

SISTEM MANAJEMEN *BANDWIDTH* DENGAN *HIERARCHICAL TOKEN BUCKET* PADA LINUX SERVER CENTOS (STUDI KASUS: JARINGAN KANTOR UNIVERSITAS MADURA)

Andhis Yoga Prasetya¹, Ubaidi^{2,*}

^{1,2}Program Studi Informatika, Fakultas Teknik, Universitas Madura
Email: *ubed@unira.ac.id

ABSTRACT

The internet is one of the media needed in the Madura University Office, whether to find information or download and upload research files. The Peer Connection Queue (HTB) method used for bandwidth distribution in the Madura University Office LAN network is weak. It is not optimal in using the network when traffic is heavy, and there is a bandwidth leak, and the divided bandwidth is not following the rules. Hierarchical Token Bucket (HTB) is a method that can maximize bandwidth and manage bandwidth on the network hierarchically with a priority system based on the class on the network. This study aims to compare the previous method, namely PCQ, with HTB, which is implemented on a Linux CentOS server. The method will be compared using the calculation of QoS (Quality Of Service), which is a method of calculating network quality, in this study will calculate delay, jitter, throughput on the PCQ and HTB methods. In this study, the authors use the open-source HTB-Tools application to implement the HTB method, which is expected to maximize bandwidth usage and comfort users in the Madura University environment.

Keywords: Bandwidth, Hierarchical Token Bucket, CentOS, HTB-Tools

ABSTRAK

Internet merupakan salah satu media yang dibutuhkan di lingkungan Kantor Universitas Madura, baik itu untuk mencari informasi, ataupun mengunduh dan mengupload berkas penelitian. Metode Peer Connection Queue (HTB) yang digunakan untuk pembagian bandwidth di jaringan LAN Kantor Universitas Madura memiliki kelemahan sehingga tidak optimal dalam penggunaan jaringan saat *traffic* sedang padat dan terjadi kebocoran *bandwidth* sehingga *bandwidth* yang terbagi tidak sesuai dengan aturan. Hierarchical Token Bucket (HTB) merupakan metode yang mampu memaksimalkan *bandwidth* dan mengatur *bandwidth* pada jaringan secara hierarki dengan sistem prioritas berdasarkan kelas yang berada pada jaringan. Tujuan dari penelitian membandingkan metode sebelumnya yaitu PCQ dengan HTB yang diimplementasikan di server Linux CentOS. Metode tersebut akan dibandingkan menggunakan perhitungan QoS (Quality Of Service) yang merupakan metode perhitungan kualitas jaringan, pada penelitian ini akan menghitung *delay*, *jitter*, *throughput* pada metode PCQ dan HTB. Pada Penelitian ini penulis menggunakan aplikasi *open source* HTB-Tools untuk menerapkan metode HTB yang diharapkan dapat memaksimalkan penggunaan *bandwidth* serta memberi kenyamanan bagi pengguna di lingkungan Universitas Madura.

Kata Kunci: *Bandwidth*, Hierarchical Token Bucket, CentOS, HTB-Tools

1. Pendahuluan

Universitas Madura merupakan salah satu Perguruan Tinggi swasta yang didirikan oleh Yayasan

Universitas Madura (YUM) pada tanggal 02 Mei 1978 bertempat di Jl. Raya Panglegur KM. 3,5 Pamekasan. Aktivitas pada Universitas Madura

dituangkan dalam Tri Dharma Perguruan Tinggi, yaitu Pendidikan dan Pengajaran, Penelitian dan Pengabdian kepada Masyarakat. Untuk mencapai aktivitas tersebut, dibutuhkan akses jaringan internet yang stabil terutama karyawan yang bekerja di lingkungan Gedung Rektorat ataupun Fakultas yang menggunakan jaringan kantor.

Internet merupakan salah satu media yang dibutuhkan di lingkungan Kantor Universitas Madura, baik itu untuk mencari informasi, *chatting*, bermain *game online* untuk mengisi jam istirahat, ataupun mengunduh dan mengunggah berkas penelitian. Manajemen *bandwidth* yang diterapkan pada jaringan kantor di Universitas Madura yaitu menggunakan perangkat *router* mikrotik dengan teknik pembagian *bandwidth* metode PCQ. Metode yang digunakan untuk pembagian *bandwidth* saat ini di jaringan LAN Kantor Universitas Madura memiliki kelemahan sehingga tidak optimal dalam penggunaan jaringan saat *traffic* sedang padat dan terjadi kebocoran *bandwidth* sehingga *bandwidth* yang terbagi tidak sesuai dengan aturan.

Dalam kasus ini, maka diperlukan batasan dalam penggunaan jaringan internet, agar proses pengiriman data atau berkas penelitian berjalan dengan tepat dan cepat. Selain itu, saat semua *client* aktif secara bersamaan dalam mengakses internet akan menyebabkan *traffic* padat sehingga akan terjadi keterlambatan dalam pengiriman data, maka dari itu diperlukannya manajemen *bandwidth* yang ada sistem prioritas sehingga dalam keadaan *traffic* padat system tersebut bisa membantu *client* yang memiliki hak akses tersebut. Oleh karena itu, dibutuhkan sebuah metode manajemen *bandwidth* yang baik agar

bisa mengatasi masalah tersebut, HTB merupakan teknik pembagian *bandwidth*, pembagian dilakukan secara hirarki yang dibagi-bagi ke dalam kelas sehingga mempermudah pengaturan *bandwidth*.

