

# ANALISIS PERBANDINGAN MANAJEMEN BANDWIDTH MENGUNAKAN ROUTER MIKROTIK RB951Ui-2HnD DENGAN METODE QUEUE TREE DAN HIERARCHICAL TOKEN BUCKET

Ananta Ichsansyah<sup>1</sup>, Muhammad Syahroni<sup>2</sup>, Nasri<sup>3</sup>

<sup>1,2,3</sup>Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe

Email: anantaichsansyah2002@gmail.com<sup>1</sup>, msyahroni@pnl.ac.id<sup>2</sup>, nasrimt@gmail.com<sup>3</sup>

**Abstrak** – Permasalahan *bandwidth* yang tidak merata disebabkan oleh beberapa *user* yang sering melakukan kegiatan *browsing*, *streaming* film dan lagu bahkan kadang juga mengunduhnya. Akibatnya terjadi kemacetan jaringan, dan tidak stabilnya jaringan internet bagi pengguna. Manajemen *bandwidth* merupakan suatu cara yang dapat mengatur *bandwidth* dalam mengatasi pemerataan pemakaiannya. Sebuah jaringan memerlukan pengelolaan *bandwidth* yang baik, sehingga lalu lintas data dapat berjalan dengan tepat. Metode penelitian ini bertujuan untuk membandingkan antara *Queue Tree* dan *Hierarchical Token Bucket* (HTB) implementasi dari simple queue. Penelitian ini melakukan perancangan jaringan komputer dengan 4 *user* dengan cara melakukan pengujian menggunakan metode *Queue Tree* dan *Hierarchical Token Bucket* (HTB). Hasil analisa menggunakan *Quality of Service* (QoS) dengan beberapa parameter, yaitu *throughput*, *delay*, dan *packet loss*. Hasil analisa QoS didapatkan untuk pengujian *upload* maka *throughput* menggunakan metode *Queue Tree* saat melakukan *download* lebih mendekati limit yang diberikan dengan rata-rata 285,5 kbps, 561,5 kbps dan 938,75 kbps dibandingkan HTB, namun sebaliknya pada pengujian *throughput upload* menggunakan metode HTB lebih mendekati limit yang diberikan dengan rata-rata 301,25 kbps, 493 kbps dan 1038,5 kbps dari pada *Queue Tree*. *Packet loss* yang didapatkan menggunakan metode *Queue Tree* sebesar 0,73% pada pengujian *upload* dan 0,43% pada pengujian *download* sedangkan menggunakan HTB sebesar 0,5% pada pengujian *upload* dan 0,46% pada pengujian *download*. *Packet loss* yang didapatkan HTB merupakan *packet loss* terbaik dikarenakan memiliki *packet loss* terkecil. *Delay* yang didapatkan menggunakan metode *Queue Tree* sebesar 9 ms pada percobaan *upload* dan 10,16 ms pada percobaan *download*, sedangkan HTB sebesar 8,25 ms pada pengujian *upload* dan 9,5 ms pada pengujian *download*. *Delay* yang diperoleh HTB merupakan *delay* terbaik.

**Kata-kata kunci:** *Queue Tree, Hierarchical Token Bucket, Quality of Service, throughput, delay, packet loss*

**Abstract** – The problem of uneven bandwidth is caused by several users who often browse, stream movies and songs and sometimes even download them. As a result, there is network congestion and unstable internet network for users. Bandwidth management is a way to manage bandwidth in overcoming the distribution of its use. A network requires good bandwidth management, so that data traffic can run properly. This research method aims to compare *Queue Tree* and *Hierarchical Token Bucket* (HTB) implementations of simple queue. This study designs a computer network with 4 users by testing using the *Queue Tree* and *Hierarchical Token Bucket* (HTB) methods. The results of the analysis use *Quality of Service* (QoS) with several parameters, namely *throughput*, *delay*, and *packet loss*. The results of the QoS analysis were obtained for upload testing, the *throughput* using the *Queue Tree* method when downloading was closer to the given limit with an average of 285.5 kbps, 561.5 kbps and 938.75 kbps compared to HTB, but on the contrary, in the *upload throughput* test using the HTB method was closer to the given limit with an average of 301.25 kbps, 493 kbps and 1038.5 kbps than *Queue Tree*. The *packet loss* obtained using the *Queue Tree* method was 0.73% in the *upload* test and 0.43% in the *download* test, while using HTB was 0.5% in the *upload* test and 0.46% in the *download* test. The *packet loss* obtained by HTB was the best *packet loss* because it had the smallest *packet loss*. The *delay* obtained using the *Queue Tree* method was 9 ms in the *upload* test and 10.16 ms in the *download* test, while HTB was 8.25 ms in the *upload* test and 9.5 ms in the *download* test. The *delay* obtained by HTB is the best *delay*.

