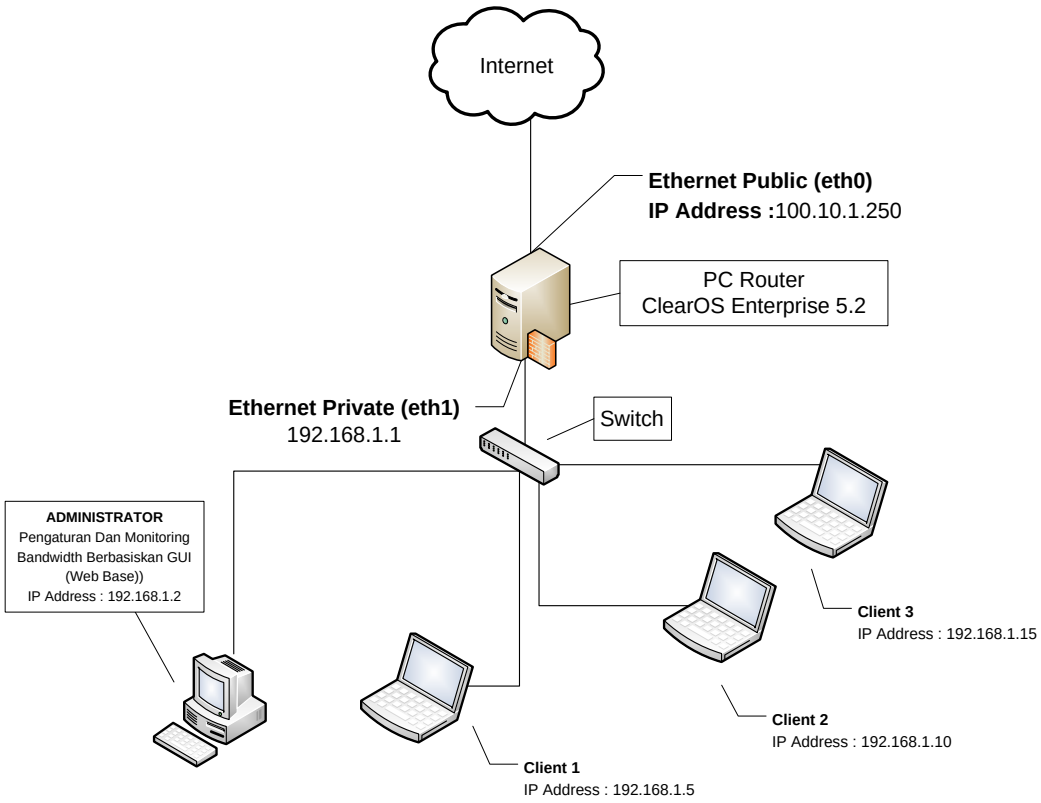
BAB III
PERANCANGAN SISTEM

Bab ini akan memaparkan bagaimana sistem dirancang sedemikian rupa sehingga menjadi PC *Router* yang mampu mengatur *bandwidth* dan prioritas ke setiap *client* dengan reabilitas yang baik, untuk lebih jelasnya dapat dilihat topologi pada gambar 3.1 :



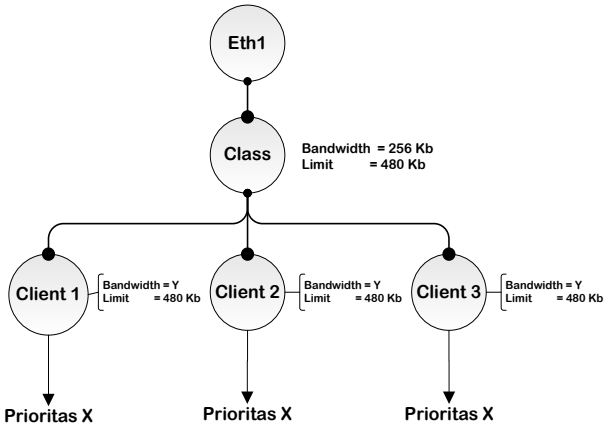
Gambar 3.1 Topologi Perancangan Jaringan Secara Umum

Dari gambar 3.1 diatas dijelaskan bagaimana topologi perancangan sistem pengaturan *bandwidth* berbasiskan GUI (G*raphical User Interface*) :

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 e*thernet 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* dan prioritas ke *Client*1 IP*address* 192.168.1.5 dan *client2* dengan IP*address* 192.168.1.10 serta *client3* dengan IP *address* 192.168.15.
7. Pengaturan *bandwidth* dan monitoring dilakukan pada PC *administrator* yang mempunyai alamat IP *address* 192.168.1.2.

