

IMPLEMENTASI QUALITY OF SERVICE DENGAN METODE HTB (HIERARCHICAL TOKEN BUCKET) PADA PT.KOMUNIKA LIMA DUABELAS

Yunus Arifin

Program Studi Teknik Informatika, Jurusan Ilmu Komputer,
Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Udayana.
Email: yunus.arifin@cs.unud.ac.id

ABSTRAK

Penggunaan *bandwidth* di sebuah jaringan seringkali kurang dimanfaatkan secara optimal. Sering juga kita menemukan pengguna yang tidak di ketahui karena tidak adanya sistem yang mengatur itu sehingga membuat sembarang orang dapat masuk dan menggunakan *bandwidth* pada suatu tempat dengan seenaknya dan merugikan pihak tertentu. *Quality of Service* (QoS) bukan membatasi tetapi lebih kepada menjaga kualitas *bandwidth*, tanpa adanya *Quality of Service* dalam sebuah Jaringan Intranet mengakibatkan ketidaksinambungan *bandwidth* yang diterima *client*.

Berdasarkan hasil penelitian yang telah dilakukan, Setiap paket memperoleh *bandwidth* minimal pada CIR (*Committed Information Rates*) dan lebih dari CIR tetapi tidak melebihi MIR (*Maximum Information Rates*), Selama *traffic* pada parentnya tidak penuh, serta pemerataan *bandwidth* sesuai prioritasnya saat kondisi *traffic* seluruh paket penuh. *Hierarchical Token Bucket* (HTB) mampu memaksimalkan *bandwidth* yang tidak terpakai, sehingga kualitas pelayanan menjadi lebih meningkat.

Kata Kunci : *Quality of Service, HTB, Bandwidth management*

The use of bandwidth in a network is often not optimally utilized. Often we find unknown user in the absence of a system to regulate it that as to make just anyone can log in and use the bandwidth at a place with a loose and harm certain parties. Quality of Service (QoS) is not limiting still rather to maintain the quality of bandwidth, without Quality of Service in network, bandwidth resulting in discontinuity received client. Based on the research that has been done, each packet gain bandwidth of least the CIR (Committed Information Rates) and more than the CIR but not exceeding MIR (Maximum Information Rates), During the traffic on parentnya not full, and equal bandwidth according to traffic conditions throughout his priority packets full. Hierarchical Token Bucket (HTB) is able to maximize the unused bandwidth, thus enhancement the quality of service to be better.

Keywords : *Quality of Service, HTB, Bandwidth management*

PENDAHULUAN

Dewasa ini Kebutuhan akan akses internet sangat tinggi, baik untuk mencari informasi, artikel, pengetahuan terbaru atau hanya untuk chatting. Mikrotik RouterOS merupakan sistem operasi yang mampu membuat komputer menjadi router network atau sering disebut PC router. Pada saat ini sudah terdapat routerboard mikrotik sebagai device yang langsung dapat digunakan tanpa kita harus menginstall mikrotik RouterOS lagi. Sistem

operasi tersebut mencakup berbagai fitur lengkap untuk wireline dan wireless, salah satunya adalah manajemen bandwidth.

Dengan menawarkan produk murah dan berkualitas, berarti pihak penyedia layanan internet harus dapat menyesuaikan bandwidth dengan harga yang relative murah tetapi tetap memperhatikan kualitas, sehingga pelanggan mendapatkan kepuasan berinternet.

Untuk mengatur bandwidth, window server mempunyai *terminal service* seperti *Thinprint*, linux mempunyai CBQ (*Class*

based Queueing) dan HTB (*Hierarchical Token Bucket*). PT. Komunika Lima dua belas menggunakan Server berbasis Mikrotik RouterOS, yang mempunyai keanekaragaman teknik manajemen bandwidth seperti *simple Queue*, HTB (*Hierarchical Token Bucket*), FIFO (*Packet First In First Out*) dan BFIFO (*Bytes First In First Out*), RED (*Random Early Drop*), SFQ (*Stochastic Fairness Queuin*) dan PCQ (*Per Connection Queue*).

