

ANALISIS QOS (QUALITY OF SERVICE) PADA WARNET DENGAN METODE HTB (HIERARCHICAL TOKEN BUCKET)

Azmuri Wahyu Azinar¹⁾, Ragil Sapta Adi²⁾

^{1), 2)} Teknik Informatika, Institut Teknologi Adhi Tama Surabaya
Jl. Arief Rahman Hakim No 100 Surabaya
Email : azmuri@itats.ac.id¹⁾, ragilsapta07@gmail.com²⁾

ABSTRAK

Dalam penggunaan bandwidth di Melati Net yang berada di desa Sonorejo kabupaten Kediri sering kali kurang dimanfaatkan secara optimal oleh pengguna. Sering juga menemukan pengguna yang mengalami ketidakpuasan karena tidak dapat menikmati koneksi internet yang cepat dan dapat mengurangi kunjungan pengguna di warnet tersebut. Sehingga perlu diterapkan manajemen bandwidth dengan metode HTB (Hierarchical Token Bucket) karena dapat memaksimalkan bandwidth yang dipakai. Berdasarkan hasil penelitian yang telah dilakukan, Setiap paket memperoleh bandwidth minimal pada Limit-At dan lebih dari Limit-At tetapi tidak melebihi Max-Limit dari parent selama traffic pada parent tidak penuh, serta pemerataan bandwidth sesuai prioritasnya saat kondisi traffic seluruh paket penuh. HTB (Hierarchical Token Bucket) mampu memaksimalkan bandwidth yang tidak terpakai, sehingga kualitas pelayanan menjadi lebih meningkat. Berdasarkan hasil analisa QoS (Quality of Service) delay tiap paket mengalami peningkatan, seperti browsing 31.82%, streaming 10.13%, download 8.31% dan game online 18.02%. Sedangkan untuk throughput tiap paket mengalami peningkatan, seperti browsing 36.58%, streaming 38.49%, game online 15.22%, kecuali paket download mengalami penurunan 36.47% karena sudah dibagi rata dengan paket yang lainnya sehingga tidak bisa memakai bandwidth melebihi dari batas yang diberikan.

Kata kunci: Quality of Service, HTB (Hierarchical Token Bucket), Bandwidth management

ABSTRACT

The available bandwidth in Melati Net that is located in Sonorejo Kediri, is often not maximally used. Frequently some users are not satisfied because they cannot enjoy the quick internet connection, consequently less users visited the warnet. Therefore it is necessary to apply a bandwidth management using HTB (Hierarchical Token Bucket) because it can maximize bandwidth used. From the survey result, each package gets a minimal bandwidth on Limit_At and more than Limit-At but less than Max-Limit of parents as long as the parent traffic is not busy, and bandwidth distribution used by priority when traffic condition of all packages are full. HTB (Hierarchical Token Bucket) is able to maximize the unused bandwidth that will result better service quality. Result analysis of QoS (Quality of Service), delay of each package is increased, such as browsing 31.82%, steaming 10.13% downloading 8.31% and online game 18.02%. Whereas the throughput of each package is increased, such as browsing 36.58%, streaming 38.49%, game online 15.22% except downloading is decreased by 36.47% because it had been distributed evenly with other packages that can be used above bandwidth limit.

Keywords : Quality of Service, HTB (Hierarchical Token Bucket), Bandwidth management..

1. PENDAHULUAN

Transportasi Semakin berkembangnya teknologi informasi sekarang ini, maka kebutuhan akan informasi semakin meningkat. Dimana setiap orang membutuhkan informasi dalam waktu yang cepat, singkat dan akurat. Oleh karena itu dibutuhkan suatu sarana yang dapat mendukung hal tersebut. Salah satunya adalah koneksi internet yang cepat dan stabil. Warnet (Warung Internet) masih menjadi sebuah pilihan masyarakat untuk dapat menikmati layanan internet.

Bandwidth adalah banyaknya data dalam satuan *bits per second* yang dapat ditranmisikan lewat sebuah *medium* jaringan dalam satu satuan waktu. Ada juga yang mengartikan besaran yang menunjukkan seberapa banyak data yang dapat dilewatkan dalam koneksi melalui sebuah

network [1]. Istilah ini berasal dari bidang teknik listrik, di mana *bandwidth* menunjukkan total jarak yang berkisar diantara jarak tertinggi dan terendah sinyal pada saluran komunikasi. Terdapat dua jenis *bandwidth*, yaitu : *Digital Bandwidth* dan *Analog Bandwidth*. Dengan *bandwidth* tersebut harus bisa melayani beberapa pengguna yang ingin menggunakan internet secara bersamaan. Jika tidak diatur, kemungkinan besar *traffic* dan *bandwidth* akan penuh ketika digunakan oleh semua pengguna.