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 3.2 Skema Dari Teknik Antrian Dan Pioritas HTB

Penjelasan dari skema pada gambar 3.2 dalam perancangan pengaturan *bandwidth* menggunakan teknik antrian 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 variabel 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 ditentukan pada saat pengujian.

3.1 Komponen Sistem

Pada perancangan perangkat PC *Router* terdapat beberapa komponen dalam proses perancangannya baik perangkat keras maupun perangkat lunak.

3.1.1 Perangkat Keras

Pada bagian ini menjelaskan perangkat keras yang dibutuhkan untuk server *PC Router* untuk pengaturan *bandwidth* berbasiskan GUI, di jelaskan pada tabel 3.1.

Tabel 3.1 Perangkat Keras Yang Digunakan

Perangkat Keras	Jumlah	Spesifikasi
PC Router	1 unit	a) Prosesor Intel Dual Core E 5500 2,80 Ghz b) RAM 1GB c) Harddisk 40GB d) 2 buah LAN card D-Link 10/100 *Fast EthernetPCI*
Client	3 unit	PC atau Notebook
Switch	1 unit	Switch 10/100 Mbps
Kabel UTP	4 Meter	Belden Australia

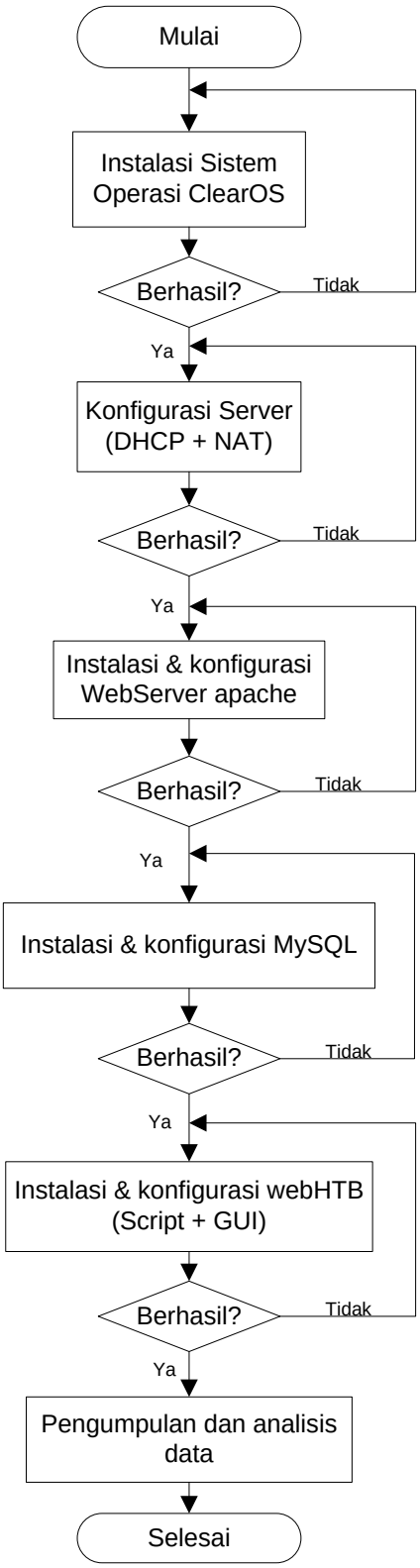
3.1.2 Perangkat Lunak

Pada bagian ini menjelaskan perangkat lunak yang dibutuhkan untuk server *PC Router* untuk pengaturan *bandwidth* berbasiskan GUI.

1. ClearOS *Enterprise* 5.2 *Service Pack* 1 sebagai sistem operasi Linux.
2. *Iptables* sebagai NAT (*Network Address Translation*) di Linux.
3. *Dnsmasq* sebagai *server* ip (DHCP) di Linux.
4. *Apache* sebagai *web server* di Linux.
5. WebHTB sebagai aplikasi berbasis *web* yang digunakan untuk pengaturan *bandwidth* dan prioritas *client* pada suatu jaringan.
6. MySQL sebagai penyimpanan *database. database* ini cukup mendukung bahasa pemograman PHP dan web *server Apache*.
7. Ping, yaitu perangkat lunak yang digunakan untuk mengukur *response time* dan *packet loss.*

3.2 Instalasi Sistem

Setelah komponen perangkat keras yang berupa komputer yang dijadikan router dan perangkat jaringan lainya serta komponen perangkat lunak yang mendukung pembuatan sistem pengaturan *bandwidth* telah siap, selanjutnya instalasi dan konfigurasi sistem. Berikut Pada gambar 3.1 dijelaskan diagram alir perancangan secara keseluruhan PC Router yang mampu mengatur *bandwidth* berbasiskan GUI.



