

Analisis Perbandingan Manajemen *Bandwidth* Menggunakan Metode *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket* pada Hotspot Mikrotik

Luthfia Riska¹, Husaini^{2*}, Fachri Yanuar Rudi F³

^{1,2,3} Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA

¹luthfiariska8@gmail.com

^{2*}husaini@pnl.ac.id

³fachri@pnl.ac.id

Abstrak— Manajemen *bandwidth* sangatlah penting untuk mengendalikan pemakaian *bandwidth* yang akan digunakan oleh setiap *client*. Manajemen *bandwidth* pada jaringan internet diperlukan untuk mengatur tiap data yang lewat sehingga pembagian *bandwidth* menjadi adil dan merata. Pada saat ini di SMK Negeri 1 Lhokseumawe sudah menyediakan *bandwidth*, namun tidak ada manajemen pemakaian *bandwidth* sehingga koneksi internet menjadi lambat. Untuk memudahkan dalam pengelolaan manajemen *bandwidth* maka penulis akan melakukan perbandingan metode *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket* untuk mengetahui mana yang paling efektif. Pengujian akan dilakukan dengan melakukan *upload* dan *download file* pada *google drive* dan diukur kualitas jaringan menggunakan parameter QoS kemudian dilakukan analisis perbandingan dari ketiga metode tersebut. Berdasarkan analisis data hasil pada ruang lab ditinjau dari nilai rata-rata *upload* pada *throughput* dan *delay*, metode HTB lebih unggul dari pada metode *Queue Tree* dan *Simple Queue* dengan nilai 5953 kbps dan 1,47 ms dan nilai *packet loss* dan *jitter* lebih unggul metode *Queue Tree* daripada *Simple Queue* dan HTB dengan nilai 0,0% dan 0,0 ms. Sedangkan hasil data dari nilai rata-rata *download* pada *throughput* dan *delay*, metode HTB lebih unggul dari pada metode *Queue Tree* dan *Simple Queue* dengan nilai 4400 kbps dan 1,51 ms. Nilai *packet loss* lebih unggul metode *Queue Tree* dan HTB yaitu 0,0%. Dan nilai *jitter* lebih unggul pada metode *Queue Tree* sebesar 0,01 ms. Pada ruang kelas ditinjau dari nilai rata-rata *upload* *throughput* lebih unggul metode HTB sebesar 3829 kbps, nilai *packet loss* lebih unggul metode *Queue Tree* dan *Simple Queue* sebesar 0,0%, nilai *delay* lebih unggul metode *Queue Tree* sebesar 2,30 ms dan nilai *jitter* unggul di ketiga metode. Sedangkan hasil data dari nilai rata-rata *download* pada *throughput*, *delay* dan *jitter* lebih unggul metode HTB sebesar 2474 kbps, 2,60 ms dan 0,01 ms. Nilai *packet loss* lebih unggul metode *Queue Tree* sebesar 0,0%. Setelah dilakukan perbandingan dan analisa dari ketiga metode maka metode *Hierarchical Token Bucket* jauh lebih stabil pada saat melakukan manajemen *bandwidth* dan dapat mengontrol penggunaan internet dengan baik. Dengan demikian metode *Hierarchical Token Bucket* akan lebih tepat jika diterapkan pada SMK Negeri 1 Lhokseumawe.

Kata kunci— Manajemen *Bandwidth*, *Simple Queue*, *Queue Tree*, *Hierarchical Token Bucket* (HTB) dan QoS.

Abstract— *Bandwidth management* is very important to control *bandwidth usage* that will be used by each *client*. *Bandwidth management* on the internet network is needed to manage each data that passes so that the distribution of *bandwidth* becomes fair and equitable. Currently, SMKN 1 Lhokseumawe has provided *bandwidth*, but there is no management of *bandwidth usage* so the internet connection is slow. To make it easier to manage *bandwidth management*, the author will compare the *Simple Queue*, *Queue Tree* and *Hierarchical Token Bucket* methods to find out which is the most effective. Testing will be carried out by uploading and downloading files on Google Drive and measuring network quality using QoS parameters and then conducting a comparative analysis of the three methods. Based on analysis of the results data in the lab room in terms of average *upload* *throughput* and *delay* values, the HTB method is superior to the *Queue Tree* and *Simple Queue* methods with values of 5953 kbps and 1.47 ms and the *packet loss* and *jitter* values are superior to the *Queue method Tree* from *Simple Queue* and HTB with a value of 0.0% and 0.0 ms. Meanwhile, the data results from the average *download value* on *throughput* and *delay*, the HTB method is superior to the *Queue Tree* and *Simple Queue* methods with values of 4400 kbps and 1.51 ms. The *packet loss value* is superior to the *Queue Tree* and HTB methods, namely 0.0%. And the *jitter value* is superior to the *Queue Tree* method of 0.01 ms. In the classroom, the average *upload* *throughput* value is superior to the HTB method by 3829 kbps, the *packet loss value* is superior to the *Queue Tree* and *Simple Queue* methods by 0.0%, the *delay value* is superior to the *Queue Tree* method by 2.30 ms and the value superior *jitter* in all three methods. Meanwhile, the data results from the average *download value* in terms of *throughput*, *delay* and *jitter* are superior to the HTB method at 2474 kbps, 2.60 ms and 0.01 ms. The *packet loss value* is superior to the *Queue Tree* method by 0.0%. After comparing and analyzing the three methods, the *Hierarchical Token Bucket* method is much more stable when managing *bandwidth* and can control internet usage well. Thus, the *Hierarchical Token Bucket* method would be more appropriate if applied to SMKN 1 Lhokseumawe.

Keywords— *Bandwidth Management*, *Simple Queue*, *Queue Tree*, *Hierarchical Token Bucket* (HTB) and QoS.