Penelitian ini mengimplementasikan Quality of Service, dimana Cyber Akses (PT.Komunika Lima Duabelas) menyediakan paket berlangganan unlimited personal, Paket Office, Paket Bisnis dan juga voucher-voucher hotspot prepaid. Disinilah penelitian terhadap QoS menjadi hal yang vital dimana pihak pelayanan sangat mengharapkan bagaimana mengoptimalkan bandwidth yang tidak terpakai dengan tetap menjaga keseimbangan antara paket-paket bandwidth yang ditawarkan.

TINJAUAN PUSTAKA

Quality of Service

Quality of Service (disingkat menjadi *QoS*) merupakan mekanisme jaringan yang memungkinkan aplikasi-aplikasi atau layanan dapat beroperasi sesuai dengan yang diharapkan (Bunafit, 2005).

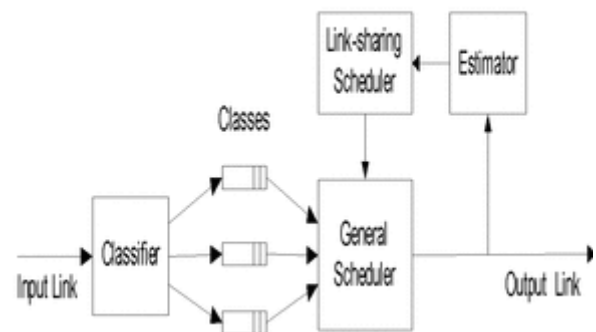
Kinerja jaringan komputer dapat bervariasi akibat beberapa masalah, seperti halnya masalah *bandwidth*, *latency* dan *jitter*, yang dapat membuat efek yang cukup besar bagi banyak aplikasi. Sebagai contoh, komunikasi suara (seperti *VoIP* atau *IP Telephony*) serta *video streaming* dapat membuat pengguna frustrasi ketika paket data aplikasi tersebut dialirkan di atas jaringan dengan bandwidth yang tidak cukup, dengan latency yang tidak dapat diprediksi, atau jitter yang berlebih. Fitur *Quality of Service (QoS)* ini dapat menjadikan *bandwidth*, *latency*, dan *jitter* dapat diprediksi dan dicocokkan dengan kebutuhan aplikasi yang digunakan di dalam jaringan tersebut.

Hierarchical Token Bucket

Hierarchical Token Bucket (HTB) merupakan teknik penjadwalan paket yang sering digunakan bagi router-router berbasis

Linux, dikembangkan pertama kali oleh Martin Devera (Bunafit, 2005).

Cara Kerja HTB tidak ada perbedaan dengan pendahulunya yaitu CBQ, hanya saja pada *General Scheduler* HTB menggunakan mekanisme Deficit Round Robin (DRR) dan pada blok umpan baliknya, *Estimator* HTB tidak menggunakan *Eksponential Weighted Moving Average (EWMA)* melainkan *Token Bucket Filter (TBF)*.



Gambar 1. Deficit Round Robin

Pada HTB terdapat parameter *ceil* sehingga kelas akan selalu mendapatkan *bandwidth* di antara base link dan nilai *ceil* linknya. Parameter ini dapat dianggap sebagai *Estimator* kedua, sehingga setiap kelas dapat meminjam *bandwidth* selama *bandwidth* total yang diperoleh memiliki nilai di bawah nilai *ceil*. Hal ini mudah diimplementasikan dengan cara tidak mengizinkan proses peminjaman *bandwidth* pada saat kelas telah melampaui link ini (keduanya *leaves* dan interior dapat memiliki *ceil*). Sebagai catatan, apabila nilai *ceil* sama dengan nilai base link, maka akan memiliki fungsi yang sama seperti parameter *bounded* pada CBQ, di mana kelas-kelas tidak diijinkan untuk meminjam *bandwidth*. Sedangkan jika nilai *ceil* diset tak terbatas atau dengan nilai yang lebih tinggi seperti kecepatan link yang dimiliki, maka akan didapat fungsi yang sama seperti kelas *non-bounded*.