Gambar 3.3 Diagram Alir Perancangan Sistem Secara Umum

Berdasarkan pada gambar 3.2 dapat dijelaskan sebagai berikut:

1. Pada proses pertama yaitu diawali dengan instalasi sistem operasi pada PC *router*. Sistem operasi yang digunakan pada PC Router adalah Linux *ClearOS Enterprise 5.2 Service Pack 1* setelah selesai instalasi dan berhasil selanjutnya akan dikonfigurasi NAT dan DHCP bila tidak berhasil kembali pada proses instalasi sistem operasi.
2. Selanjutnya konfigurasi NAT agar IP dalam jaringan *local* dapat tersambung pada jaringan internet. Dan konfigurasi DHCP agar *client* dapat di konfigurasi secara otomatis jika gagal kembali mengkonfigurasi NAT dan DHCP, jika berhasil bisa melanjukan ke tahap selanjutnya.
3. Berikutnya adalah instalasi dan konfigurasi *web server apache, web server* diperlukan karena sistem pengaturan *bandwidth* dan layanan-layanan administrasi server dilakukan dengan menggunakan program aplikasi berbasis *web* jika gagal kembali pada tahap sebelumnya dan jika berhasil selanjunya mengkonfigurasi DBMS.
4. Melakukan Instalasi dan konfigurasi DBMS (*Databases Management System)* MySQL dan phpMyAdmin selanjunya ke tahap berikutnya jika tidak ulangi prosesnya.
5. Berikutnya instalasi dan konfigurasi webHTB dalam bentuk *script* agar dapat terhubung dengan database yang terdapat pada *web server* yang selanjutnya dapat dilakukan konfigurasi batasan *bandwidth* dan prioritas menggunakan webHTB berbasiskan GUI.
6. Setelah semua proses selesai tahap terakhir adalah melakukan pengujian pada sistem dan pengumpulan data serta menganalisis.

3.2.1 Instalasi Sistem Operasi

Sistem operasi yang digunakan pada PC *Router* adalah Linux ClearOS *Enterprise* 5.2 *Service Pack* 1 sedangkan pada *client* menggunakan sistem operasi Windows XP dan Windows 7. Instalasi Sistem Operasi Linux dapat dilakukan melalui beberapa media, diantaranya melalui CDROM atau melalui HTTP, pada instalasi kali ini akan digunakan CDROM untuk instalasi *base* sistem dan

selanjutnya instalasi dilakukan dengan melalui media internet yang sebagaimana terlampir pada lampiran B.

3.2.2 Instalasi dan Konfigurasi DHCP Server

DHCP (*Dynamic Host Cotrol Protocol*) server merupakan sistem yang digunakan untuk melayani *client* dalam membuat konfigurasi alamat IP secara otomatis sehingga tidak perlu untuk konfigurasi *client* dengan pengaturan jaringan secara manual satu-persatu. Dengan menggunakan DHCP server *client* cukup dikonfigurasi untuk menggunakan IP otomatis maka *client* akan secara otomatis akan mendapatkan konfigurasi jaringan yang kita tentukan sebelumnya pada DHCP server seperti konfigurasi ip, *nermask*, *default gateway*, dan juga DNS *server*.

Untuk memfungsikan Linux sebagai DHCP Server maka kita membutuhkan aplikasi dhcpd, yaitu aplikasi DHCP server pada Linux, berikut adalah perintah untuk menginstall dhcpd menggunakan `yum` :

```
# yum install dhcpd
```

Setelah instalasi selesai, konfigurasi file `dhcp.conf`yang berada pada *directory*`/etc/dnsmasq/dhcp.conf`,kemudianbuka file tersebut lalu masukan konfigurasi di bawah ini.

```
dhcp-option=eth1,1,255.255.255.0
dhcp-option=eth1,28,192.168.1.255
dhcp-option=eth1,3,192.168.1.1
dhcp-option=eth1,6,8.8.8.8,8.8.4.4
dhcp-range=eth1,192.168.1.2,192.168.1.15,12h
read-ethers
```

3.2.3 Konfigurasi NAT

Selanjutnya diperlukan konfigurasi NAT pada *router* agar semua *client* dapat terhubung pada jaringan luar. Berikut konfigurasi awal NAT pada Linux yang selanjutnya dapat dilihat pada lampiran B.

```
iptables --table nat --append POSTROUTING --out-
interface eth0 -j MASQUERADE
```

3.2.4 Instalasi Web Server

Pada komputer server diperlukan *web server* karena sistem pengaturan *bandwidth* dan layanan-layanan administrasi server dilakukan dengan menggunakan program aplikasi berbasis *web*. *Web server* yang digunakan pada sistem ini adalah *apache.* Langkah-langkah konfigurasi *apache* dengan mengetikan perintah-perintah berikut pada *shell*:

