

## IMPLEMENTASI PENGATURAN DAN PRIORITAS BANDWIDTH DENGAN HIERARCHICAL TOKEN BUCKET BERBASISKAN GUI PADA LINUX SERVER CLEAROS

Sukmajati Prayoga

Jurusan Teknik Komputer Universitas Komputer Indonesia, Bandung  
yoga.tux@gmail.com

### ABSTRAK

*Hierarchical Token Bucket (HTB) adalah metode yang digunakan untuk mengatur bandwidth pada jaringan dengan mengatur packet yang dikirimkan pada jaringan. Pembagian kelas berdasarkan Priority Service (PS) adalah program atau interface yang dapat di konfigurasi. Untuk dapat melakukan konfigurasi tersebut dilakukan pada sistem operasi Linux dengan menggunakan Command Line Interface (CLI). ClearOS adalah Linux server yang distribusikan berdasarkan CentOS dan Red Hat Enterprise Linux, di desain khusus untuk keperluan gateway dan server pada perusahaan kecil dan menengah dengan menggunakan sistem administrator berbasiskan Graphical User Interface (GUI). Selain itu digunakan aplikasi webHTB sebagai tools untuk mengatur bandwidth dan prioritas langsung pada trafik control (TC) pada kernel Linux ClearOS berbasiskan GUI. Dengan mengimplementasikan metode Hierarchical Token Bucket (HTB) pada sistem operasi ClearOS diharapkan menjadikan jaringan yang menerapkan Bandwidth manajemen sistem ini dapat mengatur bandwidth dan prioritas bandwidth dengan baik. Pengujian dilakukan dengan cara pembatasan bandwidth dan merubah index prioritas dengan beberapa client. Hasil pengujian webHTB terhadap Linux ClearOS menunjukkan bahwa nilai prioritas pada metode Hierarchical Token Bucket (HTB) memegang peranan paling besar untuk kecepatan pada client. Dimana bandwidth menentukan nilai transfer rate minimal.*

*Kata kunci: Bandwidth, HTB, Priority Service, Clear OS*

### 1. PENDAHULUAN

*Hierarchical Token Bucket (HTB) merupakan salah satu Sistem untuk manajemen bandwidth. Bandwidth manajemen sistem berfungsi untuk memaksimalkan sebuah bandwidth dengan mengatur atau membagi bandwidth sehingga penggunaannya menjadi maksimal. Hierarchical Token Bucket (HTB) adalah metode yang digunakan untuk mengatur bandwidth pada jaringan dengan membagi-bagi packet yang berada pada jaringan. Pembagian kelas berdasarkan prioritas, program, atau interface yang dapat di konfigurasi. Untuk dapat melakukan konfigurasi tersebut dilakukan pada sistem operasi Linux dengan menggunakan Command Line Interface (CLI).*

ClearOS adalah Linux server yang distribusikan berdasarkan CentOS dan Red Hat Enterprise Linux, di desain khusus untuk keperluan gateway dan server pada perusahaan kecil dan menengah dengan

menggunakan sistem administrator berbasiskan Graphical User Interface (GUI) sebagai alternatif Windows Small Bussines Server. Dengan mengimplementasikan metode Hierarchical Token Bucket (HTB) pada sistem operasi ClearOS menjadikan jaringan yang menerapkan Bandwidth manajemen sistem ini dapat mengatur bandwidth dan prioritas dengan baik.

Selain itu digunakan aplikasi webHTB sebagai tools untuk mengatur bandwidth dan prioritas langsung pada trafik control (tc) pada kernel Linux ClearOS berbasiskan GUI. Artinya semua konfigurasi pengaturan bandwidth dilakukan melalui sebuah halaman web interface yang diharapkan akan memudahkan seorang administrator jaringan dalam melakukan konfigurasi pengaturan bandwidth terhadap PC router, tanpa harus menguasai perintah-perintah dasar yang ada pada PC router tersebut

# Implementasi Pengaturan Dan Prioritas Bandwidth Htb (Hierarchical Token Bucket) Berbasiskan Gui Pada Linux Server ClearOS

## 2. TINJAUAN PUSTAKA

### Router

*Router* merupakan sebuah *device* atau alat yang dapat menghubungkan dua atau lebih jaringan komputer yang berbeda. Secara umum *router* adalah alat pada suatu jaringan komputer yang bekerja di *network layer* pada lapisan OSI. Dalam *router* ini terdapat *routing table* yaitu tabel yang berisi alamat-alamat jaringan yang dibutuhkan untuk menentukan tujuan dari paket-paket data yang akan dilewatkan pada suatu jaringan tersebut[5].