I. PENDAHULUAN

Perkembangan teknologi informasi saat ini menjadikan kebutuhan akses internet semakin meningkat untuk

melakukan berbagai macam kegiatan misalnya untuk berkomunikasi, *browsing*, mengakses situs web, mengunggah data, mengunduh data dan lain sebagainya sesuai kebutuhan si pengguna. Setiap pengguna pasti menginginkan akses

internet yang cepat dan stabil. Karena banyaknya pengguna mengakses internet sering kali muncul masalah *bandwidth*. Pemakaian yang berlebihan akan menyebabkan *user* yang lain mendapatkan alokasi *bandwidth* yang kecil dan akan mengalami *loading* yang lama saat mengakses internet. Manajemen *bandwidth* sangatlah penting untuk mengendalikan pemakaian *bandwidth* yang akan digunakan oleh *user*. Jika tidak akan terjadinya pemakaian *bandwidth* secara berlebihan oleh beberapa *user*. Manajemen *bandwidth* pada jaringan internet diperlukan untuk mengatur tiap data yang lewat sehingga pembagian *bandwidth* menjadi adil dan merata. Oleh karena itu, akan digunakan QoS untuk mengukur kualitas koneksi jaringan dengan menggunakan beberapa parameter sehingga dapat menghindari perebutan alokasi *bandwidth* terhadap *user* yang ada di dalam jaringan.

SMK Negeri 1 Lhokseumawe memiliki jaringan akses internet yang digunakan oleh guru dan siswa. Jaringan internet pada SMK Negeri 1 Lhokseumawe tidak stabil karena belum dilakukan manajemen *bandwidth*. Jadi setiap pengguna yang mengakses internet tidak dibatasi *bandwidth* sehingga pengguna yang lain saat mengakses internet menjadi lambat. Dengan demikian, pihak sekolah memerlukan adanya manajemen *bandwidth* agar jaringan internet menjadi stabil dan setiap guru dan siswa mendapatkan *bandwidth* yang adil. Agar memudahkan pengelolaan manajemen *bandwidth*, maka penulis akan melakukan analisis perbandingan metode manajemen *bandwidth* yang nantinya akan diterapkan pada SMK Negeri 1 Lhokseumawe. Metode yang akan digunakan adalah *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket*. Dengan digunakannya tiga metode penelitian ini diharapkan mendapatkan hasil yang lebih baik untuk diterapkannya manajemen *bandwidth* pada sekolah SMK Negeri 1 Lhokseumawe.

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Analisis Perbandingan Quality of Service (QoS) pada Performa *Bandwidth* dengan Metode *Hierarchical Token Bucket* (HTB) dan *Per Connection Queue* (PCQ)”. Penelitian ini melakukan analisis variabel *Quality of Service* terhadap performa *bandwidth* dengan membandingkan dua metode antrian yaitu metode *Hierarchical Token Bucket* dan metode *Per Connection Queue*. Penelitian dilakukan untuk mengetahui metode manakah lebih baik untuk di implementasikan oleh administrator jaringan agar efisien dan tepat untuk digunakan[1].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Implementasi Metode *Queue Tree* untuk Optimalisasi Jaringan pada Router Mikrotik”. Penelitian ini dilakukan untuk memperoleh hasil dalam menunjukkan *client* yang diberikan prioritas tertinggi mendapatkan alokasi kecepatan *throughput* tertinggi[2].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Perbandingan Metode *Simple Queue* dan *Queue Tree* untuk Optimalisasi Manajemen *Bandwidth* Menggunakan Mikrotik (Studi di Asrama Wisma Muslim)”. Penelitian ini melakukan pengujian pada 5 *client Simple Queue* dan *Queue Tree*. Pada nilai QoS *Simple Queue* lebih bagus dibandingkan dengan *Queue Tree*, sedangkan pada pengujian 10 *client Simple Queue* dan *Queue Tree*. Pada nilai QoS *Queue Tree* lebih bagus dibandingkan dengan *Simple Queue*. Berdasarkan hasil maka dapat disimpulkan bahwa metode *Simple Queue* lebih sesuai untuk jumlah *client* kurang dari 5, sedangkan untuk jumlah *client* lebih dari 5, direkomendasikan menggunakan *Queue Tree*[3].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Analisis Perbandingan Metode *Queue* pada Mikrotik”. Penelitian ini dilakukan untuk melakukan manajemen trafik dengan mengatur *bandwidth upload* dan *download* agar teratur pada *Simple Queue* dan *Queue Tree*[4].

Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Penerapan Manajemen *Bandwidth* Menggunakan Metode *Hierarchical Token Bucket* pada Layanan Hotspot Mikrotik Undiksha”. Penelitian ini dilakukan untuk manajemen *bandwidth* dengan menggabungkan layanan hotspot mikrotik dan hasil rata-rata *download* dan *upload* dari metode HTB lebih besar dibandingkan *Simple Queue* dan dapat dikatakan metode HTB melakukan manajemen *bandwidth* yang disebar ke *client* dengan baik[5].

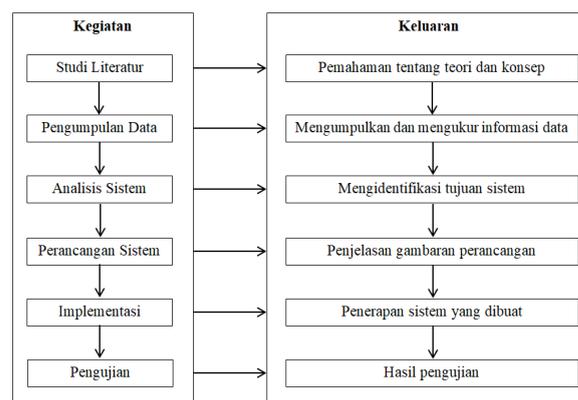
Penelitian ini berkaitan dengan penelitian sebelumnya dengan judul “Analisa QoS dengan *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket* (Studi Kasus Pro Net Bangkingan)”. Hasil penelitian yang telah dilakukan yaitu manajemen *bandwidth* lebih optimal menggunakan *Queue Tree* pada Mikrotik dibandingkan dengan *Simple Queue* pada Mikrotik dan HTB pada Linux Ubuntu, karena hasil QoS pada *Queue Tree* yang didapat lebih besar, yaitu dengan hasil akhir nilai rata-ratanya 13, sedangkan *Simple Queue* pada Mikrotik dan HTB pada Linux Ubuntu hasil akhir rata-ratanya dengan nilai 12[6].