1. Instalasi *apache*

Untuk melakukan instalasi *apache*,ketikan perintah berikut pada *shell*Linux. Baris pertama pada perintah dibawah ini berfungsi untuk memasang paket *apache*yang terdapat pada Linux server clearOS.

```
#yum install apache
```

2. Konfigurasi *filehttdoc.conf*

Pada *web server apache* terdapat pengaturan lokasi tempat *menyimpan file-file web* yang akan ditampilkan ke *browser*. Jadi *file-file web* tidak boleh disimpan di sembarang lokasi. Pengaturan lokasi tersebut dapat dilakukan dengan melakukan konfigurasi pada file *httdoc.conf*. Apabila *file web* disimpan di luar lokasi yang ditentukan, maka *file* tersebut tidak akan dibaca oleh *web browser.* Pada Linux penempatan*file-fileweb*di direktori khusus yaitu direktori `/var/www/` atau dalam `/home/user/public_html`.

3. *Restart apache*

Agar konfigurasi baru yang telahdimasukan dapat dijalankan maka sistem*apache* perlu di *restart*terlebih dahulu.

```
#service httpd restart
```

3.2.5 Instalasi MySQL

Sebagai media penyimpanan data, sistem ini menggunakan basis dataMySQL. Pemilihan basis data ini karena MySQL merupakan basis data yang *free* namun memiliki kehandalan yang sangat baik. Dan basis data ini juga cukup mendukung bahasa pemograman PHP dan web *server apache*. Instalasi MySQL :

```
#yum install app-mysql
```

3.3 Instalasi dan Konfigurasi webHTB

Setelah *webserver* dan MySQL *server* terinstall pada ClearOS, selanjutnya tahap instalasi aplikasi webHTB. Berikut konfigurasi *script* awal instalasi HTB yang selanjutnya dapat dilihat pada lampiran B.

1. Masuk ke direktori konfigurasi HTB.

```
#cd /var/www/html
```

2. Mengunduh program aplikasi webHTB.

```
#wget http://webhtb.nethd.ro/downloads/WebHTB_V2.9.bz2
```

3. Setelah selesai mengunduh selanjutnya exstrak file instalasi webHTB dengan perintah

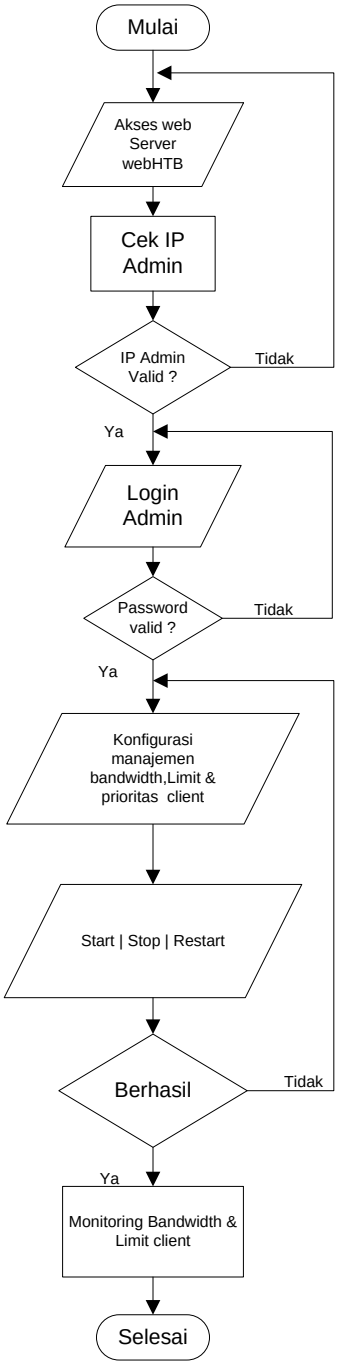
```
#tar -xjvf WebHTB_V2.9.bz2
#cd webhtb
```

4. Selanjutnya mengubah akses folder direktori agar bisa di konfigurasi

```
#chmod -R 777 *
```

5. Tahap selanjutnya administrator membuka aplikasi webHTB dengan cara mengetikan alamat IP *router* diikuti direktori webHTB pada *browser*.

Pada gambar 3.4 dijelaskan bagaimana cara kerja konfigurasi pengaturan *bandwidth*, limit dan prioritas berbasiskan GUI menggunakan aplikasi webHTB.