Untuk membuat *router*, bisa memanfaatkan suatu jenis sistem operasi Windows, Unix, Linux atau jenis sistem operasi lain pada komputer PC dengan hanya menambahkan dua buah *network interface card* (NIC).

Di dalam PC *router* dapat di pasang beberapa aplikasi yang dibutuhkan untuk aplikasi *webHTB* supaya dapat berjalan dengan baik pada sistem. Salah satu diantaranya adalah *Webserver Apache*, *MySQL* dan *Network Address Translation* (NAT).

### Bandwidth

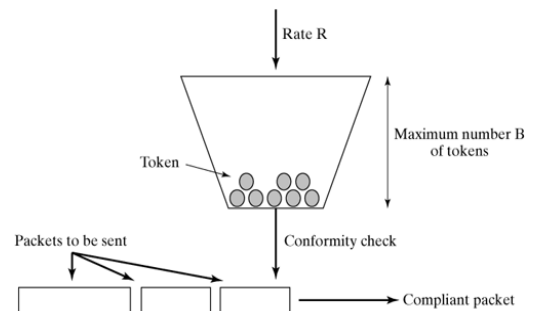
Definisi dari *Bandwidth* adalah banyaknya ukuran suatu data atau informasi yang dapat mengalir dari suatu tempat ke tempat lain dalam sebuah *network* di waktu tertentu. *Bandwidth* dapat dipakai untuk mengukur baik aliran data analog maupun data digital. Sekarang sudah menjadi umum jika kata *bandwidth* lebih banyak dipakai untuk mengukur aliran data digital.

### Prioritas Bandwidth

Prioritas *Bandwidth*, merupakan penyampaian tingkat layanan berdasarkan pada dari arti penting koneksi dan permintaan untuk lalu lintas jaringan sehubungan dengan koneksi yang lain. Selama periode lalu lintas penuh, akan memperlambat prioritas lebih rendah dari aplikasinya dengan menurunkan prioritas aplikasinya sehingga meningkatkan *bandwidth* ke aplikasi prioritas yang lebih tinggi.

### Hierarchical Token Bucket (HTB)

HTB (*Hierarchy Token Bucket*) merupakan salah satu teknik antrian yang memiliki tujuan untuk menerapkan *link sharing*. Dalam konsep *link sharing*, jika suatu kelas meminta kurang dari jumlah *service* yang telah ditetapkan untuknya, sisa *bandwidth* akan di distribusikan ke kelas-kelas lain yang meminta *service*. HTB menggunakan TBF sebagai estimator yang sangat mudah diimplementasikan. Estimator ini hanya menggunakan *rate*, sebagai akibatnya seorang administrator hanya perlu mengeset *rate* yang akan di berikan ke suatu kelas[6].



Gambar 1 Konsep *Token Bucket Filter* (TBF)

Pada HTB ini memiliki parameter *ceil* sehingga kelas akan selalu mendapatkan *bandwidth* di antara *baserate* dan nilai *ceil ratenya*. Parameter ini dianggap sebagai estimator kedua, sehingga setiap kelas dapat meminjam *bandwidth* selama *bandwidth* total yang diperoleh memiliki nilai di bawah nilai *ceil*.

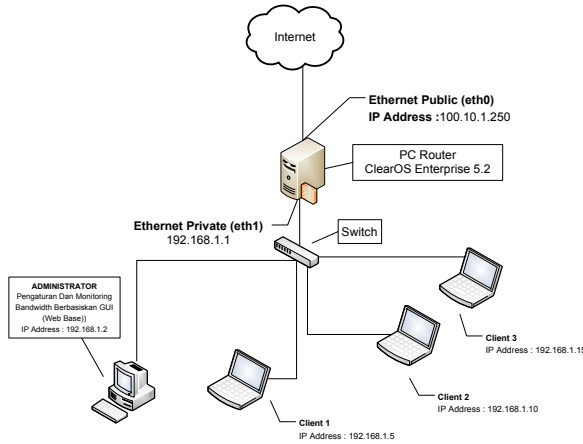
### Linux ClearOS

ClearOS diciptakan sebagai Sistem Operasi untuk Router. Linux ClearOS bersifat bebas untuk versi *Enterprise Edition*, stabil dan konfigurasi dapat dilakukan dengan mudah (*web-based*). ClearOS dikonfigurasi dengan *Linux Command*, ClearOS juga dapat dikonfigurasi lewat *console* (text based). ClearOS tersedia juga dalam versi berbayar dengan beberapa aplikasi tambahan serta *technical support online* dari pengembang. fitur yang dapat dikonfigurasi pada ClearOS adalah *Gateway Service*, *DNS*