**Keywords:** *Queue Tree, Hierarchical Token Bucket, Quality of Service, throughput, delay, packet loss*

## I. PENDAHULUAN

Pada era globalisasi saat ini layanan komunikasi data mempunyai peranan yang sangat penting dalam kehidupan. Hampir di setiap bidang kehidupan telah menggunakan layanan ini. Layanan ini tidak hanya di gunakan secara individual tetapi juga secara umum.

Hampir semua lembaga atau organisasi menggunakan akses internetnya secara umum, misalnya: lembaga pendidikan, perkantoran, warnet dan masih banyak lagi lembaga-lembaga lainnya. Penggunaan akses internet secara massa ini akan mengakibatkan turunnya performa jaringan seiring dengan peningkatan jumlah

pengguna. Apalagi jika *bandwidth* yang tidak dikelola sebaik mungkin.

Permasalahan *bandwidth* yang tidak merata disebabkan oleh beberapa *user* yang sering melakukan kegiatan *browsing*, *streaming* film dan lagu bahkan kadang juga mengunduhnya. Akibat dari kegiatan tersebut dapat menghabiskan *source bandwidth* yang tidak sedikit, sehingga mengganggu aktifitas-aktifitas yang lain. Akibatnya terjadi kemacetan jaringan, dan tidak stabilnya jaringan internet bagi pengguna.

Manajemen *bandwidth* merupakan suatu cara yang dapat mengatur *bandwidth* dalam mengatasi pemerataan pemakaiannya. Sebuah jaringan memerlukan pengolaan *bandwidth* yang baik, sehingga lalu lintas data dapat berjalan dengan tepat. Manajemen *bandwidth* dapat membantu dalam proses lalu lintas data internet menjadi lebih efisien penggunaannya, dengan mengoptimalkan penggunaan *bandwidth* yang sangat terbatas.

## II. TINJAUAN PUSTAKA

### A. *Bandwidth*

*Bandwidth* adalah banyaknya ukuran suatu data atau informasi yang dapat mengalir dari suatu tempat ke tempat lain dalam sebuah *networks* dalam waktu tertentu. *Bandwidth* dapat dipakai untuk mengukur baik aliran data analog maupun data digital. Sekarang sudah menjadi umum jika kata *bandwidth* lebih banyak untuk mengukur aliran data digital [1]. Satuan yang dipakai untuk *bandwidth* adalah bits per second atau sering disingkat bps. Seperti diketahui bahwa bit atau biner digit adalah basis angka yang terdiri dari 0 dan 1. Satuan ini menggambarkan beberapa bit (angka 0 dan 1) yang dapat mengalir dari suatu tempat ke tempat yang lain dalam setiap detiknya melalui suatu media

### B. Manajemen *Bandwidth*

Manajemen *bandwidth* adalah suatu mekanisme untuk dapat mengatur rate dan prioritas *traffic data* yang dikirimkan melalui suatu *Networks interfaces* ke jaringan. Istilah manajemen *bandwidth* ini sering disebut dengan *traffic control*. Pengaturan ini sangat penting dalam menjamin kualitas layanan jaringan, bisa di bayangkan bila dalam jaringan ada sebuah komputer yang men-*download* data dari internet dengan ukuran *file* yang di-*download* cukup besar maka *bandwidth* pengguna akses internet bisa jadi tidak tersedia [2].

Untuk itu diperlukan suatu sistem yang mampu mengatur secara adil agar konsumsi *bandwidth* dalam jaringan tidak dimonopoli oleh satu atau beberapa komputer saja. Solusi untuk kejadian seperti itu adalah dengan menggunakan sebuah device khusus yang dapat mengatur trafik atau *bandwidth* yang terpasang diantara modem dan jaringan LAN. Biasanya solusi tersebut memerlukan *coast* yang cukup besar.