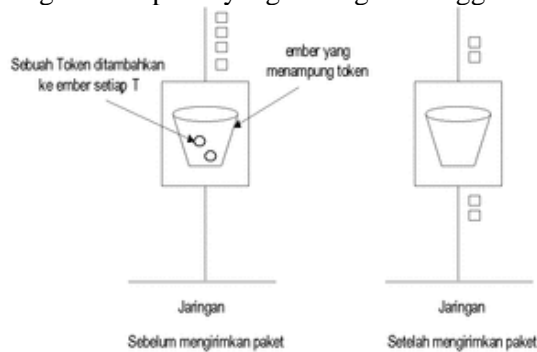
General Scheduler HTB

HTB menganggap hirarki kelas lengkap dan trafik dipisah-pisah menjadi

beberapa aliran trafik, algoritma untuk penjadwalan paket adalah sebagai berikut: pertama memilih kelas pada cabang terendah (*leaf class*) yang *link*-nya belum mencapai batas kemudian mulai mengirimkan paket dari kelas yang memiliki prioritas tertinggi kemudian berlanjut ke yang rendah, apabila *link* semua kelas melampaui batas *link* maka dilakukan suatu test melalui suatu putaran lengkap untuk menemukan *leaf class* yang dapat meminjam *bandwidth* dari kelas di atasnya (*parent class*) jika tidak ada maka putaran diulangi dengan mencoba meminjam *bandwidth* dari kelas di atas *parent class* (*grandfather class*).

Estimator

Hierarchical Token Bucket (HTB) menggunakan Token Bucket Filter (TBF) sebagai *Estimator* untuk menentukan apakah suatu kelas/prioritas berada dalam keadaan *underlimit*, *atlimit* atau *overlimit*. TBF bekerja dengan dasar algoritma ember token, setiap paket yang akan dikirimkan harus memiliki token yang berada dalam ember token, jika token tak tersedia didalam ember maka paket-paket yang akan dikirimkan harus menunggu sampai tersedia token yang cukup untuk mengirimkan paket yang sedang menunggu.



Gambar 2. Token Bucket Filter

Implementasi TBF terdiri dari sebuah *buffer* (*bucket*), yang secara konstan diisi oleh beberapa informasi virtual yang dinamakan token, pada *link* yang spesifik (*token link*). Parameter paling penting dari *bucket* adalah ukurannya, yaitu banyaknya token yang dapat disimpan. Setiap token yang masuk mengumpulkan satu paket yang datang dari antrian data dan kemudian dihapus dari

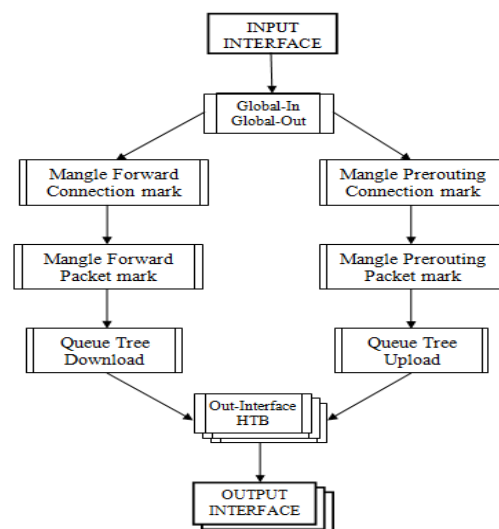
bucket. Dengan menghubungkan algoritma ini dengan dua aliran-token dan data, akan didapatkan tiga buah kemungkinan skenario:

Data yang datang pada TBF memiliki *link* yang sama dengan masuknya token. Dalam hal ini, setiap paket yang masuk memiliki tokennya masing-masing dan akan melewati antrian tanpa adanya *delay*.