Service, Content Filtering, Monitoring System, dan Web Server

### 3. PERANCANGAN SISTEM

Sistem dirancang sedemikian rupa sehingga menjadi PC Router yang mampu mengatur bandwidth ke setiap client dengan reabilitas yang baik, dapat dilihat dari diagram pada gambar 2 :



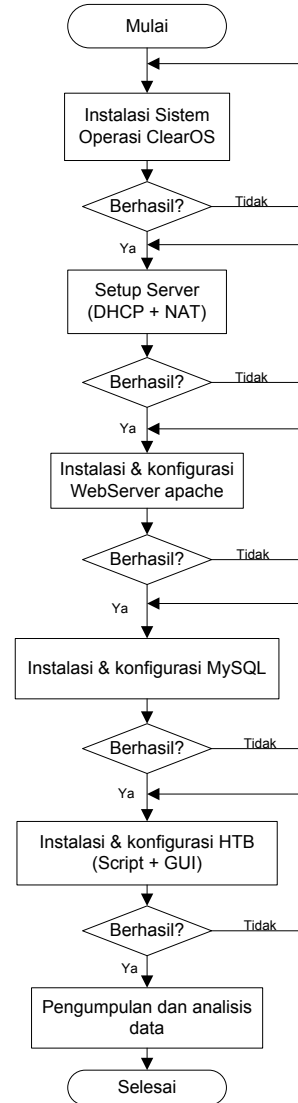
Gambar 2 Topologi Perancangan Jaringan Secara Umum

Dari gambar 2 dapat di jelaskan sebagai berikut :

1. Pada PC router dipasang dua buah ethernet card (ethernet private dan ethernet public).
2. Masing – masing ethernet card tergabung dalam jaringan yang berbeda.
3. Pada eth0 merupakan ethernet public untuk jaringan internet dengan alamat ip 100.10.1.250 .
4. Pada eth1 merupakan ethernet private untuk jaringan local (lan) dengan alamat ip 192.168.1.1 .
5. PC router dan semua perangkat jaringan local (lan) di hubungkan oleh satu konsentrator yaitu switch.
6. Selanjutnya konfigurasi PC router agar memberikan bandwidth ke client 1 ip address 192.168.1.5 dan client 2 dengan ip address 192.168.1.10 serta client 3 ip address 192.168.15.

7. Pengaturan bandwidth dan monitoring dilakukan pada PC administrator yang mempunyai alamat ip address 192.168.1.2.

Pada gambar 3 dijelaskan diagram alir perancangan secara keseluruhan PC Router yang mampu mengatur bandwidth berbasis GUI.



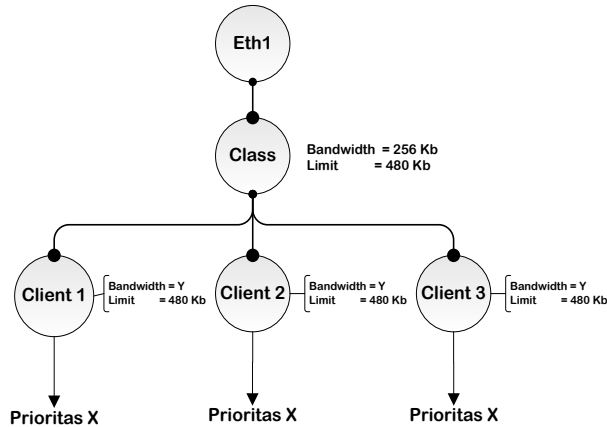
Gambar 3 Diagram Alur Perancangan Sistem Secara Umum

Pada diagram alir gambar 3 menjelaskan tentang perancangan sistem secara umum yang diawali dengan instalasi sistem operasi, konfigurasi server, instalasi aplikasi pendukung dan instalasi dan konfigurasi webHTB yang selanjutnya

## Implementasi Pengaturan Dan Prioritas Bandwidth Htb (Hierarchical Token Bucket) Berbasiskan Gui Pada Linux Server ClearOS

mengumpulkan data serta menganalisis hasilnya.