II. METODOLOGI PENELITIAN

Metode yang digunakan dalam penelitian ini adalah *simple queue*, *queue tree* dan *hierarchical token bucket* dengan melakukan limitasi pada *bandwidth*.

A. Tahapan Penelitian

Langkah-langkah yang dilakukan peneliti dalam menerapkan sistem yang dirancang dapat dilihat pada Gambar 1 berikut.



Gambar 1. Tahapan Penelitian

B. Pengumpulan Data

Proses pengumpulan data dalam penelitian ini dilakukan untuk memajemenkan *bandwidth* pada hotspot mikrotik menggunakan metode *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket*. Pengumpulan data dilakukan melalui objek penelitian pada SMK Negeri 1 Lhokseumawe. Adapun tahapan pengumpulan data yang akan penulis lakukan adalah sebagai berikut:

1. Observasi

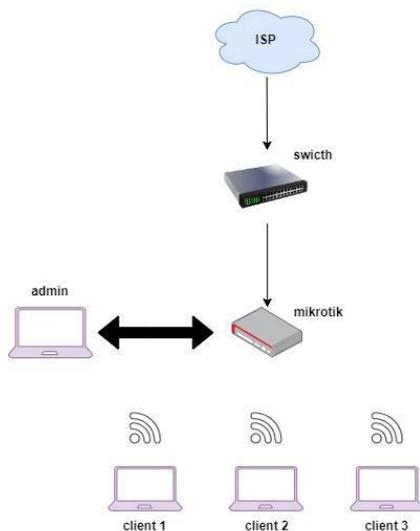
Pengumpulan data pengguna pada SMK Negeri 1 Lhokseumawe untuk melakukan penelitian terhadap manajemen *bandwidth* menggunakan metode *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket*.

2. Studi Pustaka

Dalam tahapan ini penulis melakukan pengumpulan artikel yang berkaitan dengan penelitian yang akan penulis lakukan guna sebagai acuan dalam melaksanakan penelitian. Studi pustaka ini berupa skripsi, artikel penelitian dan *website* yang didapat dari internet.

C. Rancangan Sistem

Rancangan sistem yang akan dibuat yaitu menggunakan 1 perangkat *switch*, 1 perangkat router, 1PC/laptop sebagai admin dan 3 *client*. Topologi yang digunakan pada metode *simple queue*, *queue tree* dan *hierarchical token bucket* sama, hanya saja saat melakukan konfigurasi yang berbeda. Pada perancangan ini ISP digunakan untuk menghubungkan pengguna ke sumber jaringan internet global. Konektivitas jaringan ISP dapat menghubungkan pengguna ke internet melalui kabel maupun nirkabel. Perancangan ini juga menggunakan perangkat keras *switch* yang akan dihubungkan ke router mikrotik. Router mikrotik ini yang nantinya akan menghubungkan jaringan lokal ke sumber internet. Kemudian terdapat PC/laptop admin yang digunakan untuk melakukan konfigurasi mikrotik dan terdapat PC/laptop *client* yang digunakan untuk pengujian. Berikut rancangan topologi yang disimulasikan seperti pada Gambar 2 berikut.



Gambar 2. Rancangan Sistem

Berdasarkan rancangan topologi pada gambar yang akan diterapkan pada SMK Negeri 1 Lhokseumawe untuk menghubungkan beberapa *client* dalam satu jaringan hotspot yang bertujuan untuk melakukan manajemen *bandwidth* pada setiap *client* yang terhubung.

D. Teknik Pengujian

Tahapan ini dilakukan teknik pengujian manajemen *bandwidth* menggunakan metode *simple queue*, *queue tree* dan *hierarchical token bucket* dengan membangun rancangan sistem terlebih dahulu kemudian melakukan konfigurasi mikrotik melalui aplikasi winbox. Setelah sistem selesai dirancang dan dikonfigurasi, kemudian melakukan

manajemen *bandwidth* untuk memberikan *limit* pada masing-masing *client* dan dilakukan pengujian terhadap sistem yang telah dibuat. Pengujian dilakukan dengan melakukan *upload* dan *download file* pada *google drive* dan disaat bersamaan aplikasi wireshark dibuka untuk merekam dan menganalisis *traffic* yang berjalan, kemudian dilakukan perhitungan untuk mendapatkan nilai QoS seperti *throughput*, *packet loss*, *delay* dan *jitter* berdasarkan standar Tiphon.

Setelah data dari hasil pengujian didapatkan, maka selanjutnya membuat grafik untuk melihat metode mana yang lebih unggul untuk diterapkan pada SMK Negeri 1 Lhokseumawe dengan melakukan analisa perbandingan manajemen *bandwidth* antara *simple queue*, *queue tree* dan *hierarchical token bucket*.