Data yang datang pada TBF memiliki *link* yang lebih kecil daripada *link* token. Hanya sebagian token yang dihapus pada output pada tiap paket data yang dikirim ke antrian, dan token akan menumpuk, memenuhi ukuran bucket. Token yang tidak digunakan kemudian akan dapat digunakan untuk mengirimkan data pada kecepatan yang melampaui *link* token standar, ini terjadi jika ada ledakan data yang pendek.

Data yang datang pada TBF memiliki *link* yang lebih besar daripada *link* token. Hal ini berarti bucket akan segera kosong dari token, yang menyebabkan TBF akan menutup alirannya untuk sementara. Hal inilah yang dinamakan situasi *overlimit*. Jika paket-paket tetap datang, maka paket-paket akan segera dibuang.

Implementasi Hierarchical Tokken Bucket (HTB)



Gambar 3. Diagram Alur Hierarchical Token Bucket (HTB)

Berikut merupakan paket bandwidth yang ditawarkan kepada client.

Paket Bandwidth pada PT. Komunika Lima Duabelas yang ditawarkan kepada *customer*

No	Jenis Paket	Priority	Paket Bandwidth	
			Minimal	Maksimal
1	Bisnis	1	1024 kbps	2048 kbps
2	Office	2	512 kbps	1024 kbps
3	Personal	3	256 kbps	512 kbps

Tabel 1. Paket download bandwidth

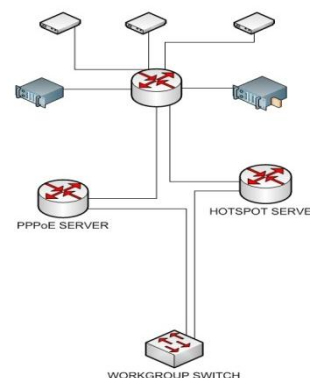
Paket upload bandwidth pada yang akan diberikan kepada *customer*

No	Jenis Paket	Priority	Paket upload Bandwidth	
			Minimal	Maksimal
1	Bisnis	1	256 kbps	357 kbps
2	Office	2	64 kbps	256 kbps
3	Personal	3	32 kbps	64 kbps

Tabel 2. Paket upload bandwidth

HASIL DAN PEMBAHASAN

Topologi server pada PT Komunika Lima Duabelas



Gambar 4. Topologi server pada PT Komunika Lima Duabelas

Implementasi HTB dapat diterapkan dengan langkah-langkah sebagai berikut :

1. Menandai paket-paket untuk tiap koneksi *client* melalui konfigurasi *Mangle*.
2. Menginput CIR, MIR, *Parent* dan prioritas dari tiap paket melalui konfigurasi *Queue*.

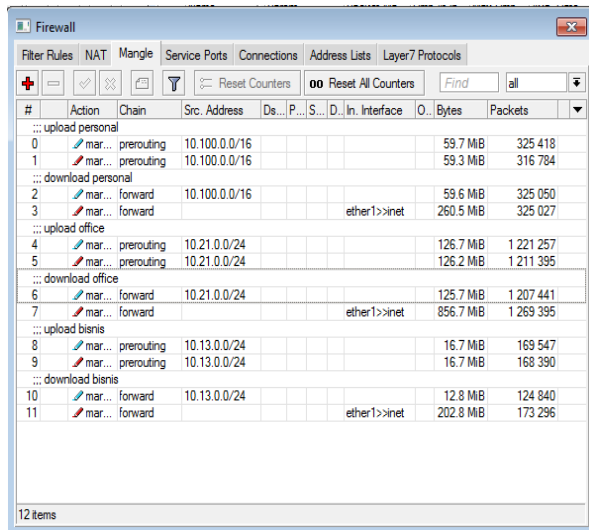
Artinya, setiap ada permintaan *bandwidth* dari *Client* apakah itu upload ataupun download, nantinya akan ditandai secara otomatis oleh *Mangle* lalu akan mendapatkan batasan bandwidth (baik batas atas maupun batas bawah) secara otomatis oleh *Queue*. Siklus penandaan dan pembatasan ini akan berjalan terus menerus selama permintaan akan *bandwidth* terus dilakukan *client* tersebut.