Bandwidth limiter dengan menggunakan Mikrotik Router adalah salah satu bentuk *management bandwidth* yang sangat mudah dan efisien digunakan oleh setiap penyedia layanan jasa internet karena dengan menggunakan *bandwidth limiter* setiap pengguna akan mendapatkan *bandwidth* dengan kadar atau ukuran yang sama tanpa mengganggu *bandwidth* dari pengguna yang lain [2].

2. DASAR TEORI

2.1. *Quality of Service (QoS)*

QoS merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari satu. *QoS* digunakan untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu servis. *QoS* merupakan mekanisme jaringan yang memungkinkan aplikasi-aplikasi atau layanan dapat beroperasi sesuai dengan yang diharapkan [3].

QoS didesain untuk membantu *end user* menjadi lebih produktif dengan memastikan bahwa user mendapatkan kinerja yang handal dari aplikasi-aplikasi berbasis jaringan. *QoS* mengacu pada kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu melalui teknologi yang berbeda-beda. *QoS* menawarkan kemampuan untuk mendefinisikan atribut-atribut layanan jaringan yang disediakan, baik secara kualitatif maupun kuantitatif [4].

Jika dilihat dari ketersediaan suatu jaringan, terdapat karakteristik kuantitatif yang dapat dikontrol untuk menyediakan suatu layanan dengan kualitas tertentu. Kinerja jaringan dapat dievaluasi berdasarkan parameter – parameter kualitas layanan, yaitu *delay*, *jitter*, *packetloss*, *throughput*.

2.2. *Hierarchical Token Bucket (HTB)*

HTB adalah metode yang berfungsi untuk mengatur pembagian *bandwidth*, pembagian dilakukan secara *hirarki* yang dibagi-bagi kedalam kelas sehingga mempermudah pengaturan *bandwidth* [5]. HTB diklaim menawarkan kemudahan pemakaian dengan teknik peminjaman dan implementasi pembagian trafik yang lebih akurat. Teknik antrian HTB memberikan fasilitas pembatasan trafik pada setiap *level* maupun klasifikasi, *bandwidth* yang tidak terpakai bisa digunakan oleh klasifikasi yang lebih rendah [6].

Ada tiga tipe kelas dalam HTB, yaitu : *root*, *inner*, dan *leaf*. *Root class* berada paling atas, dan semua trafik harus melewati kelas ini. *Inner class* memiliki *parent class* dan *child classes*. Sedangkan *leaf class* adalah *terminal class* yang mempunyai *parent class* tetapi tidak mempunyai *child class*. Pada *leaf class*, trafik dari *layer* yang lebih tinggi disuntikkan melalui klasifikasi yang harus digunakan melalui *filter*, sehingga memungkinkan untuk membedakan jenis trafik dan prioritas. Sehingga, sebelum trafik memasuki *leaf class* harus diklasifikasikan melalui *filter* dengan berbagai *rules* yang berbeda.

Pada antrian HTB mempunyai parameter yang menyusunnya dalam antrian [7] yaitu :

1. *Rate*

Parameter *rate* menentukan *bandwidth* maksimum yang bisa digunakan oleh setiap *class*, jika *bandwidth* melebihi nilai “*rate*”, maka paket data akan dipotong atau dijatuhkan (*drop*).