Hierarchical Token Bucket (HTB) merupakan metode yang mampu memaksimalkan *bandwidth* dan mengatur *bandwidth* pada jaringan secara hierarki dengan sistem prioritas berdasarkan kelas yang berada pada jaringan [1]. Untuk dapat menerapkan sistem tersebut konfigurasi dilakukan pada sistem operasi Linux server CentOS. Linux server CentOS nantinya digunakan sebagai *router* atau *gateway* bagi *client* yang berbasis CLI (*command line interface*), adapun *tools* yang digunakan untuk membantu dalam penerapan HTB yang akan berjalan di server CentOS tersebut. Diharapkan dengan adanya penggunaan metode ini dapat memaksimalkan *bandwidth* saat terjadinya kepadatan *traffic* yang berada di jaringan kantor universitas madura sehingga koneksi jaringan internet menjadi nyaman dan akurat sesuai dengan pemakaiannya.

2. Tinjauan Pustaka

2.1. *Bandwidth*

Bandwidth adalah kapasitas atau daya tampung kabel *ethernet* agar dapat dilewati *traffic* paket data dalam jumlah tertentu. *Bandwidth* juga bisa berarti jumlah konsumsi paket data per satuan waktu dinyatakan dengan satuan *bit per second* (bps). *Bandwidth* internet disediakan oleh *provider* internet dengan jumlah tertentu tergantung sewa pelanggan. *Bandwidth* dapat didefinisikan sebagai kapasitas atau daya tampung suatu *channel* komunikasi (medium komunikasi) untuk dapat dilewati sejumlah *traffic*

informasi atau data dalam satuan waktu tertentu [2].

2.2. Manajemen *Bandwidth*

Manajemen merupakan pembaruan dari bahasa Inggris “to manage” yang mempunyai arti mengatur, mengurus dan mengelola. Sedangkan *bandwidth* adalah lebar data yang dapat diproses di suatu komunikasi data melalui jaringan computer yang dihitung dengan besaran *bit per second* [2]–[7].

Berdasarkan definisi di atas maka dapat disimpulkan bahwa manajemen *bandwidth* yaitu suatu kegiatan mengatur agar data yang lewat sesuai kapasitas maksimal atau *bandwidth* di dalam suatu jaringan komputer yang terhubung dengan internet agar kualitas jaringan terjamin. Ada lima tipe manajemen *bandwidth* yang penulis ketahui yaitu, *queue simple*, *queue tree*, *peer connection queue*, *class based queue*, dan *hierarchical token bucket*. Lima tipe manajemen *bandwidth* ini memiliki keunggulan masing-masing [8][9]. Namun, dalam permasalahan ini penulis hanya menggunakan tipe *hierarchical token bucket* sebagai metode yang akan digunakan dalam penelitian.

2.3. Linux

Nama Linux berasal dari nama pembuatnya, yang diperkenalkan tahun 1991 oleh Linus Torvalds. Peralatan sistem dan pustakanya umumnya berasal dari sistem operasi GNU, yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi GNU adalah dasar dari munculnya nama alternatif GNU/Linux[10].

Linux telah lama digunakan sebagai sistem operasi server, dan didukung oleh perusahaan-perusahaan komputer. Linux juga digunakan sebagai sistem operasi di berbagai

macam jenis perangkat keras komputer, termasuk desktop, *supercomputer*, dan sistem benam seperti yang akan digunakan dalam penelitian ini sebagai *router*. Para pengamat teknologi informatika beranggapan kesuksesan Linux dikarenakan Linux tidak bergantung kepada vendor (*vendor independence*), biaya operasional yang rendah, dan kompatibilitas yang tinggi dibandingkan versi UNIX tak bebas, serta faktor keamanan dan kestabilannya yang tinggi dibandingkan dengan sistem operasi lainnya seperti Microsoft Windows. Ciri-ciri ini juga menjadi bukti atas keunggulan model pengembangan perangkat lunak sumber terbuka [10][11].

2.4. PCQ (Peer Connection Queue)

PCQ (Peer Connection Queue) merupakan sebuah metode pembagian *bandwidth* dengan cara pembagian dengan *subclass* atau *subqueue* pada masing-masing *user*. Berdasarkan klasifikasinya dengan metode ini umum digunakan karena dapat diimplementasikan dengan mudah untuk membatasi *bandwidth* secara dinamis dengan konfigurasi sederhana [12][13]. Namun metode tersebut belum bisa memaksimalkan penggunaan *bandwidth* untuk di lingkup kampus, karena dalam jaringan kantor Universitas Madura ada yang harus diprioritaskan sewaktu-waktu maka dari itu penulis mencoba menggunakan metode HTB yang mana metode tersebut memaksimalkan *bandwidth* dengan sistem prioritasnya.