Mangle

Menandai paket dan koneksi lewat *mangle* sehingga paket-paket dan koneksi-koneksi tersebut nantinya dapat diteruskan dan dijabarkan sebagai koneksi paket bisnis, office ataupun paket personal.

Dari konfigurasi tiap-tiap paket untuk *customer* baik upload maupun download memiliki filter untuk menandai koneksi, pertama dengan menandai tiap *connection* yang lewat dan kemudian menandai *packet* yang melewati *mangle* tersebut.

Berikut tampilan seluruh *mangle* untuk paket-paket *customer* yang telah dibuat :

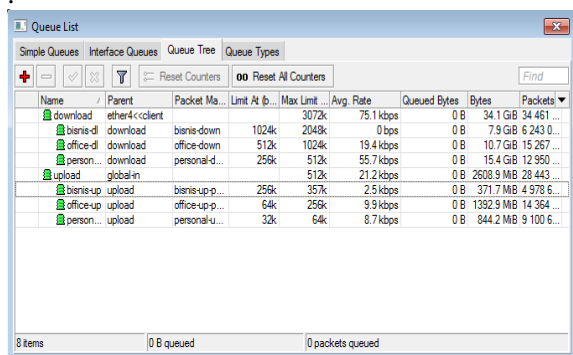


Gambar 5. Mangle view via winbox

Queue

Queue dapat diterapkan setelah mangle menandai seluruh packet pada tiap koneksi (baik download maupun upload) pada tiap packet customer. Hierarchiecal Token Bucket (HTB) mengatur bandwidth dengan parameter parent (interface utama untuk menentukan bandwidth download ataupun upload), packet-mark (mark-packet yang ditentukan pada konfigurasi mangle) dan max-limit (yang merupakan batas kecepatan maksimum) atau dikenal juga dengan MIR (Maximum Information Rate).

Secara keseluruhan hasil dari Queue yang telah dibuat dapat dilihat sebagai berikut :



Gambar 6. Queue tree view via winbox

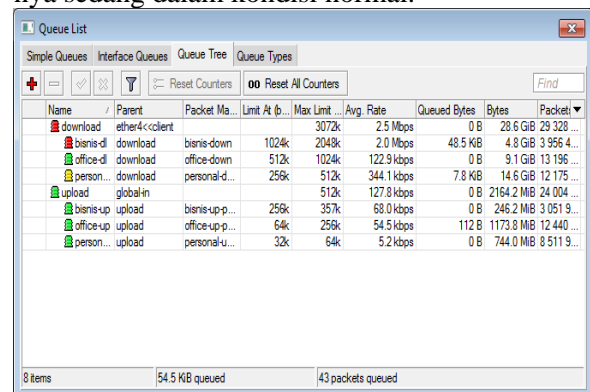
Hasil

Setelah seluruh konfigurasi Mangle dan konfigurasi Queue baik download maupun upload dibuat, maka Hierarchical Token Bucket (HTB) dapat berjalan sesuai yang diharapkan. Berikut beberapa screenshot hasil dari seluruh konfigurasi Hierarchical

Token Bucket (HTB) pada PT. Komunika lima duabelas.

