

PERANCANGAN SISTEM OPTIMALISASI MANAJEMEN *BANDWIDTH* DENGAN HTB (HIERARCHICAL TOKEN BUCKET) MENGGUNAKAN LINUX

Muhammad Iqbal

Program Studi Teknik Komputer, AMIK Mitra Gama Duri

Jl. Hangtuh No. 99 , Duri 28884

e-mail: iqbal.kun@gmail.com, muhammad.iqbal@amikmitragama.ac.id

Abstrak

Manajemen *bandwidth* menjadi hal yang mutlak diperlukan bagi jaringan multi layanan, semakin banyak dan bervariasinya aplikasi yang dapat dilayani oleh suatu jaringan berpengaruh pada penggunaan jaringan tersebut. Manajemen *bandwidth* memegang peranan penting dalam mengatur jenis aplikasi yang bisa mengakses jaringan yang ada selain itu manajemen *bandwidth* mampu memberikan garansi kepada aplikasi yang mendapat alokasi *bandwidth* untuk terus mengirimkan data sesuai dengan alokasinya sekalipun terjadi kemacetan dalam jaringan bahkan dalam keadaan tertentu. *Hierarchical Token Bucket* (HTB) sebagai implementator manajemen *bandwidth* yang tersedia secara gratis dan dapat dijalankan diatas platform sistem Operasi LINUX merupakan *Bandwidth Manager* yang layak dianalisa keunggulan dan kelemahannya, diharapkan penggunaannya yang tepat dan akurat akan membuat jaringan yang menerapkan *Bandwidth Manager* ini bekerja secara optimal. Dari penerapan yang dilakukan HTB untuk *bandwidth management* telah menghasilkan hasil yang optimal sesuai dengan ketentuan, sehingga manajemen *bandwidth* dengan HTB layak digunakan.

Kata Kunci : Jaringan, manajemen bandwidth, Linux, HTB

Abstract

Bandwidth management becomes absolutely necessary for multi-service networks, the more and varied applications that can be served by a network effect on the use of the network. Bandwidth management plays an important role in regulating the types of applications that can access the network that is in addition to bandwidth management that is able to provide guarantees to the application that received the allocation of bandwidth to continue to transmit data in accordance with the allocation even if there is congestion in the network even under certain circumstances. Hierarchical Token Bucket (HTB) as implementer bandwidth management which is available for free and can run on Linux Operating system platforms Bandwidth Manager is a viable analyzed their advantages and disadvantages, are expected to use appropriate and accurate will create a network that implements Bandwidth Manager is working optimally. Conducted from the HTB application for bandwidth management has yielded optimal results in accordance with the provisions, so that the bandwidth management with HTB fit for use and works optimally.

Keywords: Network, bandwidth management, Linux, HTB

PENDAHULUAN

Manajemen *bandwidth* merupakan salah satu cara untuk mengelola koneksi internet bagi perusahaan, institusi maupun sekolah agar koneksi internet yang terbatas bisa dinikmati oleh banyak orang walaupun *bandwidth* yang disewa tidak terlalu besar. Institusi yang menggunakan jaringan komputer dan internet sebagai pendukung pekerjaan akan memerlukan pengelolaan *bandwidth*, terutama jika jaringan tersebut mempunyai beberapa divisi, dimana tiap divisi belum tentu memerlukan koneksi internet dengan koneksi cepat. Maka dengan pengelolaan *bandwidth* akan memberikan keuntungan bagi institusi.

Ada beberapa metode dalam

melakukan manajemen *bandwidth* diantaranya adalah teknik *queueing* (**antrian**) yaitu dengan mengklasifikasikan paket, selanjutnya menempatkan paket pada antriannya dan kemudian dilakukan penjadwalan pengiriman. Pada metode ini terdapat banyak algoritma yang dapat digunakan untuk mengatur trafik misalnya CBQ, HTB, PRIO.

Teknik antrian *Hierarchy Token Bucket* (HTB) mirip dengan CBQ hanya perbedaannya terletak pada opsi, HTB lebih sedikit opsi saat konfigurasi serta lebih presisi. Teknik antrian HTB memberikan kita fasilitas pembatasan trafik 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.

HTB merupakan teknik yang menggunakan *Classful Queueing Disciplines* yang memperbolehkan paket diarahkan ke *class* tertentu dan antrian *subqueues*. Sebaik apa HTB dalam melakukan pengelolaan *bandwidth* masih perlu diketahui lebih lanjut.