2.5. Hierarchical Token Bucket

Hierarchical Token Bucket (*HTB*) merupakan teknik antrian yang mirip dengan CBQ hanya saja perbedaannya terletak pada opsi, HTB

lebih sedikit opsi saat konfigurasi serta lebih presisi. Teknik antrian HTB memberikan kita fasilitas pembatasan *traffic* pada setiap level maupun klasifikasi, *bandwidth* yang tidak terpakai bisa digunakan oleh klasifikasi yang lebih rendah. Kita juga dapat melihat HTB seperti suatu struktur organisasi dimana pada setiap bagian memiliki wewenang dan mampu membantu bagian lain yang memerlukan. Teknik antrian HTB sangat cocok diterapkan pada perusahaan dengan banyak struktur organisasi [8].

Token Bucket Filter (TBF) membatasi *bandwidth* dengan metode *shape & drop*, prinsip kerja menggunakan aliran *token* yang memasuki *bucket* dengan kecepatan (*rate*) konstan, jika *token* dalam *bucket* habis maka paket data akan diantri dan kelebihannya dibuang, setiap paket data yang dikeluarkan identik dengan *token*. *Token* dalam *bucket* akan lebih cepat habis jika aliran paket data melampaui kecepatan *token* memasuki *bucket*, jadi kita asumsikan bahwa trafik melebihi batas konfigurasi [8].

2.6. Quality of Service

QoS (Quality Of Service) merupakan mekanisme jaringan yang memungkinkan aplikasi-aplikasi atau layanan dapat beroperasi sesuai dengan yang diterapkan. Tujuan dari QoS untuk memenuhi kebutuhan-kebutuhan layanan yang berbeda, yang menggunakan infrastruktur yang sama [1].

QoS adalah teknik untuk mengelola *bandwidth*, *delay*, *jitter*, *throughput* dan *packet loss* untuk aliran dalam jaringan. Tujuan dari mekanisme QoS adalah mempengaruhi setidaknya satu diantara empat parameter dasar QoS yang telah ditentukan [14][15]. Namun penulis

hanya akan menguji kualitas *delay*, *jitter*, dan *throughput*.

Berikut ini merupakan beberapa parameter QoS yang akan digunakan dalam mengukur performansi jaringan, yaitu :

1. *Delay* adalah waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media fisik, kongesti atau juga waktu proses yang lama [14].

$$Delay = \frac{Total\ Delay}{Total\ paket\ yang\ diterima} \quad (1)$$

2. *Jitter* diakibatkan oleh variasi-variasi dalam panjang antrian, dalam waktu pengolahan data dan juga dalam waktu penghimpunan ulang paket-paket diakhir perjalanan *jitter* [15].

$$Jitter = \frac{Total\ Variasi\ Delay}{Total\ paket\ diterima-1} \quad (2)$$

3. *Throughput* adalah kecepatan (*rate*) transfer data yang efektif yang diukur dalam bps. *Throughput* merupakan jumlah total kedatangan paket yang sukses yang diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut [15].

$$Throughput = \frac{Paket\ Data\ Diterima}{Lama\ Panegmatan} \quad (3)$$

3. Metode

3.1. Metode Pengumpulan Data

Terdapat beberapa metode pengumpulan data yang akan digunakan peneliti untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian.

1. Studi Pustaka

Studi Pustaka yang dilakukan peneliti dengan mengumpulkan data yang diperlukan mulai dari penelitian-penelitian yang berhubungan dengan

penelitian saat ini. Yang diharapkan dapat mendukung serta menambah pengetahuan dalam melaksanakan penelitian ini.

2. Observasi

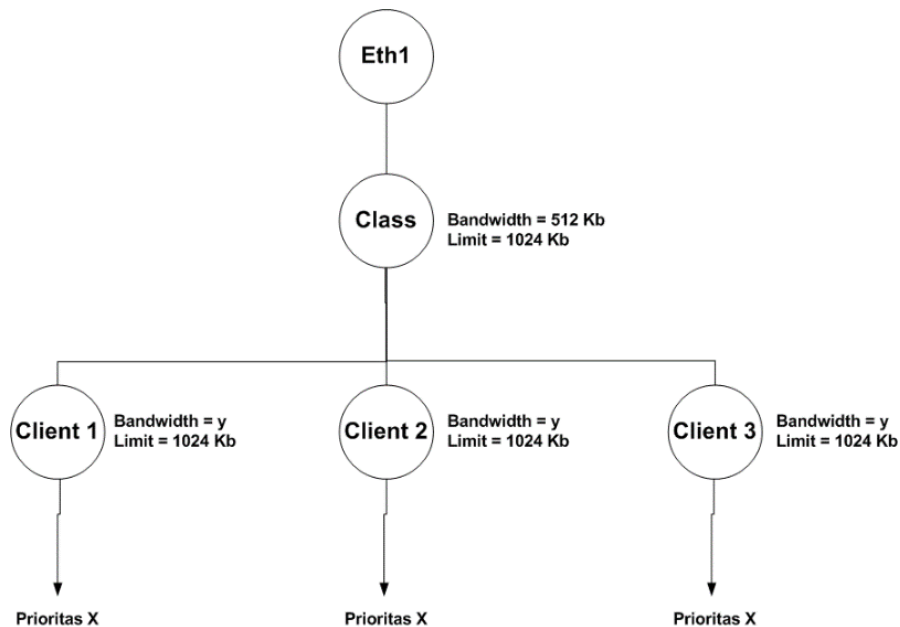
Di fase ini peneliti melakukan observasi dan melakukan pengamatan jaringan kantor Universitas Madura, pengamatan dilakukan agar mampu menganalisa yang akan menjadi bahan evaluasi untuk mengimplemetasikan sistem manajemen *bandwidth* yang akan dilakukan. Serta pengamatan manajemen *bandwidth* yang digunakan sebelumnya yang nanti menjadi bahan perbandingan untuk metode yang akan diterapkan saat ini.