Sedangkan skema pada perancangan ini, *client* telah diklasifikasikan sesuai dengan *bandwidth* dan prioritas yang telah ditentukan, setelah itu akan dianalisa hasilnya. berikut adalah skema antrian webHTB.

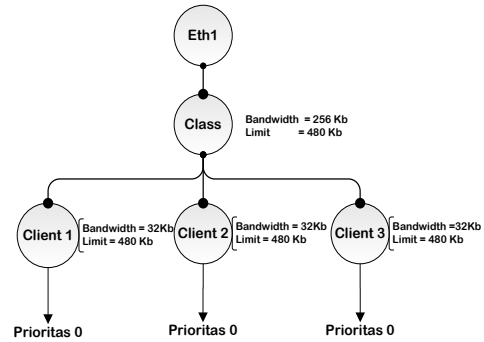


Gambar 4 Skema Dari Teknik Antrian Dan Pioritas HTB

1. Pada eth1 merupakan Ethernet untuk jaringan local (LAN).
2. Eth1 membuat sebuah kelas dengan dengan besar *bandwidth* 256 Kbps dan Limit 480 Kbps atau besar *bandwidth* maksimal 480 Kbps yang di dapat oleh client.
3. Class mempunyai 3 *client* yaitu *client* 1, *client* 2 dan *client* 3. Masing masing *client* mendapatkan limit sebesar 480kbps dan *bandwidth* dengan variable Y yang bersifat *integer* yang akan di tentukan pada saat tahapan pengujian sistem.
4. Setelah itu masing-masing *client* langsung dimasukan ke prioritas X yang sama mempunyai sifat *interger* yang akan d tentukan pada saat pengujian.

### 4. PENGUJIAN DAN ANALISA Pengujian Dengan Bandwidth Sama Tanpa Prioritas

Pada pengujian ini dilakukan dengan cara mengatur *bandwidth* dengan memberikan *bandwidth* yang sama pada *client* tanpa menggunakan prioritas seperti pada gambar 1



Gambar 6 Skema Dengan Bandwidth Sama Tanpa Prioritas

Berdasarkan pada pengujian pengaturan *bandwidth* berbeda, tanpa menggunakan prioritas, berikut akan dituliskan dalam bentuk tabel 1

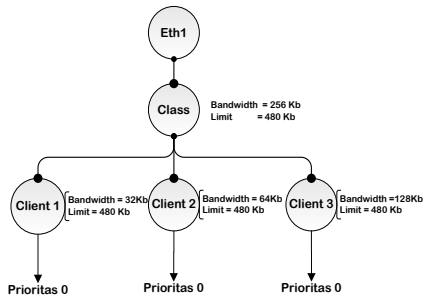
Tabel 1 Perbandingan Bandwidth Sama Tanpa Prioritas

Nama	Bandwidth (kbps)	Limit (kbps)	Prioritas	Transfer Rate (kbps)
Client 1	32	480	0	155,41
Client 2	32	480	0	156,94
Client 3	32	480	0	155,53

pengujian nilai *limit* sama dan *bandwidth* yang sama menghasilkan rata – rata *transfer rate* yang sama pada masing – masing *client*.

### Pengujian Dengan Bandwidth Berbeda Tanpa Prioritas

Pengujian tahap pertama dilakukan dengan cara mengatur *bandwidth* pada ketiga *client* dengan *bandwidth* berbeda yaitu *client* 1 32kbps, *client* 2 64kbps dan *client* 3 128kbps. Dan limit yang sama 480kbps serta tanpa menggunakan prioritas.



Gambar 7 Skema Pembagian bandwidth berbeda dan Tanpa Prioritas

Berdasarkan pada pengujian pertama, berikut akan dituliskan dalam bentuk tabel 2. Bagaimana hasil perbandingan antara pengaturan *bandwidth* berbeda, tanpa menggunakan prioritas.

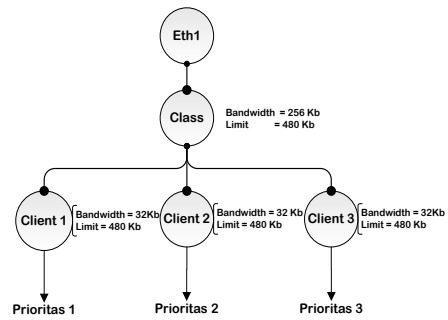
Tabel 2 Skema Pembagian *bandwidth* berbeda dan Tanpa Prioritas

Nama	Bandwidth (kbps)	Limit (kbps)	Prioritas	Transfer Rate (kbps)
Client 1	32	480	0	115,36
Client 2	64	480	0	141,48
Client 3	128	480	0	209,48

Dapat dilihat pada tabel 2 bahwa pengujian kecepatan pembagian *bandwidth* dengan nilai limit yang sama pada ketiga *client*nya dan nilai *bandwidth* yang berbeda untuk masing masing *client* kecepatan tertinggi dicapai pada pengujian *client* dengan nilai *bandwidth* yang tertinggi.