III. HASIL DAN PEMBAHASAN

A. Implementasi Sistem

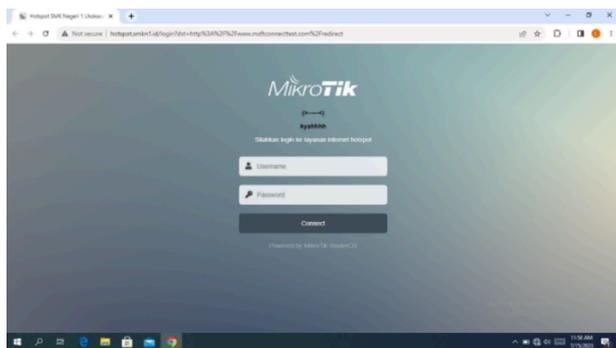
Proses penerapan perangkat keras melibatkan pemasangan perangkat jaringan seperti *switch*, mikrotik, kabel LAN dan PC/laptop. Perancangan ini dilakukan dengan menghubungkan *switch* ke mikrotik menggunakan kabel LAN. Kabel LAN akan dihubungkan ke mikrotik pada *ethernet1* sebagai sumber jaringan internet. Kemudian mikrotik akan dihubungkan ke PC admin menggunakan kabel LAN yang dimasukkan ke *ethernet2* sebagai jaringan lokal. Pembagian *bandwidth* pada jaringan hotspot dilakukan pada ruang kelas dan ruang lab yang nantinya akan digunakan oleh guru dan siswa.

Selanjutnya menerapkan perangkat lunak (*software*) menggunakan aplikasi winbox yang digunakan untuk melakukan konfigurasi pada mikrotik. Konfigurasi mikrotik pada winbox digunakan sebagai tempat administrasi. Hal ini bertujuan untuk dapat digunakannya *Graphical User Interface* sehingga memudahkan pada proses konfigurasi.

B. Konfigurasi Dasar Mikrotik

Agar manajemen *bandwidth* dapat dilakukan pada *simple queue*, *queue tree* dan *hierarchical token bucket*, harus dilakukan konfigurasi dasar mikrotik terlebih dahulu. Tujuannya adalah untuk mengatur perangkat mikrotik agar berfungsi secara optimal dalam jaringan. Untuk melakukan konfigurasi mikrotik dibutuhkan aplikasi winbox. Pada winbox terdapat *neighbors* yang berfungsi untuk melakukan *scanning IP* terhadap router yang telah terkoneksi ke PC.

Selanjutnya melakukan konfigurasi hotspot dengan mengaktifkan fitur hotspot pada mikrotik untuk autentikasi berupa halaman *login* yang diberikan nama hotspot smk1. Kemudian membuat user dan password yang akan digunakan oleh siswa dan guru saat masuk ke halaman *login* untuk melakukan *connect* agar terhubung ke jaringan hotspot yang telah dibuat seperti pada Gambar 3 berikut.

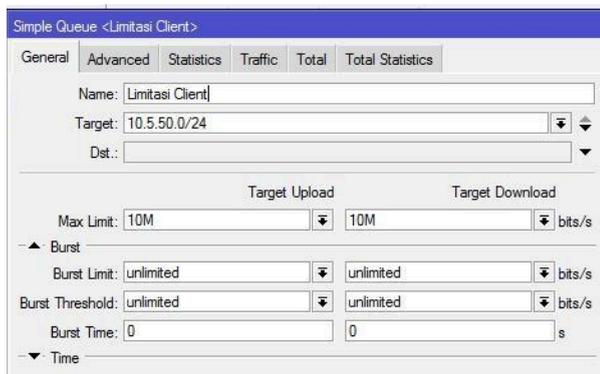


Gambar 3. Halaman Login Hotspot

Berikutnya melakukan manajemen *bandwidth* pada metode *simple queue*, *queue tree* dan *hierarchical token bucket*.

C. Konfigurasi Metode Simple Queue

Simple queue adalah sebuah metode untuk melakukan manajemen *bandwidth* dengan cara membagi *bandwidth* dari skala kecil sampai menengah dan dapat mengatur *bandwidth upload* dan *download* tiap *client*. Pada saat melakukan konfigurasi *simple queue*, IP *client* yang ingin kita *limit* ditentukan terlebih dahulu atau berdasarkan *interface* dengan membuat target *max limit* nya. *Max limit* ini adalah batasan maksimal yang bisa dicapai. *Max limit* untuk *upload* diberikan sebesar 10M begitupun dengan *max limit* untuk *download* diberikan sebesar 10M seperti pada Gambar 4 berikut.



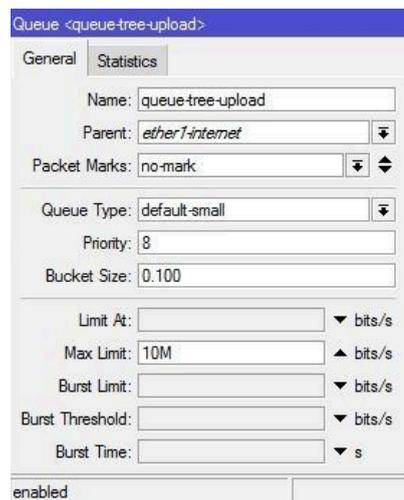
Gambar 4. Simple Queue Upload dan Download

D. Konfigurasi Metode Queue Tree

Queue Tree terdapat dalam mikrotik yang digunakan untuk mengatur jumlah *bandwidth* yang memiliki fungsi lebih kompleks dalam limitasi *bandwidth*. *Queue Tree* adalah pelimitan yang sangat rumit karena berdasarkan protokol, port dan IP *address*.

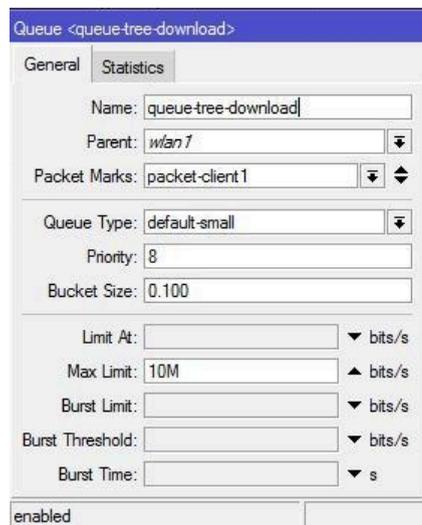
Sebelum melakukan *limit* pada *queue tree*, langkah awal yang dilakukan yaitu melakukan konfigurasi *mangle* pada *firewall* terlebih dahulu. Kemudian membuat *mark connection* dan *mark packet*.