### C. *Queue Tree*

*Queue Tree* adalah pelimitan yang sangat rumit karena pelimitan ini berdasarkan *protokol*, *ports*, *IP Address*, bahkan kita harus mengaktifkan fitur *Mangle* pada

*Firewall* jika ingin menggunakan *Queue Tree*. *Queue Tree* berfungsi untuk melimit *bandwidth* pada MikroTik yang mempunyai dua koneksi internet karena paket marknya lebih berfungsi dari pada di *Simple Queues*. *Queues tree* juga digunakan untuk membatasi satu arah koneksi saja baik itu *download* maupun *upload* [3].

*Queue Tree* suatu bentuk konfigurasi manajemen *bandwidth* yang cukup kompleks karena dapat disesuaikan berdasarkan *protokol*, nomor *port* atau pengelompokan berdasarkan alamat *IP*.sebelum melakukan limit *bandwidth*, anda harus menandai aliran paket menggunakan suatu tanda *mangle* agar paket tersebut dapat dikenal oleh *Queue Tree* [4].

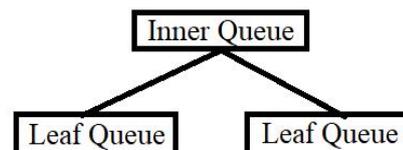
### D. *Hierarchical Token Bucket (HTB)*

*HTB (Hierarchical Token Bucket)* merupakan algoritma standar yang digunakan untuk mengendalikan QoS (semua tipe *queues* dan kontrol *bandwidth*) di dalam routerOS. HTB mendukung teknik *bursting* dan mengendalikan aliran data keluar (*outbound data flow*). Kita tidak dapat membatasi *traffic* yang masuk ke suatu *interface (inbound data flow)* [5]. Tetapi kita dapat mengontrol *traffic* dengan menggunakan teknik *buffering* (menahan sementara), atau manakala *limit buffer* terlampaui (penuh), akan dilakukan *drop* pada paket tersebut. Pada aplikasi yang berkerja menggunakan protokol TCP, paket yang di drop akan dikirimkan ulang sehingga tidak ada kehilangan paket data.

Limit bertahap (Staged Limitation)

1. CIR (*Committed Information Rate*) adalah melakukan limitasi *bandwidth* sebesar *limit-at* kepada *client* selama *bandwidth* tersedia meskipun dalam keadaan terburuk.
2. MIR (*Maximal Information Rate*) adalah melakukan limitasi *bandwidth* sebesar *max-limit* selama *bandwidth* tersedia. Dimana Router akan melakukan alokasi *bandwidth* sebesar limit-at terlebih dahulu jika ada *bandwidth* tak terpakai maka router akan mengkalkulasi *client* mana yang berhak mendapatkan *bandwidth* tambahan [6].

Ada dua kategori kelas dalam HTB, yaitu *Inner Queue* dan *Leaf Queue*. *Inner Queue* merupakan parent tertinggi dimana trafik layer yang di berikan melalui klasifikasi yang harus digunakan melalui *filter*, sehingga memungkinkan untuk membedakan jenis trafik dan prioritas sedangkan *Leaf Queue* merupakan *Queue* yang malakukan limitasi langsung pada *user*.Berikut gambar struktur dari kelas HTB.



Gbr. 1 Struktur Kelas HTB

E. Mikrotik Routerboard

Mikrotik *routerboard* merupakan sebuah perangkat jaringan komputer yang menggunakan Mikrotik RouterOS yang berbasis Linux dan diperuntukkan bagi *network router*. Mikrotik *routerboard* memiliki beberapa fasilitas seperti *bandwidth management*, *stateful firewall*, *hotspot for plug and play access remote* Winbox GUI *admin*, dan *routing*. Administrasi MicroticOs bisa dilakukan melalui Windows application (WinBox) [2].

Saat ini WinBox sudah ditampilkan secara grafis, sehingga pengguna dapat dengan mudah, efektif dan efisien mengakses dan mengkonfigurasi router sesuai kebutuhannya.