### Pengujian Dengan Bandwidth Sama Menggunakan Prioritas

Pengujian kedua dilakukan dengan cara mengatur *bandwidth* ketiga *client* dengan besar *bandwidth* yang sama yaitu 32 kbps, tetapi ketiga *client* akan mendapatkan prioritas yang berbeda dengan urutan prioritas *client* 1 dengan prioritas 1, *client* 2 dengan prioritas 2, dan *client* 3 dengan prioritas 3.



Gambar 8 Skema *Bandwidth* Sama dengan prioritas

Berdasarkan pada pengujian tahap kedua, berikut akan dituliskan dalam bentuk tabel 3 Bagaimana hasil perbandingan antara pengaturan *bandwidth* sama, dengan prioritas berbeda.

Tabel 3 Perbandingan Bandwidth Sama menggunakan Prioritas

Nama	Bandwidth (kbps)	Limit (kbps)	Prioritas	Transfer Rate (kbps)
Client 1	32	480	1	401,36
Client 2	32	480	2	32,24
Client 3	32	480	3	28,30

Dapat dilihat pada tabel 3 pengambilan data untuk pengujian kecepatan dengan prioritas kecepatan tertinggi didapat pada *client* dengan nilai prioritas satu.

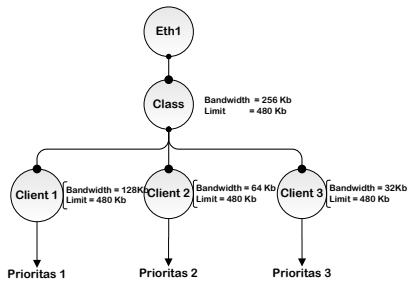
### Pengujian Dengan Bandwidth Beda Menggunakan Prioritas

Pada pengujian ini terdapat 2 kondisi yaitu

#### 1. Pengujian dengan bandwidth besar dan prioritas tinggi.

Pada pengujian *client* akan diberi bandwidth besar dan akan mendapatkan prioritas yang tinggi. Untuk lebih jelasnya dapat dilihat pada gambar 8

# Implementasi Pengaturan Dan Prioritas Bandwidth Htb (Hierarchical Token Bucket) Berbasiskan Gui Pada Linux Server ClearOS



Gambar 8 Skema Dengan Bandwidth Besar Dan Prioritas Tinggi

Berdasarkan pada pengujian dengan *bandwidth* besar dan prioritas tinggi, berikut akan di tuliskan dalam bentuk tabel 4

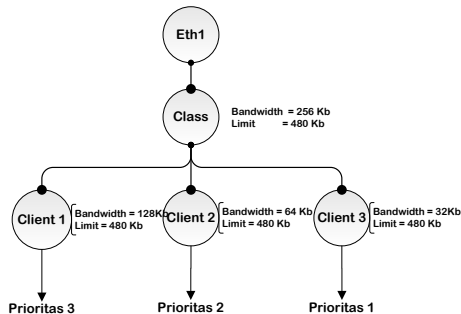
Tabel 4 Bandwidth Besar Dan Prioritas Tinggi

Nama	Bandwidth (kbps)	Limit (kbps)	Prioritas	Transfer Rate (kbps)
Client 1	128	480	1	355,22
Client 2	64	480	2	124,75
Client 3	32	480	3	30,07

Dapat dilihat pada tabel 4.4 bahwa nilai *transfer rate* dicapai oleh *client* satu dengan pengaturan *bandwidth* 128 dan prioritas satu.

## 2. Pengujian dengan bandwidth besar dan prioritas rendah.

*Client* yang mempunyai *bandwidth* besar akan mendapatkan prioritas yang rendah dan *client* yang mempunyai *bandwidth* kecil akan mendapatkan prioritas yang tinggi



Gambar 8 Skema Pembagian Bandwidth Beda dengan prioritas Berbeda

Berdasarkan pada pengujian, berikut akan dituliskan dalam bentuk tabel 5 Bagaimana hasil perbandingan antara pengatur *bandwidth* besar, dengan menggunakan prioritas yang rendah.