2. *Ceil*

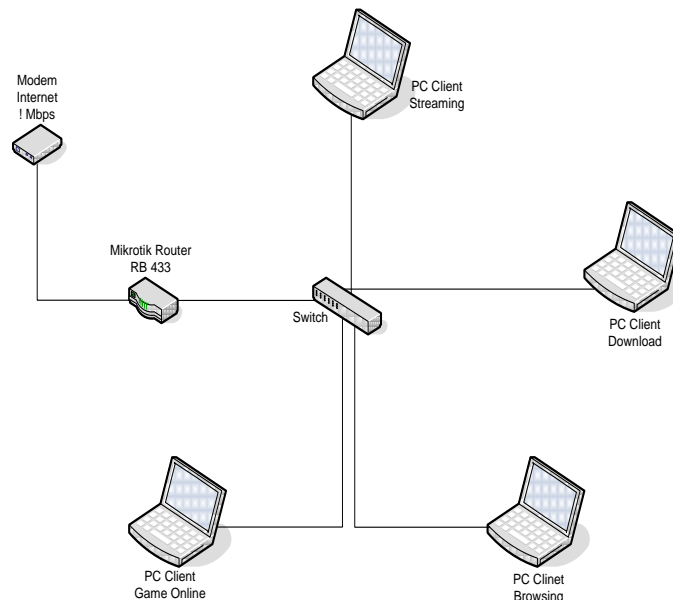
Parameter *ceil* *di-set* untuk menentukan peminjaman *bandwidth* antar *class* (kelas), peminjaman *bandwidth* dilakukan kelas paling bawah ke kelas di atasnya, teknik ini disebut *link sharing*.

3. *Random Early Detection (RED)*

Random Early Detection atau bisa disebut *Random Early Drop* biasanya digunakan untuk *gateway/router backbone* dengan tingkat trafik yang sangat tinggi. RED mengendalikan trafik jaringan sehingga terhindar dari kemacetan pada saat trafik tinggi berdasarkan pemantauan perubahan nilai antrian minimum dan maksimum. Jika isi antrian dibawah nilai minimum, maka *mode 'drop'* tidak berlaku, saat antrian mulai terisi hingga melebihi nilai maksimum, maka RED akan membuang (*drop*) paket data secara acak sehingga kemacetan pada jaringan dapat dihindari. RED juga mempunyai parameter yang menyusunnya, yaitu : *Max, Min, Probability, Limit, Burst, Avpkt, Bandwith, Ecn (Explicit Congestion Notification)*.

3. Metodologi Penelitian

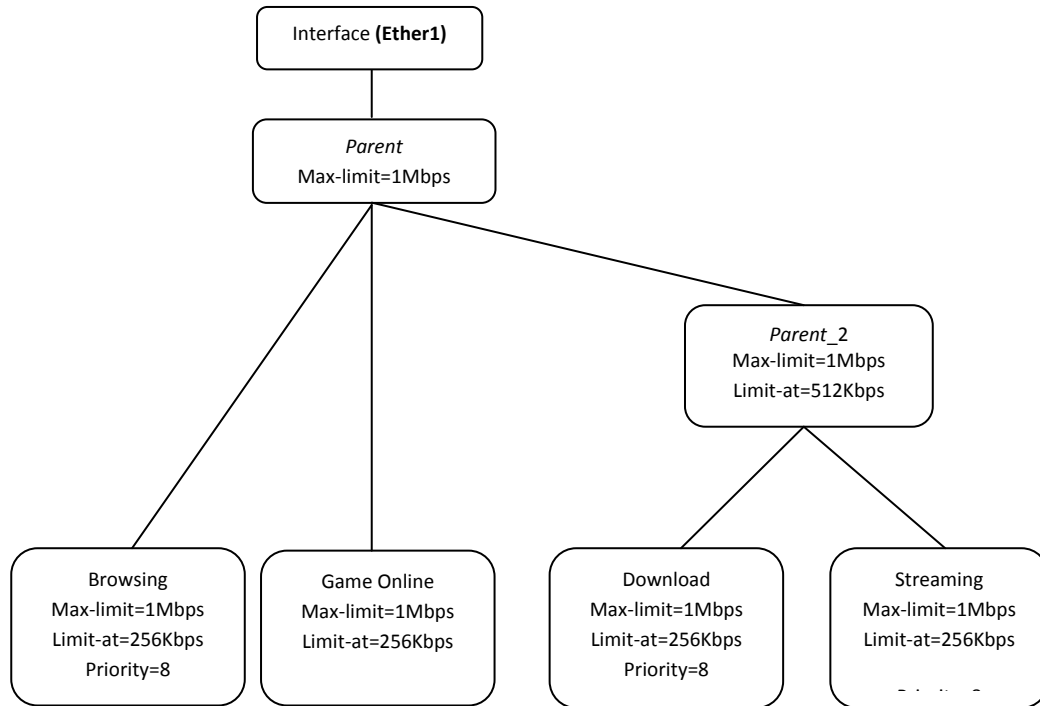
Topologi yang digunakan dalam mengembangkan penelitian ini menggunakan topologi star. Modem internet memiliki bandwidth 1 Mbps yang berfungsi sebagai *access point*. Setelah itu modem di alirkan ke mikrotik untuk proses konfigurasi penerapan metode HTB (*Hierarchical Token Bucket*). Setelah proses konfigurasi selesai, output dari mikrotik di alirkan ke switch dan selanjutnya di alirkan ke PC client untuk menjalankan aktivitas *browsing, streaming, download* dan *game online*.