Selanjutnya membuat *limit queue tree upload*, *parent* yang digunakan yaitu *ether1-internet* dan diberikan *max limit* sebesar 10M seperti pada Gambar 5 berikut.



Gambar 5. Queue Tree Upload

Kemudian membuat *limit queue tree download*, *parent* yang digunakan yaitu *wlan1* diberikan *max limit* sebesar 10M seperti pada Gambar 6 berikut.

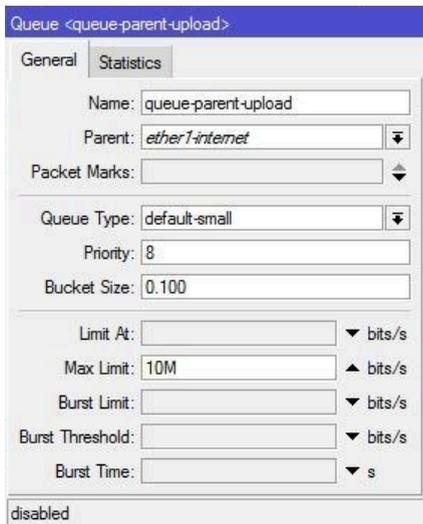


Gambar 6. Queue Tree Download

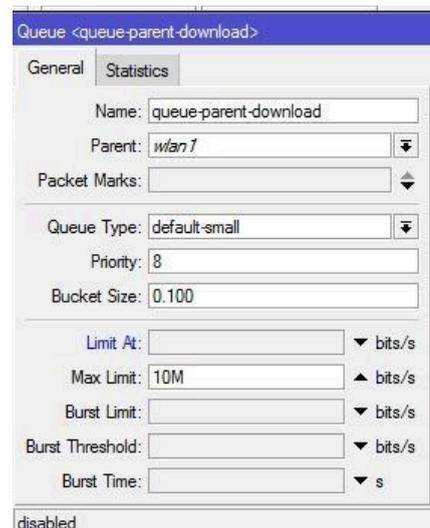
E. Konfigurasi Metode Hierarchical Token Bucket

Hierarchical token bucket melakukan *bandwidth* manajemen dengan membuat struktur *queue* dalam bentuk pola hirarki. HTB dapat melakukan *limit* tidak hanya pada setiap *user* akan tetapi dapat melakukan *limit* setiap divisi-divisi nya. HTB juga dapat menentukan hubungan antara *parent* dan *child* kemudian *child* yang satu dengan *child* yang lainnya.

Cara melakukan *limit* pada HTB *upload* adalah dengan membuat *queue* untuk *parent* terlebih dahulu. Setelah itu menentukan *interface* yang akan digunakan, yaitu *ether1-internet* sebagai *parent*, tujuannya *interface* secara total dapat menyalurkan *traffic*. Kemudian menentukan *max limit* yang digunakan sebesar 10M. Pada *parent* tidak perlu menentukan *packet mark* karena *packet mark* akan ditentukan pada *child*, seperti pada Gambar 7 berikut.



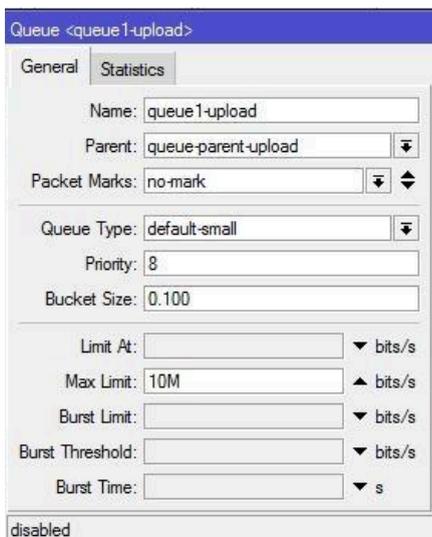
Gambar 7. Queue Parent Upload



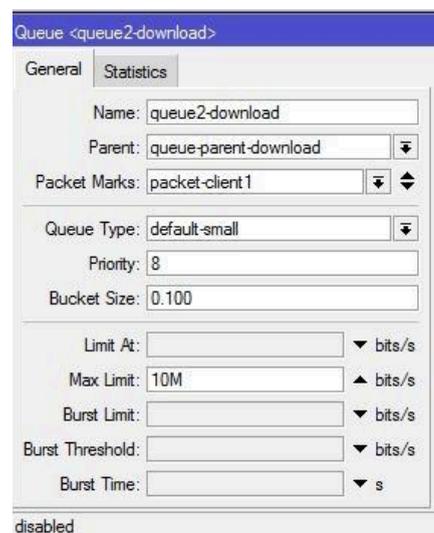
Gambar 9. Queue Parent Download

Selanjutnya membuat *queue* untuk *child* yang akan dimasukkan ke *queue* parent dengan mengubah *parent* menjadi *queue-parent-upload* dan *packet mark* diisi dengan *no-mark* kemudian *max limit* diberikan sebesar 10M, seperti pada Gambar 8 berikut.

Berikutnya membuat *queue* untuk *child* yang akan dimasukkan ke *queue* parent dengan mengubah *parent* menjadi *queue-parent-download* dan *packet mark* diisi dengan *packet-client1* kemudian diberikan *max limit* sebesar 10M, seperti pada Gambar 10 berikut.