Tabel 5 Perbandingan Bandwidth berbeda dengan Prioritas Berbeda

Nama	Bandwidth (kbps)	Limit (kbps)	Prioritas	Transfer Rate (kbps)
Client 1	128	480	3	129,33
Client 2	64	480	2	59,98
Client 3	32	480	1	275,79

## Analisa

Dari pengujian yang telah dilakukan yaitu ketika *client* di konfigurasi *bandwidth* yang sama dan berbeda tanpa menggunakan prioritas membuktikan bahwa konsep *link sharing* pada metoda *Hierarchical Token Bucket* (HTB) telah berjalan semestinya, karena jumlah *bandwidth* pada *client* diantara *base rate* nilai *ratanya*. Hal ini di buktikan dengan semua hasil pengujian yaitu jumlah *transfer rate* pada semua *client* tidak melebihi dari limit atau *rate* yang telah di tetapkan pada kelas tersebut yaitu sebesar 480 kbps.

Pengujian dengan menggunakan prioritas telah membuktikan bahwa nilai prioritas pada metoda *Hierarchical Token Bucket* (HTB) memiliki peranan paling penting pada *transfer rate* dan sesuai dengan konsep *priority service* (PS) yaitu Selama periode lalulintas penuh, akan memperlambat prioritas lebih rendah dari aplikasinya dengan menurunkan prioritas aplikasinya sehingga meningkatkan *bandwidth* ke aplikasi prioritas yang lebih tinggi. Dimana nilai *bandwidth* menentukan *transfer rate* minimal pada suatu *client*.

## 5. KESIMPULAN DAN SARAN

### Kesimpulan

- Pada pengujian *bandwidth* sama dan tanpa prioritas terdapat nilai *transfer rate* *client1*, *client2*, *client3* masing – masing 155,41 kbps, 156,94 kbps dan 155,58 kbps sedangkan menggunakan

- prioritas masing – masing *client* 401,36 kbps , 32,24 kbps dan 28,30 kbps.
2. Pada pengujian *bandwidth* berbeda tanpa prioritas terdapat nilai *transfer rate client1, client2, client3* masing 115,36 kbps, 141,48 kbps dan 209,48 kbps sedangkan menggunakan prioritas terdapat dua kondisi yang pertama *bandwidth* besar prioritas tinggi masing – masing *transfer rate* 355,22 kbps, 124,75 kbps dan 30,07 kbps yang kedua *bandwidth* besar prioritas rendah masing – masing *transfer rate* 129,33 kbps , 59,98 kbps dan 275,79 kbps.
  3. Nilai prioritas pada metode HTB memegang peranan paling besar untuk *transfer rate* pada *client*. Dimana *bandwidth* menentukan nilai *transfer rate* minimal
  4. Metode HTB yang diimplementasikan pada linux server Clear OS dapat melakukan pembatasan *bandwidth* dari seluruh *client* yang ada dan aplikasi *webHTB* dapat membantu memudahkan administrator dalam melakukan manajemen *bandwidth* di lingkungan jaringan yang dikelolanya.
- [4] Forouzan, Behrouz A and Chung Fegan Sophia , *Data Communication And Networking*. Fourth Edition. New York: McGraw-Hill, 2007.
- [5] Yugianto, Gin-Gin dan Rachman Oscar, *Router Teknologi, Konsep, Konfigurasi, dan Troubleshooting Berbasis Windows, Cisco, MacOS, Linux & Mikrotik Router*, Informatika Bandung, 2012.

#### Saran

1. Untuk mempermudah pengaturan *bandwidth* ini. Sistem yang sudah di konfigurasi dibuat ke dalam *live CD*, sehingga mudah digunakan.
2. Untuk perkembangan selanjutnya perlu dicoba juga dengan menggunakan OS lain seperti FreeBSD atau NetBSD dengan mengkhususkan pada penelitian tentang metode HTB.

#### DAFTAR PUSTAKA

- [1] Sopandi, D., *Instalasi Dan Konfigurasi Jaringan Komputer* Bandung: Informatika Bandung, 2008.
- [2] Sofana, I, *Membangun Jaringan Komputer*, Informatika Bandung, 2008.
- [3] Andi *Panduan Lengkap Pengembangan Jaringan Linux*. Semarang: Wahana Komputer, 2008.