Gambar 1 Topologi Rancang Bangun Jaringan Warnet

3.1 Arsitektur HTB Pada Warnet

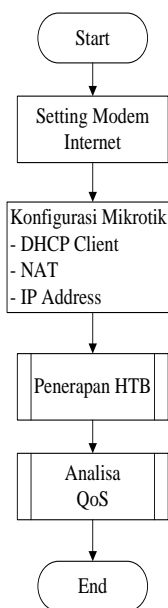
Arsitektur HTB dimulai dengan modem sebagai LAN (*Local Area Network*) *Ether1* diterima sebagai *access point*. Selanjutnya dibuat *queue tree* dengan nama *parent* yang memiliki nilai *max-limit* sebesar 1 Mbps dan tidak memiliki *limit-at*. *Browsing* sebagai *ceil* dan induknya adalah *parent* yang diberi nilai *max-limit* 1 Mbps dan *limit-at* 256 kbps. Untuk paket *game online* sebagai *ceil* dan induknya adalah *parent* yang diberi nilai *max-limit* 1 Mbps dan *limit-at* 256 kbps. *Parent_2* adalah sebagai *ceil* dari *parent* dan juga sebagai *parent* dari paket *download* dan *game online*. *Download* dan *streaming* sebagai *ceil* dimana induknya adalah sebagai *parent_2*. yang diberi nilai *max-limit* 1 Mbps dan *limit-at* 256 kbps. Berikut adalah arsitektur metode HTB pada warnet :



Gambar 2 Arsitektur HTB

3.2 Flowchart Sistem

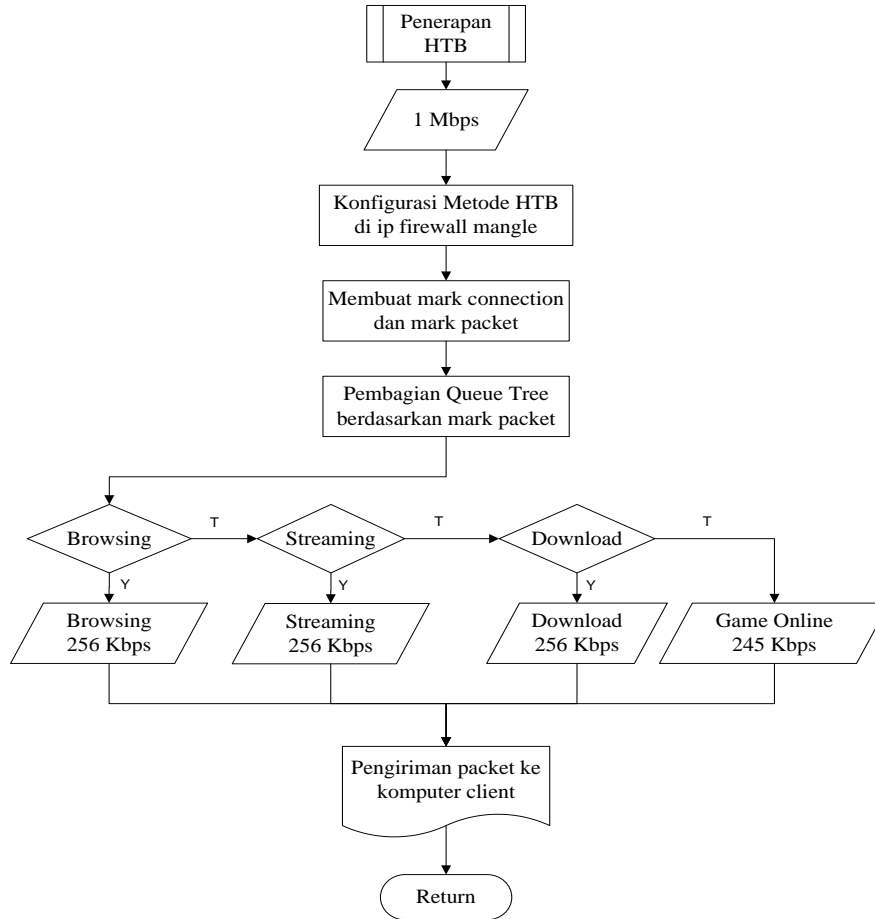
Melakukan Konfigurasi modem internet supaya bisa diterima jaringan internet oleh Mikrotik Router. Setelah itu dilakukan konfigurasi DHCP Client agar mikrotik menerima IP Address dari modem internet. Selanjutnya konfigurasi IP Address agar dapat membentuk jaringan LAN dan melakukan Konfigurasi NAT (*Network Address Translation*) agar jaringan LAN dapat terkoneksi dengan IP Address dari modem internet, sehingga jaringan LAN dapat terkoneksi dengan internet. Selanjutnya penerapan metode HTB dengan melakukan konfigurasi IP Firewall Mangle agar dapat memisahkan jenis koneksi dari client. Setelah proses penerapan metode HTB selesai, maka dilakukan analisa parameter QoS (*Quality of Service*) seperti delay dan throughput. Alur trafik masuk pada sistem sebagai berikut:



Gambar 3 Flowchart Sistem

3.3 Flowchart HTB (Hierarchical Token Bucket)

Bandwidth yang diterima dari *access point* sebesar 1 Mbps. Kemudian dialirkan ke mikrotik router untuk dilakukan konfigurasi metode HTB di ip *firewall mangle*. Selanjutnya membuat *mark connection* dan *mark packet* untuk proses pemfilteran *packet* seperti *packet browsing*, *streaming*, *download* dan *game online*. Setelah dilakukan pembagian *queue tree* berdasarkan *mark packet* yang telah dibuat. Yang terakhir adalah mengirimkan *packet* ke komputer *client* berdasarkan permintaan *packet*.



Gambar 4 Flowchart HTB

4. Pengujian dan Pembahasan

4.1. Pengujian

Pada tahapan ini, peneliti membangun jaringan komputer dengan menggunakan router mikrotik dengan dikonfigurasi metode HTB (*Hierarchical Token Bucket*) dan setelah itu dilakukan pengambilan data, peneliti menggunakan aplikasi perhitungan QoS dengan pemrograman php dan *web service xampp* yang digunakan untuk menghitung QoS yang telah di *capture* dari *wireshark* dan dijalankan lewat aplikasi dengan menggunakan *browser*.

Langkah-langkah dalam tahapan ini terdiri dari:

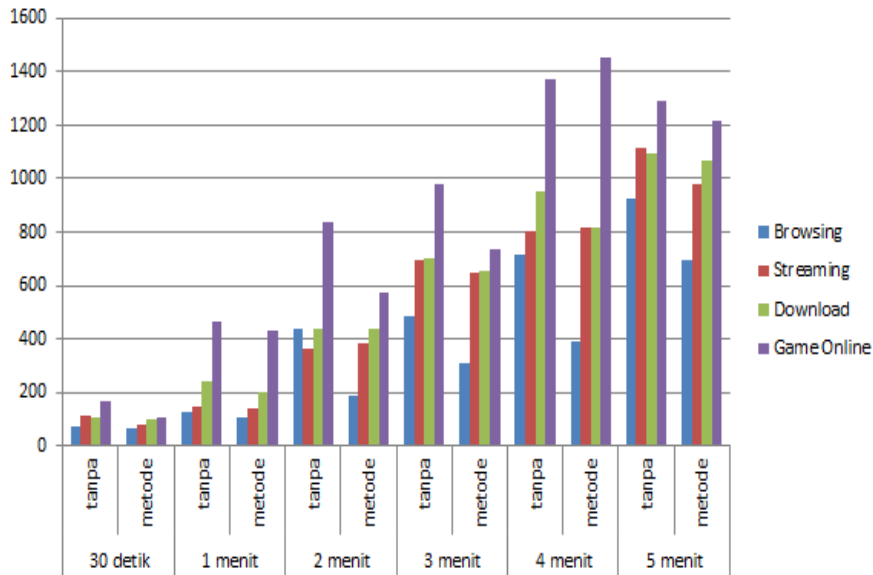
- a. Konfigurasi DHCP *client*
- b. Konfigurasi DHCP *Server* dan DNS *Server*
- c. Konfigurasi NAT (*Network Address Translation*)
- d. Konfigurasi IP *Firewall Mangle*
- e. Konfigurasi *queue tree*