Gambar 8. Queue Child Upload



Gambar 10. Queue Child Download

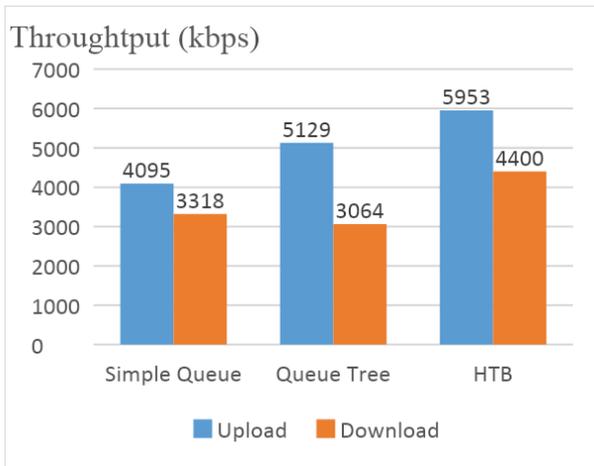
Kemudian membuat *limit* pada HTB *download* dan menentukan *interface* yang digunakan yaitu *wlan1* sebagai *parent* dan menennukan *max limit* yang digunakan sebesar 10M. Pada *parent* tidak perlu menentukan *packet mark* karena *packet mark* akan ditentukan pada *child queue*, seperti pada Gambar 9 berikut.

F. Hasil Pengujian

Setelah melakukan manajemen *bandwidth* pada *simple queue*, *queue tree* dan *hierarchical token bucket* kemudian melakukan pengujian *upload* dan *download file* pada *google drive*.

1. Hasil Pengujian Throughput

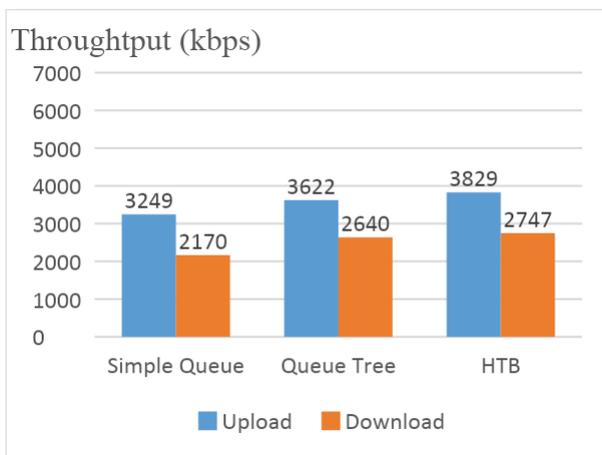
Pengujian *throughput* dilakukan untuk mengetahui kecepatan pada saat mengirim paket data. Adapun hasil nilai pengujian pada ruang lab dengan dilakukan tiga kali percobaan dari ketiga metode dapat dilihat pada Gambar 11berikut.



Gambar 11. Grafik Upload dan Download Ruang Lab

Pada *throughput upload* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 5953 kbps, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 4095 kbps dan *Queue Tree* sebesar 5129 kbps. Pada *throughput download* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 4400 kbps, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 3318 kbps dan *Queue Tree* sebesar 3064 kbps.

Selanjutnya melakukan pengujian pada ruang kelas dengan dilakukan tiga kali percobaan dari ketiga metode dapat dilihat pada Gambar 12 berikut.

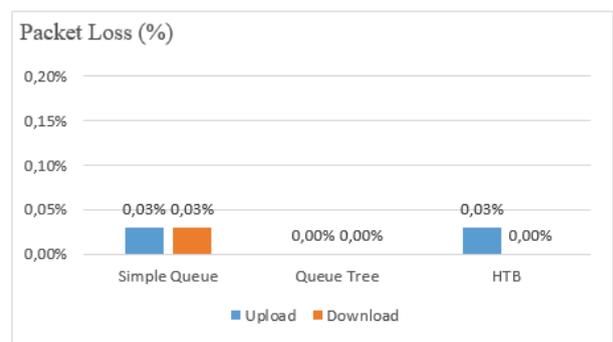


Gambar 12. Grafik Upload dan Download Ruang Kelas

Pada *throughput upload* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 3829 kbps, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 3249 kbps dan *Queue Tree* sebesar 3622 kbps. Pada *throughput download* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 2747 kbps, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 2170 kbps dan *Queue Tree* sebesar 2640 kbps.

2. Hasil Pengujian Packet Loss

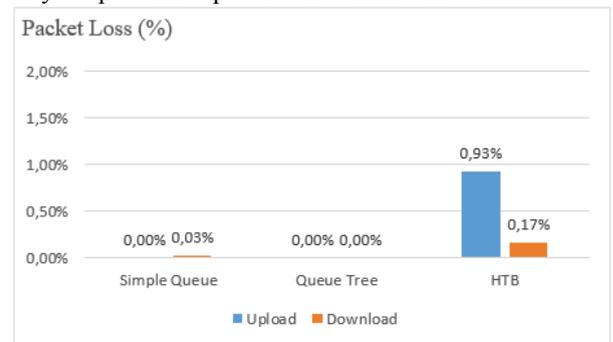
Pengujian *packet loss* dilakukan untuk mengetahui suatu kondisi yang menunjukkan jumlah total paket yang tidak sampai pada tujuan diakibatkan oleh kemacetan pada jaringan. Adapun hasil nilai pengujian pada ruang lab dengan dilakukan tiga kali percobaan dari ketiga metode dapat dilihat pada Gambar 13 berikut.



Gambar 13. Grafik Upload dan Download Ruang Lab

Pada *packet loss upload* nilai rata-rata lebih unggul dimetode *Queue Tree* sebesar 0,0% sedangkan *Simple Queue* dan *Hierarchical Token Bucket* memiliki nilai rata-rata sebesar 0,03%. Pada *packet loss download* nilai rata-rata lebih unggul dimetode *Queue Tree* dan *Hierarchical Token Bucket* sebesar 0,0%, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 0,03%.

Selanjutnya melakukan pengujian pada ruang kelas dan hasilnya dapat dilihat pada Gambar 14 berikut.