Serta melakukan observasi terhadap perangkat yang akan digunakan,

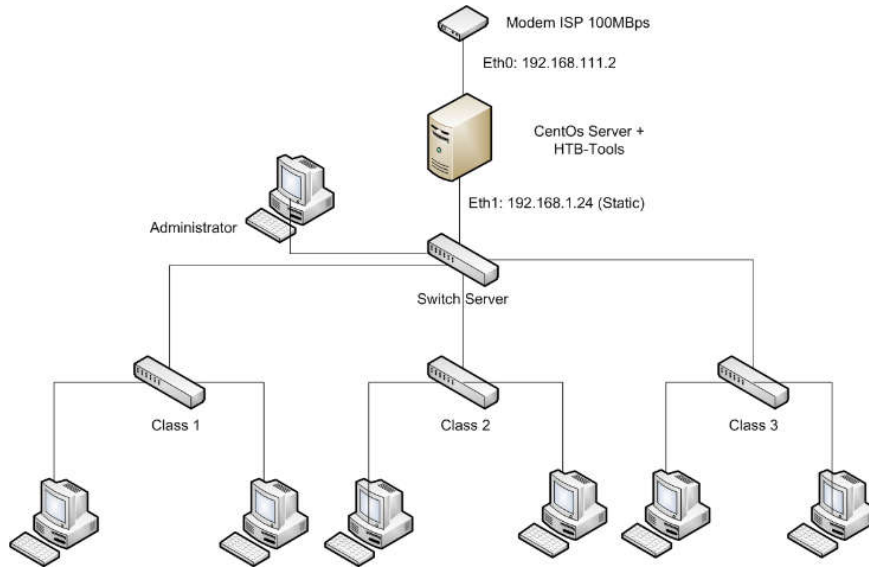
Perangkat yang digunakan untuk penelitian ini menggunakan sebuah *computer* server yang didalam terdapat sistem operasi Linux Centos Server, untuk aplikasi manajemen *bandwidth* peneliti menggunakan aplikasi *open source* HTB-Tools yang diklaim menawarkan kemampuan untuk pembatasan *traffic* yang sesuai dengan metode peneliti gunakan.

3.2. Skema Perancangan

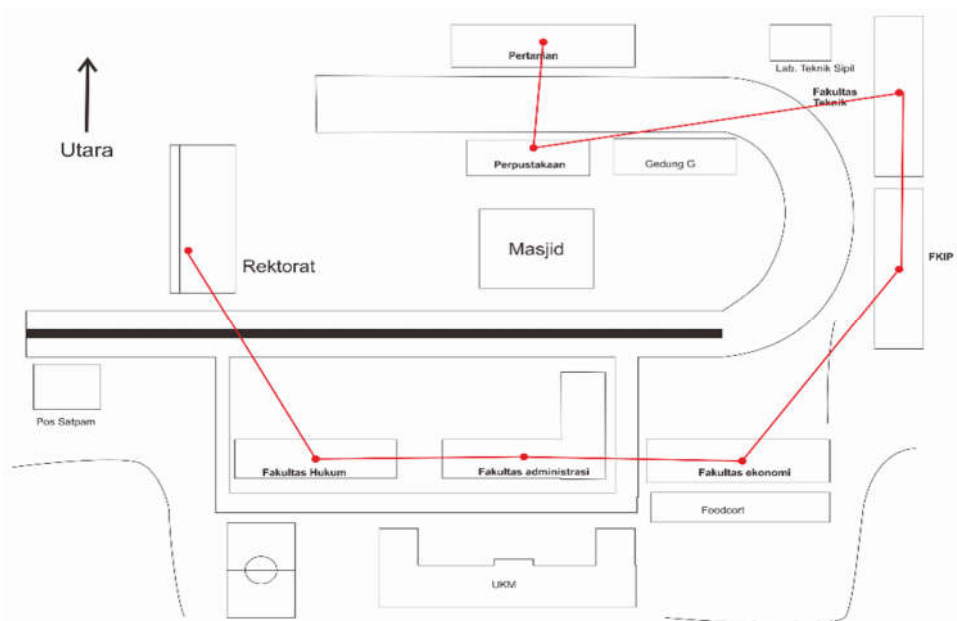
Perancangan manajemen *bandwidth* yang dilakukan dengan menggunakan HTB-Tools di jaringan kantor Universitas Madura, berikut adalah skema perancangan antrian seperti pada Gambar 1.