Pengujian pada penelitian ini dilakukan dengan dua cara:

1. Tanpa menggunakan metode HTB
2. Menggunakan metode HTB

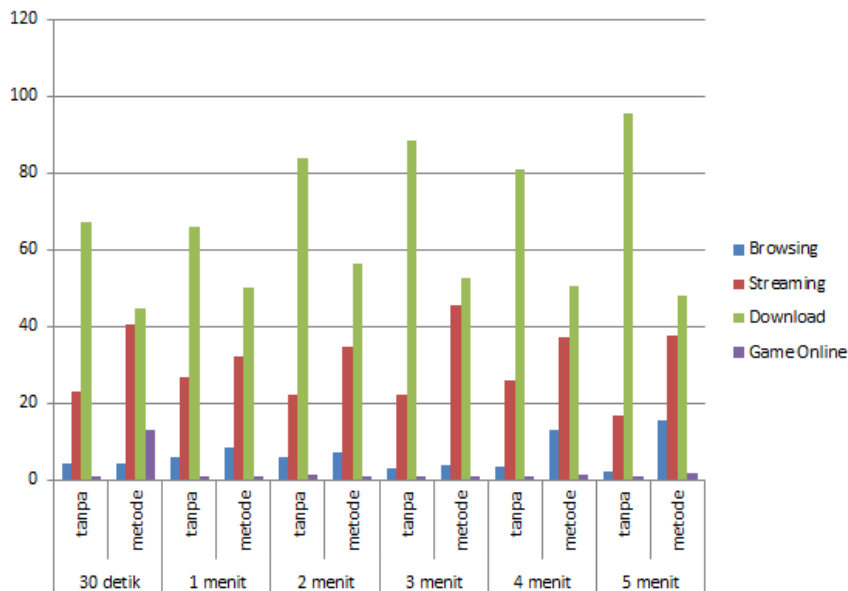
Untuk hasil uji coba keseluruhan *delay* dan *throughput* dapat dilihat pada gambar 5 dan gambar 6.

1. Uji coba perhitungan *delay*



Gambar 5. Grafik *Delay* tanpa metode dan dengan metode

2. Uji coba perhitungan *throughput*



Gambar 6. Grafik *Throughput* tanpa metode dan dengan metode

4.2. Pembahasan

Efisiensi dari hasil uji coba keseluruhan *delay* dan *throughput* browsing, streaming dan game online dapat dilihat pada tabel 1, tabel 2 dan tabel 3

Tabel 1 Efisiensi *delay browsing* dan *throughput browsing*

Waktu	Delay			Throughput		
	tanpa metode	dengan metode	persen	tanpa metode	dengan metode	persen
30 detik	71.408	64.051	10.30 %	4.285	4.431	3.29 %
1 menit	130.846	107.411	17.91 %	6.059	8.383	27.72 %
2 menit	440.033	190.677	56.67 %	6.076	7.282	16.56 %
3 menit	487.195	313.238	35.71 %	3.199	3.769	15.12 %
4 menit	714.053	390.639	45.29 %	3.641	13.037	72.07 %
5 menit	923.849	692.71	25.02 %	2.407	15.721	84.69 %
rata-rata			31.82 %			36.58 %

Tabel 2 Efisiensi *delay streaming* dan *throughput streaming*

Waktu	Delay			Throughput		
	tanpa metode	dengan metode	persen	tanpa metode	dengan metode	persen
30 detik	113.644	82.98	26.98%	23.153	40.392	42.68%
1 menit	150.337	140.323	6.66%	26.935	32.189	16.32%
2 menit	386.114	364.814	5.52%	22.376	34.627	35.38%
3 menit	693.108	645.077	6.93%	22.097	45.54	51.48%
4 menit	820.069	802.61	2.13%	26.041	37.025	29.67%
5 menit	1117.714	977.43	12.55%	16.771	37.625	55.43%
rata-rata			10.13%			38.49%

