TABEL XIII  
PENILAIAN QOS METODE PCC PADA LAYANAN STREAMING

Parameter	PCC	Quality of Service
Throughput (%)	26.62	2 (Baik)
Delay (ms)	1.25	4 (Sangat Bagus)
Jitter (ms)	0.50	4 (Sangat Bagus)
Packet Loss (%)	0.83	4 (Sangat Bagus)

Berdasarkan Tabel 13 dapat disimpulkan bahwa pada layanan *streaming*, NTH mendapatkan nilai QoS 4 (Terbaik) untuk nilai *throughput* dengan nilai sebesar 17.03%, pada parameter *delay* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *delay* 1.25 ms, pada parameter *jitter* mendapatkan nilai 4 (Sangat Bagus) dengan nilai *jitter* 0.50 ms. Lalu pada parameter *packet loss* mendapatkan nilai 4 (Sanat Bagus) dengan nilai sebesar 0.83 %.

IV. KESIMPULAN

Pada layanan publik metode akan dinilai berdasarkan parameter *throughput* dan *delay* [16], pada layanan ini metode yang direkomendasikan adalah ECMP dan PCC. Metode ECMP dapat memberikan rata – rata *throughput* yang paling baik dibandingkan dua metode lainnya yaitu 23.21%, Sedangkan Metode PCC memiliki rata – rata *delay* sebesar 2.03 ms dengan nilai QoS sebesar 4 (Sangat Bagus) dan memiliki rata – rata *throughput* yang baik sebesar 22.81%. Metode NTH tidak direkomendasikan pada layanan ini karena, metode NTH memiliki rata – rata *throughput* yang kecil. Hal ini dapat berdampak tidak dapatnya NTH dalam memaksimalkan *throughput* pada karakteristik layanan publik yang memiliki *packet size* yang bervariasi.

Pada layanan *gaming* metode akan dinilai berdasarkan parameter *throughput*, *delay*, *jitter*, dan *packet loss* [10]. Pada layanan *gaming* dapat disimpulkan bahwa pada layanan ini metode yang direkomendasikan adalah NTH. Berdasarkan karakteristik layanan *gaming*, dimana layanan *gaming* membutuhkan kestabilan nilai parameter seiring dengan berjalannya waktu pengujian, metode NTH dapat meminimalkan *delay*, dan *packet loss*. Nilai *throughput* yang kecil disebabkan karena kebutuhan pengujian layanan *gaming* hanya membutuhkan besar bandwidth yang kecil yakni 3 Mbps.

Pada layanan *streaming* dapat disimpulkan bahwa pada layanan ini metode yang direkomendasikan adalah PCC dan NTH. Metode PCC dapat memberikan rata – rata *throughput* sebesar 26.62% dan nilai QoS *throughput* adalah 2 (Baik). Sedangkan untuk metode NTH, metode NTH dapat memberikan nilai rata – rata *delay* yang lebih baik yaitu 0.88 ms dan nilai QoS *delay* adalah 4 (Sangat Bagus) dan dapat memberikan nilai rata – rata *packet loss* yang baik yaitu 0.68 % dengan nilai QoS 4 (Sangat Bagus ). Layanan *streaming* memiliki karakteristik dimana layanan tersebut akan mengirimkan paket dalam ukuran

besar pada satu waktu, hal ini tidak akan memberikan dampak yang signifikan pada metode PCC yang menerapkan algoritma *hashing* ataupun pada metode NTH yang menerapkan algoritma antrian. Namun layanan dengan karakteristik serupa tidak direkomendasikan untuk metode ECMP, karena metode ECMP akan melakukan perpindahan jaringan saat mencapai *threshold* di salah satu jalur koneksi yang mengakibatkan nilai *throughput* tidak dapat dipertahankan yang akan mengakibatkan *delay* dan *packet loss*.