Gambar 1. Skema manajemen *bandwidth* secara hierarki



Gambar 2. Topologi



Gambar 3. Desain Layout Topologi

Dalam skema pada Gambar 1 yang dibuat dapat dijelaskan “eth1” yang merupakan ethernet pada jaringan lokal (LAN) membuat sebuah kelas dengan besar *bandwidth* 512 Kb dengan limit *bandwidth* 1024 Kb atau dapat diperjelas besaran *bandwidth* yang akan didapatkan maksimal sebesar 1025 Kb oleh *client*. “Class” tersebut mempunyai 3 *client*, masing-masing *client* akan mendapatkan limit

sebesar 1024 Kb dan *bandwidth* dengan variabel Y yang bersifat *integer*, *bandwidth* tersebut akan ditentukan sesuai dengan kebutuhann client namun tidak boleh melebihi limit yang telah tentukan. Sedangkan prioritas X merupakan sebuah kebijakan yang bisa didapatkan oleh masing-masing *client* nantinya. Maka dari itu perancangan yang akan dibuat penulis berdasarkan kebutuhan yang

ada di kantor Universitas Madura, ada 7 class yang nantinya akan dibuat di Gedung Rektorat dan 7 class di Fakultas beserta Perpustakaan. Gambar 2 dan Gambar 3 adalah gambar sederhana dari topologi jaringan yang nanti akan diterapkan di jaringan kantor Universitas Madura.

Pada tahap implementasi, percobaan dilakukan bertahap mulai dari proses instalasi dan konfigurasi untuk manajemen *bandwidth* serta melakukan seberapa mampu server mentransfer data ke *client* yang jaraknya 5 gedung dari server pusat. Membandingkan Quality of Service dari metode sebelumnya dan sesudah, serta percobaan *transfer rate* dengan mengambil mengambil 2 sampel yaitu Gedung Rektorat yang dekat dengan server dan Laboratorium Informatika yang jaraknya 5 gedung dari server pusat.

4. Hasil dan Pembahasan

4.1. Pengujian Sistem untuk Mengetahui Hasil QoS

Setelah sistem dirancang dan diimplementasikan di jaringan kantor universitas madura, selanjutnya dilakukannya pengujian untuk mengetahui seberapa maksimal sistem berjalan mulai dari perbandingan QoS *delay*, *jitter*, dan *throughput* metode sebelum dan sesudah serta pengujian sistem manajemen *bandwidth* menggunakan HTB-Tools di Linux Server CentOS. Mekanisme perbandingan dengan menggunakan mekanisme QoS (Quality of Service) untuk menguji seberapa baik metode HTB dibandingkan dengan yang sebelumnya yang menggunakan metode PCQ. Mekanisme perbandingan QoS pada penelitian ini meliputi *delay*, *jitter*, dan *throughput* dengan percobaan 5 *client*. Maka dari

itu berikut hasil percobaan yang dilakukan sebagai berikut:

1. Delay.

Dalam penelitian ini, *delay* di uji untuk membandingkan mana yang lebih menghasilkan waktu tunda antaran metode manajemen *bandwidth* PCQ dan HTB.

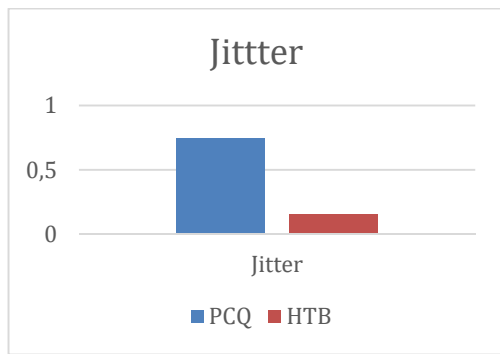
Tabel 1. Delay Per Masing-Masing Client Menggunakan PCQ dan HTB

Client ke-	Menggunakan metode PCQ	Menggunakan metode HTB
1	0,066819	0,148199
2	0,198923	0,123975
3	0,413515	0,189494
4	0,178357	0,136927
5	0,339280	0,148927
Total	1,196894	0,747522
Rata-rata delay	0,2393788	0,149504

Dari analisa data pada Tabel 1 yang menggunakan aplikasi *wireshark* per masing-masing *client* melakukan aktivitas *streaming* ataupun *searching* baik menggunakan metode PCQ maupun metode HTB dari data diatas didapatkan nilai *delay* antara masing-masing metode manajemen *bandwidth* tersebut, manajemen *bandwidth* menggunakan PCQ dengan nilai *delay* 0,2393788 ms, dan setelah menggunakan metode manajemen *bandwidth* HTB mendapatkan nilai *delay* 0,149504 ms. Dari pengujian *delay* yang telah dilakukan dapat disimpulkan bahwa metode PCQ mendapatkan nilai *delay* lebih besar dibandingkan setelah menggunakan metode HTB, hal itu disebabkan telah dilakukannya manajemen *bandwidth* sehingga terkontrol di masing-masing *client* sehingga nilai *delay* dengan menggunakan metode HTB relatif kecil.