KAJIAN LITERATUR

Dalam optimalisasi *bandwidth* ada beberapa komponen yang harus diketahui :

1. *Quality of Service*
2. *Manajemen Bandwidth*
3. *Classful Queueing Discipline*
4. HTB

Quality of Service

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. (Yunus Arifin, 2012)

Komponen-komponen dari QoS adalah:

Throughput adalah kecepatan rate (transfer data efektif) dalam bps. Throughput juga berarti banyak bit yang berhasil dikirimkan sampai ketertujuan dalam satu selang waktu pengamatan tertentu.

$$\text{Throughput} = \frac{\text{Ukuran file yang diterima (bit)}}{\text{End Time} - \text{Start Time}}$$

Delay, merupakan total waktu yang dilalui suatu paket dari pengirim ke penerima melalui jaringan. Delay dari pengirim ke penerima pada dasarnya tersusun atas hardware latency, delay akses, dan delay transmisi.

$$\text{Delay} = \frac{\text{Jumlah total (Waktu terima-waktu kirim)}}{\text{Jumlah paket yang dibandingkan}}$$

Jitter, merupakan variasi dari delay end-to-end.

Level-level yang tinggi pada *jitter* dalam aplikasi-aplikasi berbasis UDP merupakan situasi yang tidak dapat diterima di mana aplikasi-aplikasinya merupakan aplikasi real-time, seperti sinyal audio dan video.

Bandwidth, merupakan rate transfer data maksimal yang dapat diteruskan antara dua titik.

Packet Loss, *Packet Loss* menunjukkan banyaknya jumlah packet yang hilang. Umumnya perangkat memiliki *buffer* untuk menampung data yang diterima. Jika terjadi *congestion* atau macet pada jaringan maka pada umumnya akan terjadi *loss*.

Manajemen bandwidth

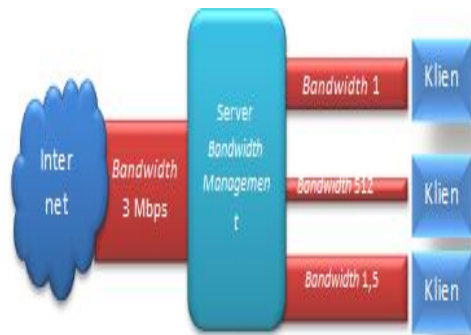
Bandwidth mengacu kepada jumlah informasi yang ditransmisikan pada satu waktu tertentu dalam jalur data. Secara umum, *bandwidth* jaringan merupakan ukuran *bit rate* yang tersedia atau penggunaan data komunikasi yang diekspresikan sebagai bits/detik atau pengaliansya (kilobits/s, megabits/s). (Richmond, et.al, 2012)

Sedangkan menurut Mahanta (2013), *Bandwidth* dalam jaringan komputer mengacu kepada kapasitas data yang melewati koneksi jaringan atau antarmukanya. Salah satu istilah umum *bandwidth* adalah bits per detik. Istilah ini berasal dari bidang teknik elektro, dimana *bandwidth* merupakan total jarak atau berkisar antara tertinggi dan terendah sinyal pada saluran komunikasi.

Bandwidth merepresentasikan kapasitas dari sebuah koneksi. Semakin besar kapasitasnya, semakin besar kemungkinan akan mengikuti kualitas performansinya, walaupun performansinya juga bergantung pada hal-hal yang lain, seperti latensi.

Bandwidth shaping adalah proses mengalokasikan bagian dari koneksi jaringan serta membangun jumlah penggunaan *bandwidth* yang sesuai dengan jenis kegiatan. *Bandwidth* shaping juga mengacu kepada alokasi *bandwidth*, manajemen *bandwidth* dan traffic shaping.

Tanpa adanya adanya manajemen *bandwidth*, sebuah aplikasi maupun seorang pengguna tidak akan bisa mengendalikan semua *bandwidth* yang tersedia dan mencegah aplikasi atau pengguna lainnya dari menggunakan jaringan. Akan sangat menyulitkan untuk membedakan antara beberapa jenis lalu lintas jaringan, dan akan sangat menyulitkan untuk mengendalikan pengguna atau aplikasi mana yang akan mendapatkan prioritas pada jaringan.



Gambar 1. Skema Manajemen Bandwidth

Cara manajemen bandwidth bekerja menggunakan dua fungsi dasar yaitu:

1. Membatasi *bandwidth* untuk transfer data
 Dengan hal ini maka dapat memotong lalu lintas yang tidak produktif yang banyak menggunakan *bandwidth*, seperti mengunduh atau meng-*upload* data yang besar, *streaming* video dan lain-lain.
2. Menyediakan *bandwidth* khusus untuk layanan tertentu
 Hal ini dimungkinkan untuk memesan *bandwidth* yang dikhususkan untuk layanan tertentu (email, IP telepon). *Bandwidth* akan selalu tersedia, terlepas dari beban lalu lintas jaringan yang berjalan.