#### REFERENSI

- [1] R. A. Halimaatussa, "Analisis QoS Video dan Audio Streaming Dengan RTMP ( Real Time Messaging Protokol )," pp. 77-90, 2020.
- [2] M. S. Adnan and S. Ikhwan, "Implementasi Load Balancing Metode ECMP , NTH dan PCC dengan Empat Link Internet Menggunakan Mikrotik," pp. 308-314, 2018.
- [3] Native, "Kebutuhan Masyarakat Tinggi, Ketua DPD RI Minta Kualitas Koneksi Internet Ditingkatkan," *Pikiran Rakyat*, 10 Maret 2022. [Online]. Available: <https://www.pikiran-rakyat.com/nasional/pr-013942182/kebutuhan-masyarakat-tinggi-ketua-dpd-ri-minta-kualitas-koneksi-internet-ditingkatkan>. [Accessed 20 Juli 2022].
- [4] A. F. Aprianto and T. T. W S, "Analisa Quality of Service Pada Teknik Load Balancing Menggunakan Metode PCC Dengan Dedicated Bandwidth," 2019.
- [5] A. E. Ramatu, "Computer network design and implementation using load balancing technique with per connection classifier (PCC) method based on MikroTik router," *Procedia Computer Science*, vol. 216, pp. 103-111, 2023.
- [6] P. R. Ledesma, "Kualitas Layanan Pada Sistem Telekomunikasi," *Riset Politeknik Telkom*, 2013.
- [7] ETSI, "Telecommunications and Internet Protocol Harmonization Over Internet (TIPHON): General Aspect of Quality of Service," 1999.
- [8] S. M. Ulfa and R. Primananda, "Pengaruh Data Video Streaming pada Kinerja Protokol Optimized Link State Routing ( OLSR ) menggunakan Teknologi Wireless Mesh Network," *Pengembangan Teknologi Informasi dan ilmu komputer*, pp. 6881-6886, 2019.
- [9] E. Kurniawan and A. Sani, "Analisis Kualitas Real Time Video Streaming Terhadap Bandwidth Jaringan Yang Tersedia," *Singuda ENSIKOM*, vol. 9, no. 2, pp. 92-96, 2014.
- [10] A. Tantoni and M. T. A. Zaen, "ANALISIS KEBUTUHAN KECEPATAN BANDWIDTH GAME ONLINE (Free fire, Mobile Legends, Pubg mobile)," *Jurnal Informatika dan Rekayasa Elektronik*, p. 81, 2019.
- [11] M. G. Hidayatullah, "Implementasi Load Balancing Metode Nth Untuk Distribusi Trafik Pada Smk Maqna ` U1 Ulum Sukowono Menggunakan Mikrotik," 2020.
- [12] S. Indratno, "Implementasi Load Balancing Mikrotik Menggunakan Metode Ecmp Pada STIE Gentiaras Bandar Lampung," *Jurnal Teknologi Pintar*, vol. 3, no. 1, pp. 1-19, 2022.
- [13] A. Verdiansyah, "Network Design Using Load Balancing Equal Cost Multi Path ( ECMP ) Method At Lexima Consultant Office," *SAGA: Journal of Technology and Information Systems*, vol. 1, no. 2, pp. 31-36, 2023.
- [14] K. T. Nugroho and B. Julianto, "QUALITY ANALYSIS OF SERVICE LOAD BALANCING USING PCC , ECMP AND NTH METHODS," *JANAPATI*, vol. 12, no. 2021, pp. 33-41, 2023.
- [15] E. Safriantim, "Peer Connection Classifier Method for Load Balancing Technique," *International Journal of Electrical, Energy and Power System Engineering*, vol. 4, no. 1, pp. 127-133, 2021.
- [16] M. Hasbi and N. R. Saputra, "Analisis Quality of Service ( Qos ) Jaringan Internet Kantor Pusat King Bukopin Dengan Menggunakan Wireshark," *Universitas Muhammadiyah Jakarta*, pp. 1-7, 2021.
- [17] J. I. Agbinya, "IP Communication and Services for NGN," CRC Press, 2010.