F. Wireshark

Wireshark merupakan salah satu dari sekian banyak *tool Network Analyzer* yang banyak digunakan oleh *Network Administrator* untuk menganalisa kinerja jaringannya termasuk protokol didalamnya. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa. Wireshark dipakai oleh *network administrator* untuk menganalisa kinerja jaringannya. Wireshark merupakan *software* untuk melakukan analisa lalu-lintas jaringan komputer, yang memiliki fungsi-fungsi yang amat berguna bagi profesional jaringan, *administrator* jaringan, peneliti, hingga pengembang piranti lunak jaringan [7].

Wireshark sendiri juga memiliki fitur yang cukup lengkap, diantaranya yaitu :

1. *Multiplatform*, bisa dipakai untuk beberapa basis sistem operasi (Unix, Mac, Windows, serta Linux).
2. Bisa lakukan *capture* paket data jaringan secara real time.
3. Bisa menampilkan informasi protokol jaringan dari paket data secara komplit.
4. Paket data bisa disimpan jadi file serta nantinya bisa di buka kembali untuk analisa lebih lanjut.
5. *Filtering* paket data jaringan.
6. Untuk lakukan *capture* paket data yang keluar maupun masuk pada jaringan, wireshark membutuhkan piranti fisik NIC (Network Interface Card).



Gbr. 2 Ikon Aplikasi Wireshark

G. Winbox

Winbox adalah utility yang digunakan untuk konektivitas dan konfigurasi MikroTik menggunakan *MAC Address* atau *protokol IP* [5]. Konfigurasi

MikroTik RouterOS dan RouterBoard dapat dilakukan dengan menggunakan mode GUI dengan cepat dan sederhana. Winbox dibuat menggunakan win32 binary tapi dapat dijalankan pada Linux, Mac OS.



Gbr. 3 Ikon Aplikasi Winbox

H. Quality of Service (QoS)

*Quality of Service (QoS)* merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat suatu layanan. QoS digunakan untuk mengukur sekumpulan atribut kinerja yang telah di spesifikasikan dan diasosiasikan dengan suatu layanan. Berikut ini merupakan beberapa parameter QoS yang sering digunakan dalam mengukur performansi jaringan [6].

1) *Throughput*

*Throughput* merupakan jumlah total kedatangan paket yang sukses yang diamati pada destination selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut. *Throughput* adalah kemampuan sebenarnya suatu jaringan dalam melakukan pengiriman data. Biasanya *throughput* selalu dikaitkan dengan *bandwidth*. Adapun nilai kategori *throughput* dengan standarisasi versi TIPHON [6].

TABEL I  
Standarisasi Troughput TIPHON

Kategori	Throughput	Indeks
Bad	0-338 kbps	0
Poor	338-700 kbps	1
Fair	700-1200 kbps	2
Good	1200 kbps-2,1 Mbps	3
Excelent	> 2,1 Mbps	4

Berikut Persamaan 1 untuk menghitung nilai *throughput* dirumuskan sebagai berikut [6].

$$Throughput = \frac{\text{Jumlah data diterima(byte)}}{\text{Waktu pengamatan(s)}} \tag{1}$$

2) *Packet loss*

*Packet loss* adalah banyaknya paket yang gagal mencapai tempat tujuan paket tersebut dikirim. Pada penelitian ini juga digunakan parameter *packet loss* untuk mengukur baik buruknya jaringan. Tabel 2 adalah nilai kategori *packet loss* dengan standarisasi versi TIPHON [6].

TABEL II  
Standarisasi *packet loss* TIPHON

Kategori	Throughput	Indeks
Sangat Bagus	0	4
Bagus	3	3
Sedang	15	2
Buruk	25	1

Perhitungan jumlah paket yang hilang selama proses Transmisi data dapat digunakan rumus Persamaan 2 berikut [6]:

$$Packet\ Loss(\%) = \frac{paket\ terkirim(byte) - paket\ diterima(byte)}{paket\ terkirim(byte)} \times 100\% \quad (2)$$

3) *Delay*

*Delay* adalah waktu yang dibutuhkan oleh satu paket dari asal ke sumber tujuan. Satuan dari parameter *delay* dinyatakan dalam ms (*millisecond*). Pada penelitian ini digunakan standar pembandingan yaitu yang dikeluarkan oleh ITU-T. Adapun nilai kategori *delay* dengan standarisasi TIPHON [6].