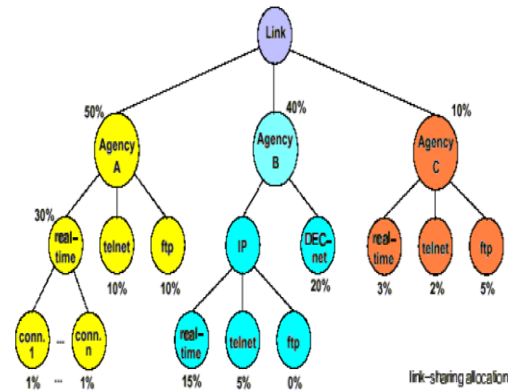
Classful Queueing Discipline

Classful Queueing merupakan suatu disiplin antrian yang akan membagi trafik ke kelas-kelas antrian selanjutnya.

Hierarchical Token Bucket

Hierarchical Token Bucket (HTB) adalah sebuah metode untuk mendapatkan distribusi *bandwidth* diantara beberapa jenis layanan dalam bentuk yang fleksibel. HTB sebenarnya adalah teknik yang paling banyak digunakan pada Linux Traffic Control untuk menyediakan QoS (Munir dkk, 2014).

Hierarchical Token Bucket (HTB) merupakan teknik penjadwalan paket yang baru-baru ini diperkenalkan bagi router berbasis Linux, dikembangkan pertama kali oleh Martin Devera pada akhir 2001 untuk diproyeksikan sebagai pilihan (atau pengganti) mekanism penjadwalan yang saat ini masih banyak dipakai yaitu CBQ (Setiawan, 2013).



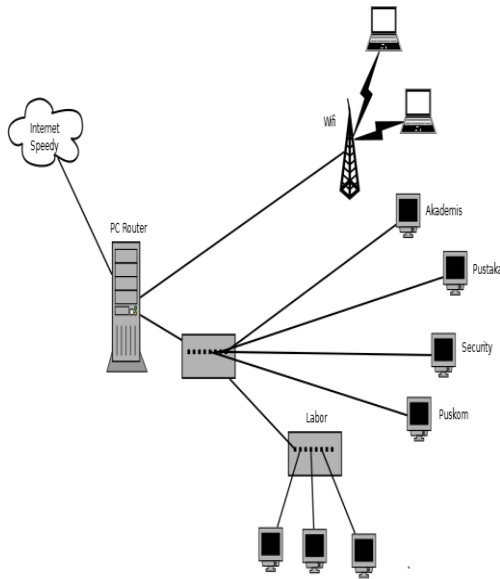
Gambar 2. struktur pembagian link pada HTB

METODE PENELITIAN

Metodologi penelitian ini dilakukan secara sistematis yang dapat digunakan sebagai pedoman untuk peneliti dalam melaksanakan penelitian, agar hasil yang dicapai tidak menyimpang dan tujuan yang diinginkan dapat terlaksana dengan baik dan sesuai dengan tujuan yang telah ditetapkan sebelumnya. Kerangka kerja dalam melakukan penelitian ini adalah mendeskripsikan masalah, analisa masalah, menerapkan metode, implementasi dan pengujian.

Analisa Perancangan

Bandwidth management adalah metode yang digunakan untuk mengelola *bandwidth* pada jaringan komputer. HTB merupakan salah satu dari teknik *management bandwidth* yang banyak digunakan, dalam hal ini HTB digunakan oleh penulis untuk memaksimalkan koneksi internet yang tersedia. Pada institusi tempat penulis melakukan penelitian yaitu AMIK Mitragama menggunakan beberapa komputer dan beberapa labor komputer, labor komputer tidak secara langsung terkoneksi ke internet, sedang untuk komputer lain di akademis, pustaka dan ruangan lainnya terkoneksi secara langsung ke internet. Karena tidak semua ruangan/divisi memerlukan koneksi internet yang cepat, maka pengelolaan *bandwidth management* merupakan hal terbaik yang memungkinkan untuk dilakukan agar pekerjaan yang memerlukan koneksi internet bisa disesuaikan.



Gambar 3 Topologi Jaringan yang Digunakan

Skenario yang digunakan adalah dengan menguji sebanyak 10 klien, setiap klien diberikan alokasi bandwidth yang berbeda-beda. Pengujian dilakukan dengan menggunakan *virtual machine*, klien-klien yang digunakan bisa dilihat pada tabel 1.