2. Jitter

Jitter didefinisikan sebagai variasi delay yang diakibatkan oleh panajang queue dalam suatu pengolahan data dan reassemble paket-paket data di akhir pengiriman akibat kegagalan sebelumnya.



Gambar 4. Perbandingan jitter PCQ dan HTB

Pada Gambar 4 diperoleh nilai jitter pada manajemen bandwidth menggunakan metode PCQ lebih besar daripada dengan menggunakan HTB, nilai jitter untuk metode PCQ adalah 0,74 ms, sedangkan dengan menggunakan metode HTB mendapat nilai sebesar 0,15 ms. Dari hasil nilai tersebut manajemen bandwidth yang telah dilakukan dengan metode HTB relatif lebih kecil daripada metode PCQ, hal ini dengan menggunakan metode HTB transfer data lebih cepat karena bandwidth setiap clientnya terbagi rata.

3. Throughput

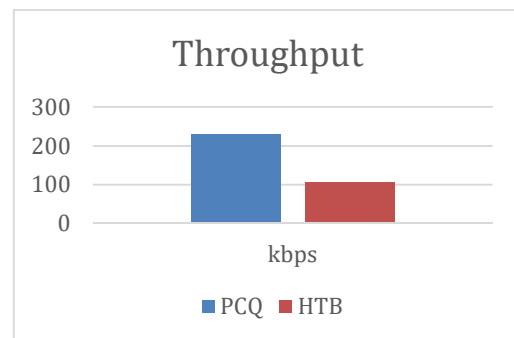
Throughput tidak sama dengan bandwidth, apabila dianalogikan dalam bentuk nyata, bandwidth sama halnya dengan sebuah pipa sedangkan air yang mengalir dalam sebuah pipa tersebut adalah throughput.

Dari pengujian throughput yang telah dilakukan, diperoleh nilai throughput dengan menggunakan manajemen bandwidth PCQ maupun HTB, seperti tersaji pada Tabel 2.

Diperoleh pada manajemen bandwidth PCQ nilai throughput sebesar 231,347 kbps, sedangkan pada manajemen bandwidth HTB memperoleh nilai throughput sebesar 107,019 kbps. Dari data tersebut nilai throughput dengan menggunakan metode HTB lebih kecil dibanding dengan metode PCQ, dikarenakan pembatasan bandwidth yang diberikan pada client menyebabkan kecepatan transfer data menurun. Gambar 5 merupakan perbandingan pengujian throughput dari metode PCQ dan HTB.

Tabel 2. Throughput Per Masing-Masing Client Menggunakan PCQ dan HTB

Client ke-	Menggunakan metode PCQ (kbps)	Menggunakan metode HTB (kbps)
1	69,908	120,741
2	192,855	87,891
3	366,767	134,767
4	150,21	98,625
5	376,995	93,074
Total	1156,995	535,098
Rata-rata Throughput	231,347	107,019



Gambar 5. Perbandingan throughput PCQ dan HTB

4.2. Pengujian Sistem Di Lokasi

Pada tahap pengujian atau percobaan yang dilakukan pada penelitian ini dengan menggunakan HTB-Tools, tahap pengujian dilakukan di 2 (dua) tempat yakni di

ruang simat yang merupakan tempat yang dekat dengan tempat server dan Laboratorium Informatika yang merupakan tempat atau ruang yang jauh dari server, Laboratorium Informatika dipilih untuk mengukur seberapa efektif *transfer rate* sampai pada ruangan tersebut. Pengujian dilakukan dengan manajemen *bandwidth* sama tanpa prioritas dan pengujian dengan *bandwidth* sama dengan prioritas menggunakan 5 *client* untuk percobaan.

A. Pengujian di ruang Simat Gedung Rektorat

Pertama pengujian yang dilakukan di ruang simat Rektorat dengan cara mengatur *bandwidth* yang diberikan pada *client* sama namun tanpa prioritas. Berdasarkan dari pengujian dengan pengaturan *bandwidth* sama tanpa prioritas di ruang simat, berikut hasil dari pengujian dituliskan dalam Tabel 3. Pengujian yang dilakukan dengan limit sama serta pembagian *bandwidth* yang sama dan menggunakan prioritas pada setiap *client* yang berada di ruangan simat menghasilkan *transfer rate* sama-sama rata hanya berbeda beberapa kbps dari masing-masing *client*. Kedua Pengujian yang dilakukan di ruang simat Rektorat dengan cara mengatur *bandwidth* yang diberikan pada *client* sama namun dengan prioritas. Dari hasil pembagian dan pengaturan yang dilakukan dengan membagi *bandwidth* masing-masing 5 *client* 409 kbps, namun 5 (lima) *client* tersebut mendapatkan prioritas berbeda. Hasil pengujian *bandwidth* sama dengan prioritas dituliskan dalam Tabel 4.