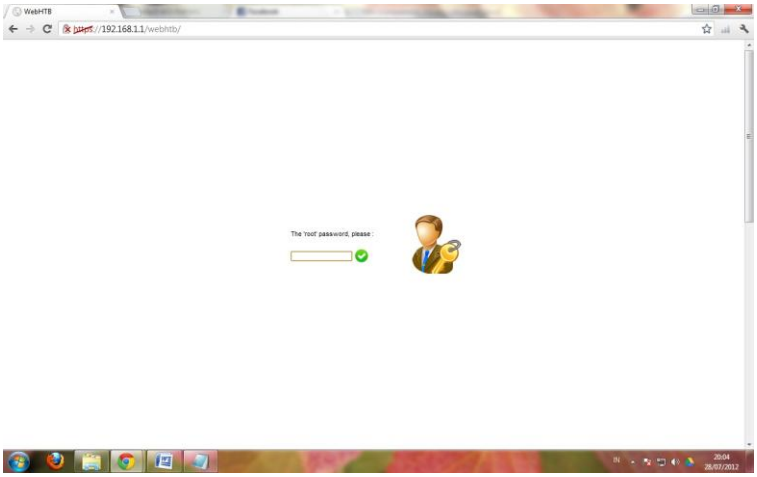


Gambar 3.4 Diagram Alir Konfigurasi *Bandwidth* Dan Limit pada WebHTB

Berdasarkan pada gambar 3.3 dapat dijelaskan sebagai berikut :

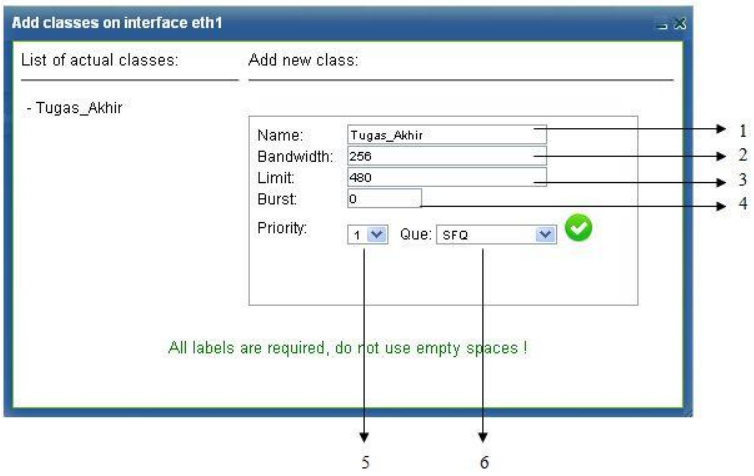
1. Proses pertama dimulai dengan administrator membuka aplikasi webHTB pada PC router dengan cara mengetikan alamat IP *router* diikuti direktori webHTB menggunakan *web browser* pada komputer administrator.
2. Kemudian PC router melakukan pengecekan terhadap IP komputer administrator jika IP komputer administrator valid maka akan masuk ke halaman *login* dan jika gagal cek kembali IP komputer administrator.
3. Proses selanjutnya yaitu administrator *login* pada halaman pengaturan *bandwidth* webHTB dan jika gagal administrator melakukan *login* ulang.
4. Selanjutnya melakukan konfigurasi pembuatan *class, sub class* dan menetukan *rule* yang sesuai skema yang telah di rencanakan.
5. Tahap selanjutnya melakukan konfigurasi *bandwidth* dan *limit* pada masing-masing *client*.
6. Berikutnya administrator melakukan *start, stop,* atau *restart script* pengaturan *bandwidth* pada webHTB dan jika berhasil selanjutnya melakukan monitor terhadap *client* , dan jika gagal administrator melakukan konfigurasi ulang.
7. Selanjutnya administrator melakukan monitoring *speed* terhadap batasan *bandwidth* dan prioritas yang telah di konfigurasi sebelumnya.
8. Administator selesai melakukan batasan *bandwidth* dan limit pada HTB.

Berikut ini merupakan tampilan GUI *login* administrator webHTB pada pengaturan *bandwidth* menggunakan teknik antrian HTB. Yang dapat dilakukan dengan cara mengakses *web server* pada Linux *server* ClearOS. Sebagai contohnya, karena server menggunakan alamat IP 192.168.1.1 maka administrator dapat mengakses alamat https://192.168.1.1/webhtb/ pada browser seperti ditunjukan pada gambar 3.5.



Gambar 3.5 Tampilan GUI Login Administrator WebHTB

Untuk membuat skema pengaturan *bandwidth* seperti pada gambar 3.2 proses pertama adalah membuat *class* dan *sub class* baru. Klik *add classes* untuk membuat *class* baru. Selanjutnya akan tampil seperti pada gambar 3.6 dan administrator melakukan konfigurasi pengaturan *bandwidth, limit* dan prioritas pada *class* tersebut.