Tabel 1 Alokasi bandwidth

No	Alokasi Bandwidth	IP Address
1	64 Kbps	192.20.20.2
2	256 Kbps	192.20.20.3
3	128 Kbps	192.20.20.4
4	128 Kbps	192.20.20.5
5	64 Kbps	192.20.20.6
6	64 Kbps	192.20.20.7
7	512 Kbps	192.20.20.8
8	512 Kbps	192.20.20.9
9	1024 Kbps	192.20.20.10
10	32 Kbps	192.20.20.11

HASIL DAN PEMBAHASAN

Hasil implementasi dengan proses unduh ke tiap klien sudah sesuai dengan yang diinginkan, hasilnya sudah optimal dikarena saat sebelum menggunakan dan setelah menggunakan HTB menghasilkan data yang berbeda. Dimana pada klien 1 sebelum menggunakan HTB mendapatkan kecepatan unduh sebesar 145 KBps dan setelah menggunakan mendapatkan

8,4 KBps jika dihitung berdasarkan *bandwidth* yang dialokasikan sebesar 256 Kbps maka paling tidak seharusnya maksimal klien 1 mendapatkan kecepatan 32 KBps. Sedang untuk klien 2 dengan alokasi *bandwidth* sebesar 64 Kbps hanya mendapatkan sebesar 7,0 KBps yang juga merupakan hasil yang sudah optimal dikarenakan sudah sesuai jika 64 Kbps dibagi dengan 8 bit maka akan mendapatkan sebesar 8 KBps. Dan hasil 7,0 lebih kecil dari yang seharusnya. Karena itu hasil tersebut untuk klien 1 dan klien 2 sudah merupakan hasil yang terbaik. Begitu juga dengan klien 3 sampai dengan klien 10 telah menghasilkan hasil yang optimal.

Klien	Alokasi	Sebelum	Sesudah
Klien 1	256 Kbps	145 KBps	8 KBps
Klien 2	64 Kbps	18 KBps	7 KBps
Klien 3	128 Kbps	255 KBps	10 KBps
Klien 4	128 Kbps	511 KBps	9 KBps
Klien 5	64 Kbps	122 KBps	6 KBps
Klien 6	64 Kbps	169 KBps	7 KBps
Klien 7	512 Kbps	84 KBps	38 KBps
Klien 8	512 Kbps	66 KBps	36 KBps
Klien 9	1024 Kbps	386 KBps	98 KBps
Klien 10	31 Kbps	372 KBps	3 KBps

SIMPULAN

Setelah melakukan perancangan dan implementasi maka penulis dapat menarik beberapa kesimpulan seperti berikut ini :

1. *Hierarchy Token Bucket* telah mengoptimalkan bandwidth yang dialokasikan kepada klien.
2. Perbedaan pengujian dengan menggunakan iperf saat sebelum dan sesudah menggunakan HTB tampak dengan jelas, bahwa HTB telah berhasil membatasi kemampuan klien dalam menggunakan layanan jaringan internet sesuai dengan bandwidth yang diberikan.
3. Pengujian dengan menggunakan proses unduh telah sesuai dengan ketentuan yang dibuat, dimana sudah sesuai dengan alokasi bandwidth yang diberikan.

REFERENSI

- [1] Yunus Arifin (2012), “Implementasi Quality Of Service Dengan Meotde HTB (Hierarchical Token Bucket) Pada PT. Komunika Lima Dua Belas”, Volume 1, Number 2.
- [2] Kanu Richmond U, Kuyoro Shade O, Ogunlere Samson O, dan Adegbenjo Aderanko A, (2012), “Management and Control of Bandwidth in Computer Networks, Volume 2, Number 3.
- [3] Devajit Mahanta, Majidul Ahmed dan Utpal Jyoti Bora, (2013), “ A Study Bandwidth Management in Computer Networks”, Volume 2, Issue 2.
- [4] Munir Kolapo Yahya-Imam, Sellapan Palaniapan dan Vathsala Devi (2014), “An Enhanced Bandwidth Management Scheme for Improved Quality of Service in Network Communication System”, Volume 2, Number 2.
- [5] Didit Aji Setiawan (2013), “Membangun Prioritasisasi Lalulintas Data (Internet) Menggunakan HTB Queueing Disciplines Pada Jaringan Lokal SMK 1 Nanggulan”.
- [6] Balan, D, G dan Potorac D, A, (2010), “Extended Linux HTB Queueing Discipline Implementations”, Volume 2, issue 2.