a. Kondisi download maksimal pada paket bisnis

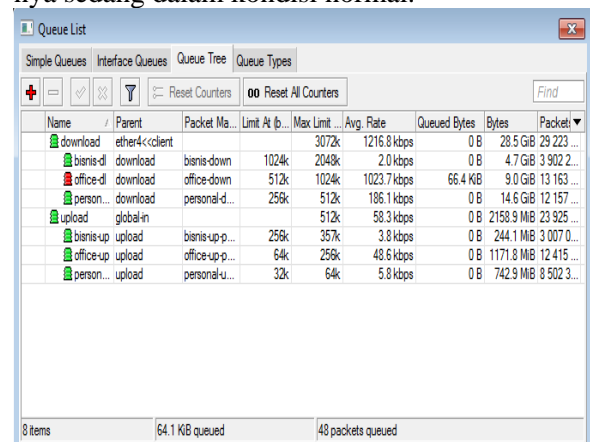
Pada kondisi berikut ini, client dengan paket bisnis melakukan download sampai batas maksimum yaitu 2048 kbps, client mendapatkan bandwidth maksimalnya sebesar 2048 kbps dari bandwidth yang ditawarkan sebesar 1024 kbps. Client mendapatkan bandwidth maksimal dikarenakan pada paket lain, yaitu office dan personal kondisi traffic nya sedang dalam kondisi normal.



Gambar 7. Queue paket bisnis full

b. Kondisi download maksimal pada paket office

Pada kondisi berikut ini, client dengan paket office melakukan download sampai batas maksimum yaitu 1024 kbps, client mendapatkan bandwidth maksimalnya sebesar 1024 kbps dari bandwidth yang ditawarkan sebesar 512 kbps. Client mendapatkan bandwidth maksimal dikarenakan pada paket lain, yaitu bisnis dan personal kondisi traffic nya sedang dalam kondisi normal.



Gambar 8. Queue paket office full

c. Kondisi download maksimal pada paket personal

Pada kondisi berikut ini, *client* dengan paket personal melakukan download sampai batas maksimum yaitu 512 kbps, *client* mendapatkan *bandwidth* maksimal (512 kbps) dikarenakan pada paket lain, yaitu office dan bisnis kondisi *traffic* nya sedang dalam kondisi normal.

Name	Parent	Packet Ma...	Limit At (b...	Max Limit	Avg. Rate	Queued Bytes	Bytes	Packets
download	ether4<<client		3072k	3072k	590.4 kbps	0 B	28.4 GiB	29 170 ...
bisnis-dl	download	bisnis-down	1024k	2048k	392 bps	0 B	4.7 GiB	3 900 8...
office-dl	download	office-down	512k	1024k	152.9 kbps	0 B	9.0 GiB	13 130 ...
person...dl	download	personal-d...	256k	512k	427.9 kbps	0 B	14.6 GiB	12 138 ...
upload	global-in		512k	46.3 kbps	0 B	2153.4 MiB	23 885 ...	
bisnis-up	upload	bisnis-up-p...	256k	357k	104 bps	0 B	243.9 MiB	3 005 4...
office-up	upload	office-up-p...	64k	256k	7.5 kbps	0 B	1169.7 MiB	12 392 ...
person...up	upload	personal-u...	32k	64k	38.7 kbps	0 B	739.7 MiB	8 497 4...

Gambar 9. Queue paket personal full

d. Kondisi download maksimal pada dua paket berbeda

Pada hasil pengujian berikut, dua paket berbeda mendapatkan *bandwidth* maksimalnya tanpa mempengaruhi paket lain yang *traffic* nya sedang dalam kondisi normal.

Name	Parent	Packet Ma...	Limit At (b...	Max Limit	Avg. Rate	Queued Bytes	Bytes	Packets
download	ether4<<client		3072k	3072k	2.8 Mbps	0 B	28.7 GiB	29 429 ...
bisnis-dl	download	bisnis-down	1024k	2048k	1682.2 kbps	0 B	4.9 GiB	4 013 0...
office-dl	download	office-down	512k	1024k	996.4 kbps	10.9 KiB	9.1 GiB	13 224 ...
person...dl	download	personal-d...	256k	512k	200.0 kbps	0 B	14.6 GiB	12 189 ...
upload	global-in		512k	64.3 kbps	0 B	2163.6 MiB	24 085 ...	
bisnis-up	upload	bisnis-up-p...	256k	357k	50.7 kbps	0 B	248.9 MiB	3 101 4...
office-up	upload	office-up-p...	64k	256k	28.0 kbps	0 B	1175.9 MiB	12 464 ...
person...up	upload	personal-u...	32k	64k	5.5 kbps	0 B	744.8 MiB	8 519 6...