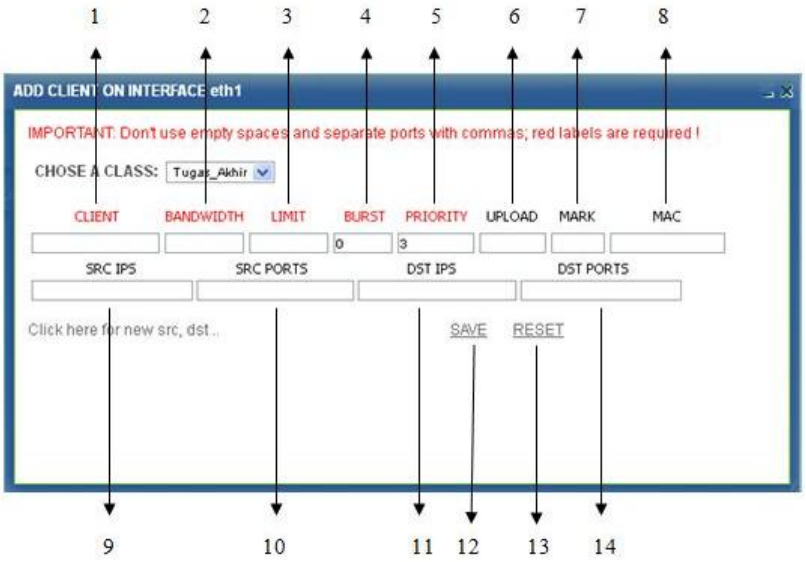
Gambar 3.6 Tampilan Form Pembuatan Sebuah *Class*

Keterangan:

1. Name; untuk membuat nama pada suatu *class*.
2. *Bandwidth*; parameter nilai kecepatan minimal yang akan diberikan terhadap *class* yang telah dibuat.

3. Limit; parameter nilai kecepatan maksimal yang akan diberikan pada suatu *class*.
4. Burst; waktu yang dibutuhkan untuk mencapai *bandwidth* limit.
5. Priorty; parameter nilai yang di berikan oleh adminstrator jaringan untuk mengalokasikan *bandwidth* berdasarkan pentingnya suatu *class*. Semakin kecil nilai priority semakin diutamakan kecepatan *bandwidth* yang diberikan.
6. Que; tipe antrian yang akan digunakan.

Setelah *class* terbentuk tahap selanjutnya administrator membuat *sub class* dengan cara klik *add client* dan melakukan konfigurasi *bandwidth* dan *limit* yang terdiri dari tiga *client* yang berbeda, ditunjukan pada gambar 3.6.



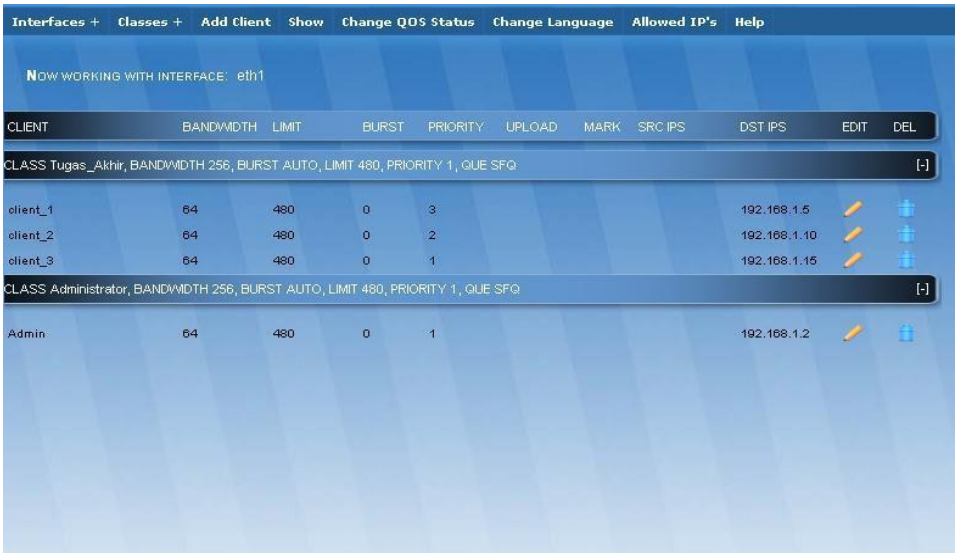
Gambar 3.7 Tampilan Form Pembuatan *Sub Class*

Keterangan:

1. *CLIENT*; untuk membuat nama pada *client* atau *user*.
2. BANDWIDTH; parameter nilai kecepatan minimal yang akan diberikan terhadap *client* yang telah dibuat.
3. LIMIT; parameter nilai kecepatan maksimal yang akan diberikan pada suatu *client*.
4. BURST; waktu yang dibutuhkan untuk mencapai *bandwidth* limit.