TABEL III  
Standarisasi *delay* TIPHON

Kategori	Throughput	Indeks
Sangat Bagus	<150	4
Bagus	150-300	3
Sedang	300-450	2
Buruk	>450	1

Nilai *delay* pada penelitian ini dihitung dengan menggunakan Persamaan 3 berikut [6]:

$$Delay\ rata - rata(ms) = \frac{Total\ delay\ (ms)}{Total\ paket\ yang\ diterima(byte)} \quad (3)$$

III. METODOLOGI

A. Alat Dan Bahan

Dalam penelitian ini penulis membutuhkan beberapa perangkat lunak (software) dan perangkat keras (hardware) sebagai penunjang, diantaranya:

Hardware

- a. komputer windows 7 4 Unit
- b. Router Boardband RB951Ui-2HND 1 Unit
- c. kabel LAN (Straight) 4 Buah

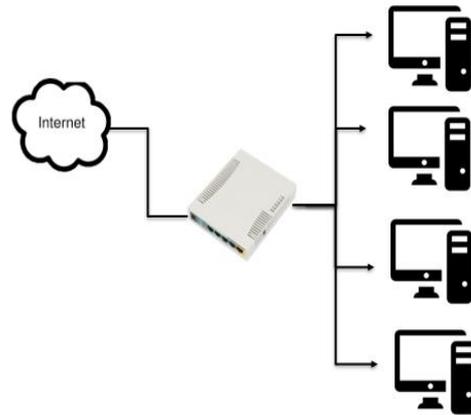
Perangkat Lunak (Software)

- a. Winbox berfungsi sebagai konfigurasi/remot ke server perangkat keras mikrotik dalam mode GUI
- b. Wireshark berfungsi sebagai tempat untuk menganalisa kinerja jaringan

B. Rancangan Percobaan

Pada Rancangan percobaan penulis ingin menguraikan cara pengambilan data pada penelitian ini sebagai

berikut. Pada penelitian ini dilakukan perancangan jaringan menggunakan metode *Queue Tree* dan Metode *Hierarchical Token Bucket*. Data yang diukur adalah *throughput*, *packet loss*, dan *delay*. Data diukur dengan cara melakukan *upload* dan *download* data.



Gbr. 4 Perancangan Jaringan *Queue Tree* Dan HTB

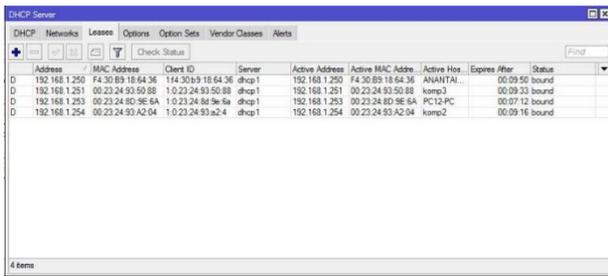
C. Metode Analisis

Pada tahapan ini pengolahan data dilakukan secara perhitungan dan pengukuran. Dimana pengukuran menggunakan *software* Wireshark yang telah di jalan kan pada setiap komputer *client*. Nantinya dimana berfungsi untuk memonitoring dan capture hasil dari QoS (*QualityOfService*) ditentukan oleh beberapa parameter, seperti *throughput*, *packet loss* dan *delay*. Dari jaringan yang telah dilakukan sebelum dan sesudah menggunakan metode *Queue Tree* dan *Hierarchical Token Bucket* (HTB), sedangkan perhitungan validasi pada *QualityOfService* (QoS) megggunakan rumus persamaan. Dimana parameternya *throughput* menggunakan persamaan, *packet loss* menggunakan persamaan, dan *delay* menggunakan persamaan. Berdasarkan hasil dari konfigurasi, pengukuran dan perhitungan dari data yang diperoleh dalam penelitian ini akan di muat dalam bentuk tabel dan grafik.