Gambar 10. Queue paket bisnis dan office full

Dari gambar 4.35 di atas, seluruh *traffic* download sebesar 2,8 Mbps hampir memenuhi batas maksimal seluruh download paket yaitu 3 Mbps. Bisnis ada pada *traffic* sebesar 1682 kbps dan office sebesar 996 kbps, sama-sama mendapatkan *bandwidth* melebihi dari batas bawah *bandwidth* yang ditawarkan, dikarenakan pada paket personal *traffic* nya hanya sebesar 200 kbps. Berikut hasil pengujian lainnya, dimana terdapat

kondisi dua paket dengan *traffic* padat (hampir memenuhi batas maksimalnya) dan satu paket lain pada *traffic* normal;

Name	Parent	Packet Ma...	Limit At (b...	Max Limit	Avg. Rate	Queued Bytes	Bytes	Packets
download	ether4<<client		3072k	3072k	2.7 Mbps	0 B	28.9 GiB	29 449 ...
bisnis-dl	download	bisnis-down	1024k	2048k	1538.1 kbps	3004 B	4.9 GiB	4 024 4...
office-dl	download	office-down	512k	1024k	670.7 kbps	0 B	9.1 GiB	13 232 ...
person...dl	download	personal-d...	256k	512k	491.0 kbps	0 B	14.6 GiB	12 192 ...
upload	global-in		512k	91.4 kbps	0 B	2170.7 MiB	24 103 ...	
bisnis-up	upload	bisnis-up-p...	256k	357k	46.6 kbps	0 B	249.5 MiB	3 111 7...
office-up	upload	office-up-p...	64k	256k	35.2 kbps	0 B	1176.3 MiB	12 470 ...
person...up	upload	personal-u...	32k	64k	9.5 kbps	0 B	744.9 MiB	8 521 3...

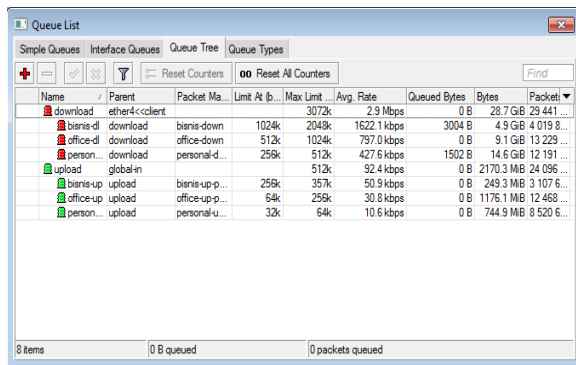
Gambar 11. Queue paket bisnis dan personal full

Name	Parent	Packet Ma...	Limit At (b...	Max Limit	Avg. Rate	Queued Bytes	Bytes	Packets
download	ether4<<client		3072k	3072k	2.6 Mbps	0 B	28.9 GiB	29 535 ...
bisnis-dl	download	bisnis-down	1024k	2048k	1090.8 kbps	0 B	5.0 GiB	4 062 3...
office-dl	download	office-down	512k	1024k	1006.8 kbps	0 B	9.1 GiB	13 266 ...
person...dl	download	personal-d...	256k	512k	575.4 kbps	35.5 KiB	14.7 GiB	12 206 ...
upload	global-in		512k	71.5 kbps	0 B	2174.5 MiB	24 174 ...	
bisnis-up	upload	bisnis-up-p...	256k	357k	48.4 kbps	0 B	251.3 MiB	3 146 1...
office-up	upload	office-up-p...	64k	256k	26.2 kbps	0 B	1177.7 MiB	12 498 ...
person...up	upload	personal-u...	32k	64k	9.9 kbps	0 B	745.3 MiB	8 529 7...