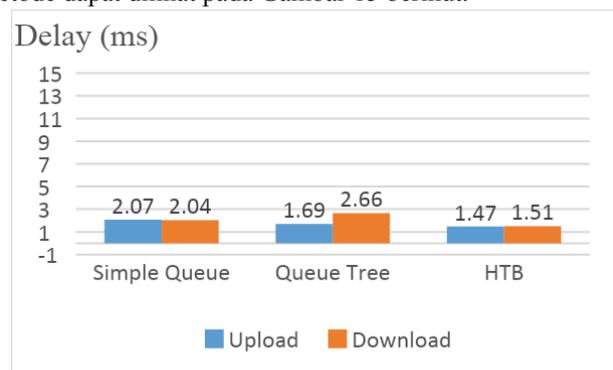


Gambar 14. Grafik Upload dan Download Ruang Kelas

Pada *packet loss upload* nilai rata-rata lebih unggul dimetode *Queue Tree* dan *Simple Queue* sebesar 0,0% sedangkan metode *Hierarchical Token Bucket* memiliki nilai rata-rata sebesar 0,93%. Pada *packet loss download* nilai rata-rata lebih unggul dimetode *Queue Tree* sebesar 0,0%, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 0,03% dan *Hierarchical Token Bucket* sebesar 0,17%.

3. Hasil Pengujian Delay

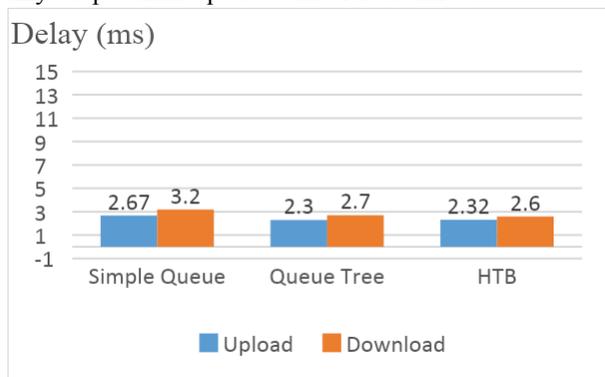
Pengujian *delay* dilakukan untuk mengetahui keterlambatan waktu yang dibutuhkan data untuk menempuh dari jarak asal ke tujuan. Adapun hasil nilai pengujian pada ruang lab dengan dilakukan tiga kali percobaan dari ketiga metode dapat dilihat pada Gambar 15 berikut.



Gambar 15. Grafik Upload dan Download Ruang Lab

Pada *delay upload* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 1,47 ms sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 2,07 ms dan *Queue Tree* sebesar 1,69 ms. Pada *delay download* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 1,51 ms, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 2,04 ms dan *Queue Tree* sebesar 2,66 ms.

Selanjutnya melakukan pengujian pada ruang kelas dan hasilnya dapat dilihat pada Gambar 16 berikut.

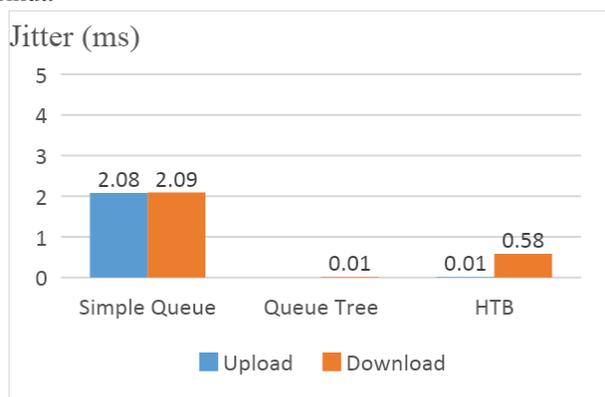


Gambar 16. Grafik Upload dan Download Ruang Kelas

Pada *delay upload* nilai rata-rata lebih unggul dimetode *Queue Tree* sebesar 2,30 ms sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 2,67 ms dan *Hierarchical Token Bucket* sebesar 2,32 ms. Pada *delay download* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 2,60 ms, sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 3,20 ms dan *Queue Tree* sebesar 2,70 ms.

4. Hasil Pengujian Jitter

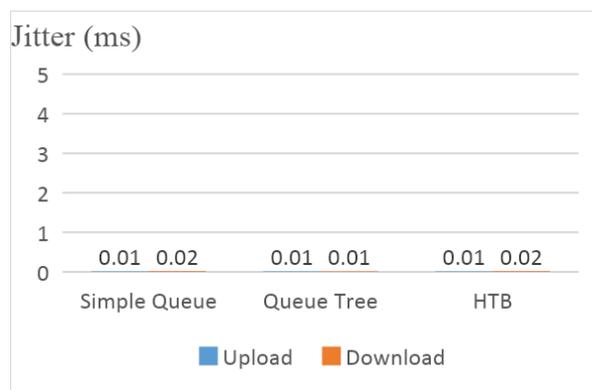
Pengujian *jitter* dilakukan untuk memperlihatkan besaran variasi *delay* dalam saluran data pada suatu jaringan. Adapun hasil nilai pengujian pada ruang lab dengan dilakukan tiga kali percobaan dari ketiga metode dapat dilihat pada Gambar 17 berikut.



Gambar 17. Grafik Upload dan Download Ruang Lab

Pada *jitter upload* nilai rata-rata lebih unggul dimetode *Queue Tree* sebesar 0,00 ms sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 2,08 ms dan *Hierarchical Token Bucket* sebesar 0,01 ms. Pada *jitter download* nilai rata-rata lebih unggul dimetode *Queue Tree* sebesar 0,01 ms sedangkan *Simple Queue* memiliki nilai rata-rata sebesar 2,09 ms dan *Hierarchical Token Bucket* sebesar 0,58 ms.

Selanjutnya melakukan pengujian pada ruang kelas dan hasilnya dapat dilihat pada Gambar 18 berikut.