Pengujian yang dilakukan dengan limit sama serta pembagian *bandwidth* yang sama namun menggunakan prioritas untuk *client* perbedaan prioritas berpengaruh sangat signifikan

pada *transfer rate* yang dimiliki *client* PMB sebesar 471,42 kbps.

Tabel 3. Pengujian *Bandwidth* Sama Tanpa Prioritas

Nama	<i>Bandwidth</i> (kbps)	Limit (kbps)	Prioritas	<i>Transfer Rate</i> (kbps)
PMB	409	2048	0	397,11
BAA_1enovo	409	2048	0	396,11
Pak_e di	409	2048	0	400,68
Pak_Chip	409	2048	0	393,33
ubed	409	2048	0	404,23

Tabel 4. Pengujian *Bandwidth* Sama Dengan Prioritas

Nama	<i>Bandwidth</i> (kbps)	Limit (kbps)	Prioritas	<i>Transfer Rate</i> (kbps)
PMB	409	2048	1	471,42
BAA_1enovo	409	2048	2	400,42
Pak_e di	409	2048	3	378,53
Pak_Chip	409	2048	2	396,46
ubed	409	2048	3	370,50

B. Pengujian di Laboratorium Informatika

Pengujian kedua dilakukan di Laboratorium Informatika yang jaraknya jauh dari server pusat HTB, uji coba dilakukan untuk mencoba seberapa mampu server mentransfer data dengan pengaturan *bandwidth* yang telah dilakukan, pengujian sama dengan pengujian yang dilakukan di ruang simat Rektorat dengan menggunakan konfigurasi *bandwidth* sama tanpa prioritas dan *bandwidth* sama menggunakan prioritas dengan 5 (lima) *client* pengujian. Pengujian pertama dengan *bandwidth* sama tanpa prioritas. Dari hasil pengujian dengan *bandwidth* sama tanpa prioritas yang

dilakukan di Laboratorium Informatika, berikut hasil pengujian dituliskan di Tabel 5. Dengan pengujian yang telah dilakukan dengan pembagian *bandwidth* sama tanpa prioritas di Laboratorium Informatika yang jarak jauh dari server pusat HTB didapatkan *transfer rate* yang sama hanya berbeda beberapa kbps. Kedua dengan pengujian di Laboratorium Informatika dengan pengaturan *bandwidth* sama namun menggunakan prioritas, Dari hasil pengujian yang dilakukan di Laboratorium Informatika dengan pengaturan *bandwidth* sama dengan prioritas, berikut hasil pengujian dituliskan dalam Tabel 6.

Tabel 5. Pengujian *Bandwidth* Sama Tanpa Prioritas Di Laboratorium Informatika

Nama	<i>Bandwidth</i> (kbps)	Limit (kbps)	Prioritas	<i>Transfer Rate</i> (kbps)
User 1	409	2048	0	393,82
User 2	409	2048	0	400,03
User 3	409	2048	0	409,20
User 4	409	2048	0	396,11
User 5	409	2048	0	400,03

Tabel 6. Pengujian *Bandwidth* Sama Dengan Prioritas di Laboratorium Informatika

Nama	<i>Bandwidth</i> (kbps)	Limit (kbps)	Prioritas	<i>Transfer Rate</i> (kbps)
User 1	409	2048	3	361,81
User 2	409	2048	4	257,00
User 3	409	2048	1	458,86
User 4	409	2048	3	388,41
User 5	409	2048	2	400,03

Pengujian yang dilakukan di Laboratorium Informatika dengan limit *bandwidth* dan *bandwidth* yang diberikan ke *client* sama namun menggunakan prioritas sangat berpengaruh dengan jarak dari server pusat pada *transfer rate* yang dimiliki

oleh masing-masing *client*. *Transfer rate* yang dimiliki oleh *user 3* dengan angka prioritas 1 sebesar 458,86 kbps berbeda sangat jauh dengan *user 2* dengan angka prioritas 4 sebesar 257,00 kbps.

5. Penutup

5.1. Kesimpulan

Dari hasil penelitian dan pengujian yang dilakukan dapat diambil kesimpulan bahwa secara keseluruhan dengan metode pengukuran jaringan menggunakan QoS dari nilai *delay* dan *jitter* dengan perbandingan metode manajemen *bandwidth* PCQ dan HTB, nilai *delay* 0,149 ms dan *jitter* 0,15 ms dari metode HTB lebih kecil dibanding *delay* 0,239 dan *jitter* 0,74 ms dari metode PCQ. *Throughput* pada metode manajemen *bandwidth* HTB yang dihasilkan menurun jauh dibandingkan PCQ karena tidak ada pembatasan *bandwidth* bagi *client* di PCQ sedangkan HTB menggunakan batasan *bandwidth* maka dari itu yang menyebabkan transfer data yang diterima *client* menurun. Pengujian yang dilakukan di 2 (dua) tempat berbeda di ruang simat yang berada didekat server dan Laboratorium Informatika yang jauh dari server, *transfer rate* yang dihasilkan dari server dibidang optimal dikarenakan *client* mendapatkan kuota *bandwidth* sesuai dengan pengaturan *bandwidth* menggunakan HTB. Metode HTB dan aplikasi HTB-Tools yang diimplementasikan pada Linux Server CentOS dapat melakukan pembatasan *bandwidth* dari seluruh *client* yang ada.