Gambar 12. Queue paket office dan personal full

e. Kondisi download maksimal pada seluruh paket

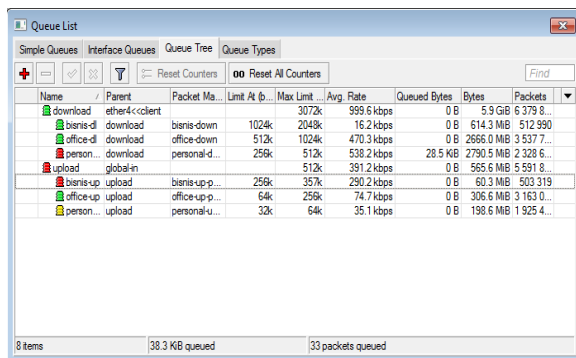
Pada hasil pengujian ini, *traffic bandwidth* pada seluruh paket *customer* sedang dalam kondisi penuh. Paket bisnis yang memiliki batas maksimal *bandwidth* sebesar 2048 kbps mendapatkan *bandwidth* sebesar 1622 kbps, paket office yang memiliki batas maksimal *bandwidth* sebesar 1024 kbps mendapatkan *bandwidth* sebesar 797 kbps, sedangkan paket personal yang mendapatkan batas maksimal *bandwidth* sebesar 512 kbps mendapatkan *bandwidth* sebesar 427 kbps.



Gambar 13. Queue seluruh paket full

f. Kondisi upload

Dalam hasil penelitian *queue* upload, tidak banyak uji coba yang bisa dilakukan, karena keterbatasan media upload dan karena minimnya *traffic* upload yang dilakukan *customer*. Dengan teknik *Hierarchical Token Bucket* (HTB) yang sama seperti pada download, tidak ada perbedaan jauh mekanisme pengoptimalan *bandwidth* nya. Berikut *screenshot* pada paket bisnis yang melakukan upload hampir pada batas maksimumnya, yaitu sebesar 290 kbps, pada paket office sebesar 74 kbps dan pada paket personal sebesar 35 kbps;



Gambar 14. Queue upload

Kesimpulan

Quality of Service (QoS) bukan membatasi tetapi lebih kepada menjaga kualitas *bandwidth*, tanpa adanya *Quality of Service* dalam sebuah Jaringan Intranet mengakibatkan ketidaksinambungan *bandwidth* yang diterima *client*.

Hierarchical Token Bucket (HTB) merupakan teknik QoS yang mampu memaksimalkan *bandwidth* yang tidak

terpakai, sehingga kualitas pelayanan menjadi lebih meningkat, berikut hasil yang dicapai;

- Setiap paket memperoleh *bandwidth* minimal pada CIR (*Committed Information Rates*).
- Setiap paket dapat memperoleh *bandwidth* lebih dari CIR tetapi tidak melebihi MIR (*Maximum Information Rates*), Selama *traffic* pada parentnya tidak penuh.
- Terjadi pemerataan *bandwidth* sesuai prioritasnya saat kondisi *traffic* seluruh paket penuh.

DAFTAR PUSTAKA

Handriyanto, D. F. (2009). Kajian Penggunaan Mikrotik Router Os™ Sebagai Router Pada Jaringan Komputer.

Bunafit Nugroho. 2005. “Instalasi & Konfigurasi Jaringan Windows & Linux”. Yogyakarta. Andi Yogyakarta.

Kustanto & Saputro Daniel T. 2008. “Membangun Server Internet dengan MikroTik OS”. Yogyakarta. Gava Media.

Forouzan, A. (2007). *Data Communications And Networking 4th Edition*. New York: McGraw-Hill.

Stallings, William. 2001 *Data & Computer Communications* (terjemahan). Jakarta: Salemba Teknika

Balan, Doru G, Alin P(2009). *Extended Linux HTB Queuing Discipline Implementations*. International Journal Of Information Studies, University Of Suceava, Romania.