Tabel 3 Efisiensi *delay game online* dan *throughput game online*

Waktu	Delay			Throughput		
	tanpa metode	dengan metode	persen	tanpa metode	dengan metode	persen
30 detik	165.564	109.734	33.72 %	1.102	1.159	4.92 %
1 menit	468.026	433.968	7.28 %	1.059	1.079	1.85 %
2 menit	837.23	575.162	31.30 %	0.981	1.189	17.49 %
3 menit	976.868	736.786	24.58 %	0.9	1.084	16.97 %
4 menit	1450.347	1368.093	5.67 %	1.108	1.204	7.97 %
5 menit	1288.922	1216.816	5.59 %	1.15	1.987	42.12 %
rata-rata			18.02 %			15.22 %

Efisiensi *delay download* mengalami peningkatan, sedangkan *throughput download* mengalami penurunan karena sudah dibatasi dari paket *download*. Dapat dilihat pada tabel 4

Tabel 4 Efisiensi *delay download* dan *throughput download*

Waktu	Delay			Throughput		
	tanpa metode	dengan metode	persen	tanpa metode	dengan metode	persen
30 detik	108.771	100.779	7.35%	67.229	44.576	33.70%
1 menit	246.315	200.156	18.74%	66.084	49.891	24.50%
2 menit	441.091	437.864	0.73%	83.984	56.42	32.82%
3 menit	701.756	655.977	6.52%	88.597	52.573	40.66%
4 menit	954.138	818.339	14.23%	80.986	50.661	37.44%
5 menit	1092.775	1067.987	2.27%	95.542	48.085	49.67%
rata-rata			8.31%			36.47%

Setelah diterapkan metode HTB (*Hierarchical Token Bucket*) efisiensi *delay* pada semua aktifitas internet mengalami peningkatan, seperti *browsing* mengalami peningkatan 31.82%, *streaming* mengalami peningkatan 10.13%, *download* mengalami peningkatan 8.31% dan *game online* mengalami peningkatan 18.02%. Sedangkan untuk *throughput*, aktifitas *browsing* mengalami kenaikan *throughput* sebesar 36.58%, *streaming* 38.49%, *game online* 15.22%. Sedangkan untuk *throughput download* mengalami penurunan sebesar 36.47%.

5. Kesimpulan

Dari penelitian yang telah dilakukan, maka dapat diambil kesimpulan antara lain Implementasi HTB dapat mengontrol *delay* dan *throughput* dari setiap client yang ada di jaringan. Hal ini terbukti pada hasil penerapan di mana kapasitas *bandwidth* 1Mbps dapat kita bagi-bagi sesuai keinginan kita, misalnya 256Kbps untuk *streaming*, 256Kbps untuk *game online*, 256Kbps untuk *browsing*, dan 256Kbps untuk *download*. Pada jaringan dengan metode HTB (*Hierarchical Token Bucket*), penggunaan *bandwidth* pada satu *client* tidak akan mempengaruhi *response time* pada *client* lainnya dalam satu jaringan. Hal ini dapat dibuktikan pada hasil analisa di atas, bahwa ketika *client1* melakukan aktifitas *download*, maka tidak akan mengganggu aktifitas *client* yang lainnya .

Daftar Pustaka

- [1] Behrouz A Fourouzan. 2007. "Data Communication and Networking, Fourth Edition". McGraw-Hill.
- [2] Sukmaaji, Rianto Anjik. 2008. "Jaringan Komputer : Konsep Dasar Pengembangan Jaringan Dan Keamanan Jaringan". Yogyakarta.
- [3] Nugroho Bunafit. 2005. "Instalasi & Kunfigurasi Jaringan Windows & Linux ". Yogyakarta.
- [4] Anggarwal Harshvardhan. 2012. "Managing Quality of Service in Computer Networks". ISSN Volume 3, Issue 11.
- [5] Devera, M. 2003. "Hierarchical Token Bucket Theory". Cluj-Napoca, Romania.
- [6] Riadi Imam, Wicaksono Wahyu. 2011. "Implementasi Quality of Service Menggunakan Metode Hierarchical Token Bucket". ISSN JUSI Vol. 1, No. 2
- [7] Sukmaaji, Rianto Anjik. 2008. "Jaringan Komputer : Konsep Dasar Pengembangan Jaringan Dan Keamanan Jaringan". Yogyakarta..