Gambar 18. Grafik Upload dan Download Ruang Kelas

Pada *jitter upload* nilai rata-rata unggul diketiga metode yaitu sebesar 0,01 ms. Sedangkan pada *jitter download* nilai rata-rata lebih unggul dimetode *Hierarchical Token Bucket* sebesar 0,01 ms dan *Simple Queue*, *Queue Tree* memiliki nilai rata-rata sebesar 0,02 ms.

G. Analisa Hasil Perbandingan Metode *Simple Queue*, *Queue Tree* dan *Hierarchical Token Bucket*

Penggunaan koneksi pada metode *Simple Queue* kurang fleksibel untuk jaringan yang lebih kompleks karena memiliki batasan *bandwidth* tertentu dan juga memiliki skala yang kecil. Pada metode *Queue Tree* dalam melakukan manajemen *bandwidth* penggunaan koneksi jaringan lebih kompleks, karena pembatasan *bandwidth* berdasarkan alamat IP, protokol atau port sehingga *Queue Tree* dapat mengendalikan lalu lintas jaringan. Sedangkan pada metode *Hierarchical Token Bucket* pembagian *bandwidth* dilakukan dengan pola hirarki. Setiap *user* yang terhubung ke dalam jaringan memiliki *token bucket* sendiri. Dengan demikian metode *Hierarchical Token Bucket* memberikan tingkat kontrol yang tinggi dan pembatasan *bandwidth* yang lebih kompleks dan dinamis.

Dari hasil pengujian tersebut, setelah dilakukan perbandingan dari ketiga metode maka manajemen *bandwidth* pada metode *Hierarchical Token Bucket* jauh lebih baik dalam mengontrol penggunaan internet yang digunakan oleh setiap *client*. Metode *Hierarchical Token Bucket* dapat mengoptimalkan lalu lintas jaringan secara terperinci dan fleksibel, khususnya dalam jaringan besar dan kompleks. Sementara pada metode *Simple Queue* hanya cocok digunakan untuk lingkungan yang berskala kecil karena kurangnya prioritas dan memiliki batasan *bandwidth* tertentu. Sedangkan metode *Queue Tree* memiliki jaringan yang lebih kompleks dan memerlukan tingkat kontrol yang lebih tinggi dan jika terjadi kesalahan pada *traffic* sulit untuk diidentifikasi masalah tersebut. Dengan demikian metode *Hierarchical Token Bucket* akan lebih tepat jika diterapkan pada SMK Negeri 1 Lhokseumawe.

IV. SIMPULAN

Adapun simpulan dari hasil pengujian dengan melakukan 3 kali percobaan pada ruang lab dan ruang kelas maka diperoleh hasil nilai rata-rata sebagai berikut:

1. Nilai rata-rata upload throughput dan delay pada ruang lab lebih unggul dimetode HTB 5953 kbps dan 0,03 ms.

- Sedangkan nilai packet loss dan jitter lebih unggul dimetode Queue Tree 0,0% dan 0,0 ms.
2. Nilai rata-rata download throughput dan delay pada ruang lab lebih unggul dimetode HTB 4400 kbps dan 1,51 ms. Nilai packet loss lebih unggul dimetode Queue Tree dan HTB 0,0% dan nilai jitter lebih unggul pada metode Queue Tree 0,01 ms.
 3. Nilai rata-rata upload throughput pada ruang kelas lebih unggul dimetode HTB 3829 kbps. Nilai packet loss lebih unggul dimetode Simple Queue dan Queue Tree 0,0%. Nilai delay lebih unggul dimetode Queue Tree 2,30 ms. Dan nilai jitter unggul diketiga metode 0,01 ms.
 4. Nilai rata-rata download throughput, delay dan jitter pada ruang kelas lebih unggul dimetode HTB 4400 kbps, 1,51 ms dan 0,01 ms. Sedangkan nilai packet loss lebih unggul dimetode Queue Tree 0,0%.
 5. Dari hasil pengujian tersebut, metode Hierarchical Token Bucket lebih stabil pada saat melakukan manajemen bandwidth dan dapat mengontrol penggunaan internet oleh setiap client dengan baik.

REFERENSI

- [1] Tambunan, A.A., dan Lukman,. “Analisis Perbandingan Quality Of Service (Qos) Pada Performa Bandwidth Jaringan Dengan Metode Hierarchical Token Bucket (Htb) Dan Per Connection Queque (Pcq)”. *Respati*, vol. 15, no. 3, hal. 24, 2020.
- [2] Fadilla, “Implementasi Metode Queue Tree untuk Optimalisasi Jaringan pada Router Mikrotik”. *Politeknik Negeri Lhokseumawe*, 2022.
- [3] Aditya, Y. R. K, dan Suraya, “Perbandingan Metode Simple Queue Dan Queue Tree Untuk Optimasi Manajemen Bandwidth Menggunakan Mikrotik (Studi Di Asrama Wisma Muslim)”. *Jarkom*, vol. 7, no. 2, hal. 150–159, 2019.
- [4] Santoso, J. D., “Analisis Perbandingan Metode Queue Pada Mikrotik”, *Pseudocode*, vol. 7, no. 1, hal. 1–7, 2020.
- [5] Putra, K.G., Santyadiputra, G.S., dan Kesiman, W. A., “Penerapan Manajemen Bandwidth Menggunakan Metode Hierarchical Token Bucket Pada Layanan Hotspot Mikrotik Undiksha”, *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 5, no. 1, hal. 146, 2020.
- [6] Toresa, D., Lisawita, L. dan Renadi, F., “Analisa Qos dengan Simple Queue, Queue Tree, dan Hierarchical Token Bucket (Studi Kasus Pro Net Bangkinang)”, *INOVTEK Polbeng - Seri Inform.*, vol. 5, no. 1, hal. 1, 2020.