IV. HASIL DAN PEMBAHASAN

Berdasarkan hasil penelitian yang telah dilakukan pada jaringan Local Area Network (LAN) pada Laboratorium Jaringan Komputer, Teknik Elektro, Politeknik Negeri Lhokseumawe. Dimana dilakukan perancangan jaringan komputer dengan 4 *user* yang dimana masing masing komputer memiliki akses internet. Telah dilakukan manajemen *bandwidth* menggunakan metode *Queue Tree* dan *Hierarchical Token Bucket* (HTB). Dapat dilihat pada Gambar 5 hasil dari jaringan 4 *user* yang menggunakan jaringan LAN yang terhubung ke Mikrotik.

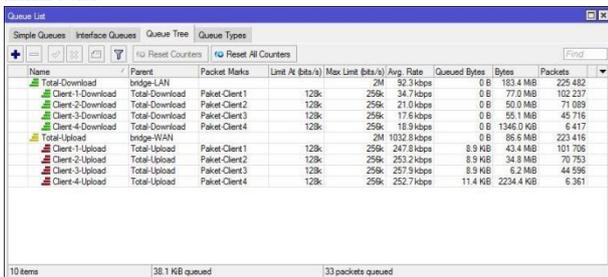
Berdasarkan Gambar 5 maka dilakukan analisis hasil dari pengukuran dan perhitungan. Pengujian saat melakukan *upload* dan *download* ditentukan oleh beberapa parameter antara lain *throughput*, *packet loss*, dan *delay*.



Gbr. 5 Konfigurasi Mikrotik

A. Hasil Konfigurasi pada mikrotik menggunakan metode *Queue Tree*

Hasil konfigurasi pada mikrotik menggunakan metode *Queue Tree* di lakukan pada 4 komputer pada percobaan *upload* dan *download*. Hasil dari capture gambar mikrotik.



Gbr. 6 Hasil Konfigurasi *Queue Tree*

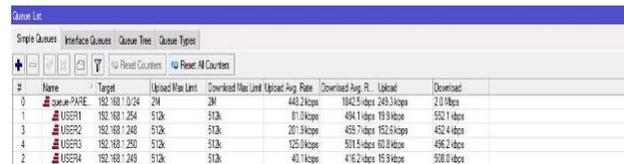
Berdasarkan Gambar 6 diketahui Parent-*download* dan *upload* mendapatkan Max-limit *bandwidth* sebesar 2 Mbps. Setiap *user* diberi Limit-at sebesar 128 kbps dan Max-limit 256 kbps yang diberikan kepada 4 *user* untuk melakukan *download* dan *upload* file sebesar 10MB.

Dari hasil konfigurasi di atas dapat dilihat saat melakukan pengujian bahwa *user* 1 mendapatkan *bandwidth* sebesar 247.8 kbps, *user* 2 mendapatkan *bandwidth* sebesar 253.2 kbps, *user* 3 mendapatkan *bandwidth* sebesar 257.9 kbps dan *user* 4 mendapatkan 252.7 kbps. dikarenakan *Queue Tree* melakukan limitasi *bandwidth* tanpa memperhentikan antrian sehingga proses pembagian *bandwidth* akan dijalankan secara bersamaan.

Sehingga *user* akan mendapatkan *bandwidth* adil dan rata walau dengan *bandwidth* yang rendah. Dikarenakan dalam trafik seburuk apapun *user* tetap akan mendapatkan *bandwidth* sehingga tidak adanya penguasa dan *user* dapat melakukan akses internet dengan nyaman.

B. Hasil Konfigurasi pada mikrotik menggunakan metode HTB

Hasil konfigurasi pada mikrotik menggunakan metode HTB dilakukan pada 4 komputer pada percobaan *upload* dan *download*. Hasil dari capture gambar mikrotik.



Gbr. 7 Hasil Konfigurasi HTB

Berdasarkan Gambar 7 diketahui Parent-*download* dan *upload* mendapatkan Max-limit *bandwidth* sebesar 2 Mbps. Setiap *user* diberi limit sebesar 512 kbps yang diberikan kepada 4 *user* untuk melakukan *download* dan *upload* file sebesar 10MB.