5.2. Saran

Untuk pengembangan dan perbaikan dimasa yang akan data maka penulis memberikan beberapa saran

yaitu untuk penelitian selanjutnya sebaiknya manajemen *bandwidth* menggunakan HTB dilakukan dengan sistem GUI agar mempermudah dalam konfigurasinya dan juga dikembangkan atau diterapkan di jaringan lainnya. Untuk pengembangan selanjutnya perlu dicoba di OS lain nya yang bisa *support* dengan HTB-Tools.

6. Daftar Pustaka

- [1] H. Hardiman, L. F. Aksara, and S. Subardin, "Analisis perbandingan QoS (Quality Of Service) Pada Manajemen Bandwidth Dengan Metode PCQ (Per Connection Queue) Dan HTB (Hierarchical Token Bucket)," *semanTIK*, vol. 4, no. 1, pp. 121–128, 2018.
- [2] A. I. Wijaya and L. B. Handoko, "Manajemen Bandwidth Dengan Metode Htb (Hierarchical Token Bucket) Pada Sekolah Menengah Pertama Negeri 5 Semarang," *Jurnal Teknik Informatika Udinus*, vol. 1, no. 1, pp. 1–3, 2015.
- [3] Wirabakti, K. Imtihan, and A. S. Pardiansyah, "Proxy Server dan Management Bandwidth Jaringan Komputer Menggunakan Mikrotik RB952Ui5ac2nD," *Jurnal Informatika & Rekayasa Elektronika*, vol. 1, no. 1, pp. 44–49, 2018.
- [4] A. Syukur, "Analisis Management Bandwidth Menggunakan Metode Per Connection Queue (PCQ) dengan Autentikasi RADIUS," *IT Journal Research and Development*, vol. 2, no. 2, pp. 78–89, 2018.
- [5] S. Agustini and A. Mudzakir, "Rancang Bangun Jaringan Komputer dengan Bandwidth Manajemen Menggunakan Teknik Brust Limit dan Firewall sebagai Pengaman Jaringan," *Jurnal Ilmiah NERO*, vol. 4, no. 3, pp. 189–195, 2019.
- [6] A. R. Muttaqi, S. Wahjuni, and S. N. Neyman, "Manajemen Alokasi Bandwidth Layanan Internet Menggunakan Fractional Knapsack Problem," *Jurnal Teknologi dan Sistem Komputer*, vol. 7, no. 2, pp. 71–76, 2019.
- [7] M. B. S. Saragih, I. Gunawan, I. O. Kirana, Sumarno, and H. Qurniawan, "Mikrotik Hotspot Network Implementation Using Simple Queue As Bandwidth Management," *Jurnal Mantik*, vol. 3, no. 4, pp. 758–765, 2020.
- [8] B. Santosa, "Manajemen Bandwidth Internet dan Intranet Quality of Service," pp. 1–30, 2004.
- [9] D. Kurnia, "Analisis QoS Pada Pembagian Bandwidth Dengan Metode Layer 7 Protocol, PCQ, HTB Dan Hotspot Di SMK Swasta Al-Washliyah Pasar Senen," *CESS (Journal of Computer Engineering System and Science)*, vol. 2, no. 2, pp. 102–111, 2017.
- [10] C. Gray, "Review: Just for Fun - The Story of an Accidental Revolutionary," *The Computer Bulletin*, vol. 45, no. 4, pp. 30–30, 2003.
- [11] A. Qustoniah and Darwanto, "Manajemen Bandwidth Jaringan Komputer Menggunakan Metode Hierarchical Token Bucket (HTB) pada PC Router Berbasis Linux," *Widya Teknika*, vol. 19, no. 1, pp. 16–24, 2011.
- [12] R. Kharisman Ndruru, "Penerapan Metode Additive Ratio Assessment (ARAS) dan Rank Order Centroid (ROC) Dalam Pemilihan Jaksa Terbaik Pada Kejaksaan Negeri Medan," *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)*, pp. 367–372, 2020.

- [13] A. Nurdianto and Deli, "Studi Komparasi Manajemen Bandwidth Antara Metode Hierarchical Token Bucket (HTB) Dan Peer Connection Queue (PCQ)," in *Conference on Business, Social Sciences and Innovation Technology*, 2020, vol. 1, no. 1, pp. 487–497.
- [14] I. Iskandar and A. Hidayat, "Analisa Quality of Service (QoS) Jaringan Internet Kampus (Studi Kasus: UIN Suska Riau)," *Jurnal CoreIT*, vol. 1, no. 2, pp. 67–76, 2015.
- [15] Z. M. Subekti and A. Suwarno, "Analisis Perbandingan Manajemen Bandwidth Jaringan Wifi Autentikasi User Password Metode Hierarchical Token Bucket (HTB) dengan Per Connection Queue (PCQ): Studi Kasus STMIK Bani Saleh," *Informatics, Science and Technologies Journal*, vol. 8, no. 1, pp. 33–42, 2018.