5. PRIORTY; parameter nilai yang di berikan oleh adminstrator jaringan untuk mengalokasikan *bandwidth* berdasarkan pentingnya suatu *client.* Semakin kecil nilai priority semakin diutamakan kecepatan *bandwidth* yang diberikan.
6. UPLOAD; kecepatan unggah maksimal.
7. MARK; penanda pada *client*.
8. MAC; pembatasan *bandwidth* dan prioritas *client* dikenali bersdasarkan MAC address.
9. SRC IPS; pembatasan *bandwidth* berdasarkan alamat IP sumber.
10. SRC PORTS; pembatasan *bandwidth* berdasarkan port sumber.
11. DST IPS; pembatasan *bandwidth* berdasarkan alamat tujuan IP.
12. SAVE; menyimpan konfigurasi yang telah dibuat.
13. RESET; menghapus semua konfihurasi pada field.
14. DSC PORTS;pembatasan *bandwidth* berdasarkan port tujuan IP.

Jika administrator selesai mengkonfigurasi *class* dan *sub class* maka akan tampil halaman webHTB seperi pada gambar 3.5 tahap selanjutnya adalah menjalankan *script* dengan cara klik start QoS pada tab change QoS status.



Gambar 3.8 Tampilan GUI webHTB Setelah Terbentuk *Class* Dan *Sub Class*

.

BAB IV
PENGUJIAN DAN ANALISA

Bab ini membahas cara pengujian dari pengaturan *bandwidth* berbasiskan GUI (*Graphical User Interface*), dalam hal ini yang mejadi objek penelitian adalah Linux ClearOS, yaitu dengan melakukan pengujian terhadap *bandwidth* dan prioritas. Tujuannya adalah untuk mengetahui kelebihan kekurangan sistem apakah sistem yang telah dibuat sesuai dengan apa yang direncanakan, dan bekerja secara efektif sesuai dengan yang diharapkan.

4.1 Persiapan Pengujian

Sebelum pengujian dilakukan ada beberapa tahap yang harus diperiksa sebagaimana dilampirkan pada bagian lampiran C yaitu :

1. Memeriksa kartu jaringan *router*

   Sebelum pengujian dilakukan periksa terlebih dahulu *interface* jaringan pada *router* linux server ClearOS. Untuk memeriksa *interface* jaringan pada *router* gunakan perintah

   ```
   #ifconfig
   ```

2. Memeriksa *firewall*

   Selanjutnya adalah memeriksa apakah *firewall* sudah berjalan atau belum, karena jika belum berjalan maka *client* belum dapat terhubung ke jaringan luar (internet) untuk mengeceknya gunakan perintah berikut :

   ```
   #iptables –L
   #iptables –t nat –L
   ```

3. Memeriksa teknik antrian

   Setelah semua *client* terhubung pada jaringan selanjutnya mengecek apakah teknik antrian pengaturan *bandwidth* sudah berjalan dengan baik atau tidak, maka untuk melihat apakah teknik antrian telah bekerja gunakan perintah di bawah ini :

   ```
   #tc –s class show dev eth1
   ```

4.2 Tahap – Tahap Pengujian dan Pengambilan Data

Berikut pengujian dari sistem yang telah dirancang yaitu :

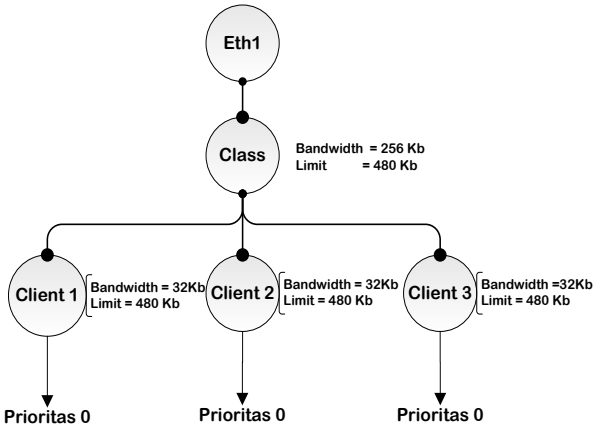
1. Dilakukan pengujian terhadap pengaturan *bandwidth* pada ketiga *client* dengan *bandwidth* sama dan tanpa menggunakan prioritas.
2. Dilakukan pengujian terhadap pengaturan *bandwidth* pada ketiga *client* dengan *bandwidth* yang berbeda dan tanpa menggunakan prioritas.
3. Dilakukan pengujian terhadap pengaturan *bandwidth* pada ketiga *client* dengan *bandwidth* sama dengan menggunakan prioritas.
4. Dilakukan pengujian terhadap pengaturan *bandwidth* pada ketiga *client* dengan *bandwidth* berbeda menggunakan prioritas.