Dari konfigurasi penerapan HTB dapat dilihat saat pengujian pengiriman data bahwa *user* 1 mendapatkan *bandwidth* sebesar 552.1 kbps yang dimana *user* 1 menjadi prioritas pertama dan *user* 3 mendapatkan 496.2 kbps yang dimana *user* 3 menjadi prioritas terakhir. Dikarenakan HTB melakukan melakukan limitasi *bandwidth* dengan memperhentikan antrian sehingga proses pembagian *bandwidth* akan dijalankan secara hirarki.

Sehingga *user* akan mendapatkan *bandwidth* adil dan rata walau dengan *bandwidth* yang rendah. Dikarenakan dalam trafik seburuk apapun *user* tetap akan mendapatkan *bandwidth* sehingga tidak adanya penguasa dan *user* dapat melakukan akses internet dengan nyaman. Jika ada *bandwidth* yang tidak terpakai dapat di gunakan oleh *user* yang diprioritaskan

C. Pembahasan Penggunaan Metode *Queue Tree* dan *Hierarchical Token Bucket* (HTB)

Setelah melakukan percobaan *download* dan *upload* menggunakan metode *Queue Tree* dan HTB maka didapatkan Rata-rata *throughput Queue Tree* sebesar 314,5 kbps, 567 kbps dan 839 kbps untuk percobaan *upload* dan 285,5 kbps, 561,5 kbps dan 938,75 kbps pada percobaan *download* sedangkan HTB memperoleh *throughput* sebesar 301,5 kbps, 493 kbps dan 1038,5 kbps pada pengujian *upload* dan 314,75 kbps, 553,25 kbps dan 1039,5 kbps pada percobaan *download* pada Limitasi 256 kbps, 512 kbps dan 1 Mbps. Pada pengujian *upload* adan *download* yang di limitasi bahwa *throughput* yang didapatkan *Queue Tree* pada percobaan *download* lebih mendekati dengan limitasi yang diberikan namun pada percobaan *upload throughput* yang dihasilkan HTB lebih mendekati limitasi yang di atur.

TABEL IV  
Rata-Rata *throughput*

Bandwidth (kbps)	Queue Tree		HTB	
	Upload (kbps)	Download (kbps)	Upload (kbps)	Download (kbps)
256	314,5	285,5	301,25	314,75
512	567	561,5	493	553,25
1000	839	938,75	1038,5	1039,5

Setelah melakukan percobaan *download* dan *upload* menggunakan metode *Queue Tree* dan HTB maka didapatkan rata-rata *packet loss* sebesar 0,73 % untuk pengujian *upload* dan 0,43 % untuk pengujian *download* menggunakan metode *Queue Tree* dan rata-rata *packet loss* sebesar 0,5 % untuk pengujian *upload* dan 0,46 % untuk pengujian *download* menggunakan metode *Hierarchical Token Bucket*. Dari rata-rata *packet loss* yang didapatkan metode HTB merupakan metode yang lebih baik dari pada *Queue Tree* dikarenakan memiliki *packet loss* yang lebih kecil.

TABEL V  
Rata-Rata *packet loss*

Bandwidth (kbps)	Queue Tree		HTB	
	Upload (%)	Download (%)	Upload (%)	Download (%)
256	0,5	0,5	0,6	0,7
512	1,2	0,7	0,6	0,4
1000	0,2	0,1	0,3	0,3
Rata-rata	0,73	0,43	0,5	0,46

Setelah melakukan percobaan *download* dan *upload* menggunakan metode *Queue Tree* dan HTB maka didapatkan rata-rata *delay* sebesar 9 ms untuk pengujian *upload* dan 10,16 ms untuk pengujian *download* menggunakan metode *Queue Tree* dan Rata-rata *delay* sebesar 8,25 ms untuk pengujian *upload* dan 9,5 ms untuk pengujian *download* menggunakan metode HTB. Dari rata-rata *delay* yang diperoleh metode HTB merupakan metode yang lebih baik di bandingkan *Queue Tree* dikarenakan memiliki *delay* yang lebih kecil.

TABEL VI  
Rata-Rata *delay*

Bandwidth (kbps)	Queue Tree		HTB	
	Upload (ms)	Download (ms)	Upload (ms)	Download (ms)
256	14,5	16,5	14,75	14,75
512	5,5	8,75	5	9
1000	7	5,25	5	5
Rata-rata	9	10,16	8,25	9,5

## V. KESIMPULAN

Dari hasil dan pembahasan dapat diambil beberapa kesimpulan mengenai perbandingan manajemen *bandwidth* menggunakan metode *Queue Tree* dan *Hierarchical Token Bucket* (HTB), yaitu:

1. Hasil pengujian *throughput* dengan besar limitasi 256 kbps, 512 kbps dan 1 Mbps dengan melakukan pengujian *upload* dan *download*, dengan file 10 MB, dimana rata-rata dari *throughput upload* yang dihasilkan dengan metode HTB 301,25 kbps, 493 kbps dan 1038,5 kbps yang mana *Hierarchical Token Bucket* lebih baik dibandingkan *Queue Tree*.

Namun pada pengujian *download Queue Tree* lebih baik dibandingkan *Hierarchical Token Bucket* dengan memperoleh rata-rata *throughput* 285,5 kbps, 561,5 kbps dan 938,75 kbps.

2. Hasil pengujian *packet loss* dengan besar limitasi 256 kbps, 512 kbps dan 1 Mbps dengan melakukan pengujian *upload* dan *download* dengan File 10 MB, maka diperoleh rata-rata *packet loss* dengan metode *Queue Tree* adalah 0,73 % pada percobaan *upload* dan 0,43 % pada percobaan *download* yang dimana termasuk kategori Sangat Bagus, sedangkan menggunakan HTB diperoleh rata-rata nilai *packet loss* sebesar 0,5 % pada percobaan *upload* dan 0,46 % pada percobaan *download* yang mana termasuk kategori sangat bagus pada standar TIPHON. Dari hasil rata-rata diperoleh bahwa metode *Queue Tree* memiliki *packet loss* yang lebih kecil pada percobaan *download* sedangkan HTB memiliki *packet loss* yang lebih kecil pada percobaan *upload*.
3. Hasil Pengujian *delay* dengan besar limitasi 256 kbps, 512 kbps dan 1 Mbps dengan melakukan pengujian *upload* dan *download* dengan file 10 MB, maka diperoleh rata-rata *delay* yang di hasilkan menggunakan *Queue Tree* adalah 9 ms pada pengujian *upload* dan 10,16 ms yang termasuk kategori sangat bagus, sedangkan menggunakan HTB diperoleh nilai *delay* 8,25 ms pada pengujian *upload* dan 9,5 ms pada pengujian *download* dimana termasuk kategori sangat bagus pada standar TIPHON. Berdasarkan rata-rata pengujian *delay* maka diperoleh bahwa HTB lebih baik dikarenakan memiliki *delay* yang lebih kecil.

## REFERENSI

- [1]. Syafrizal, M. (2020). *Pengantar jaringan komputer*. Penerbit Andi.
- [2]. Iwan, S. (2017). Jaringan Komputer berbasis MikroTik. *Informatika, Bandung*
- [3]. Susianto, D. (2016). Implementasi *Queue Tree* untuk manajemen *bandwidth* menggunakan router board mikrotik. *Jurnal Cendikia, 14*(1 April), 1-7
- [4]. Nadhif, M. F., Indriati, R., & Sucipto, S. (2019). Arsitektur Manajemen *bandwidth* Menggunakan Metode *Queue Tree*. In *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)* (Vol. 3, No. 1, pp. 145-150).
- [5]. Wijaya, A. I., Handoko, L. B., & Kom, M. (2015). Manajemen *bandwidth* Dengan Metode Htb (*Hierarchical Token Bucket*) Pada Sekolah Menengah Pertama Negeri 5 Semarang. *J. Tek. Inform. Udinus, 1*(1), 5-7.
- [6]. Fadilah, A. Z., Saedudin, R. R., & Hedyanto, U. Y. K. S. (2021). Analisis Simulasi Manajemen *bandwidth* Menggunakan Metode *Hierarchical Token Bucket* (htb) Untuk Meningkatkan Quality

- 
- Of Service (qos). *eProceedings of Engineering*, 8(5).
- [7]. Faisal, I. (2018). An Analisis Qos Pada Implementasi Manajemen Bandwith Menggunakan Metode *Queue Tree* Dan Pcq (Per Connection Queueing). *Jurnal Teknologi dan Ilmu Komputer Prima (JUTIKOMP)*, 1(1), 137-142.