4.3 Pengujian dan Pengambilan Data

Setelah sistem berjakan dengan baik selanjutnya adalah pengujian dan pengambilan data berikut adalah pengujian yang dilakukan.

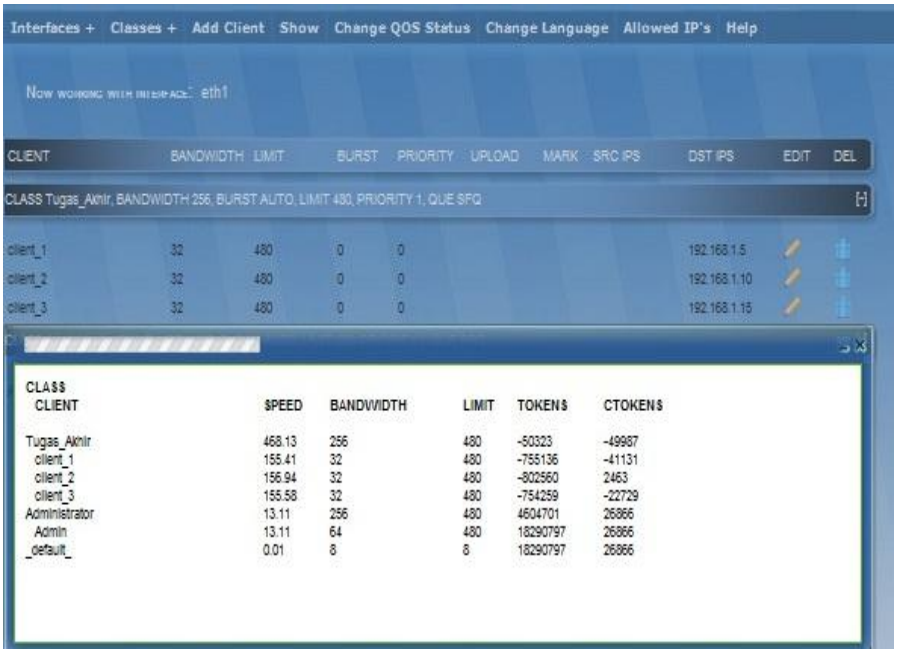
4.3.1 Pengujian dengan Bandwidth Sama Tanpa Prioritas

Pada pengujian ini dilakukan dengan cara mengatur *bandwidth* dengan memberikan *bandwidth* yang sama pada *client* tanpa menggunakan prioritas seperti pada gambar 4.1.



Gambar 4.1 Skema Pengaturan Bandwidth Sama Tanpa Prioritas

Berikut adalah tampilan webHTB setelah dikonfigurasi dengan *bandwidth* sama dan tanpa menggunakan prioritas seperti terlihat pada gambar 4.2.



Gambar 4.2 Tampilan WebHTB Pengaturan Bandwidth Sama Tanpa Prioritas

Berdasarkan pada pengujian pengaturan *bandwidth* sama, tanpa menggunakan prioritas, berikut akan dituliskan dalam bentuk tabel 4.1

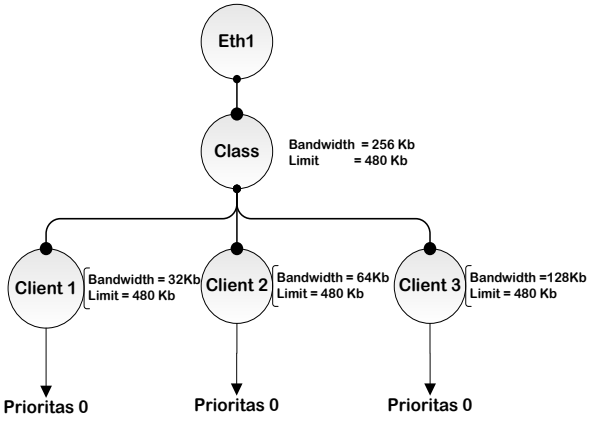
Tabel 4.1 Perbandingan Bandwidth Sama Tanpa Prioritas

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	155,58

Dapat dilihat pada tabel 4.1 pengujian nilai *limit* sama dan *bandwidth* yang sama menghasilkan rata – rata *transfer rate* yang sama pada masing – masing *client*.

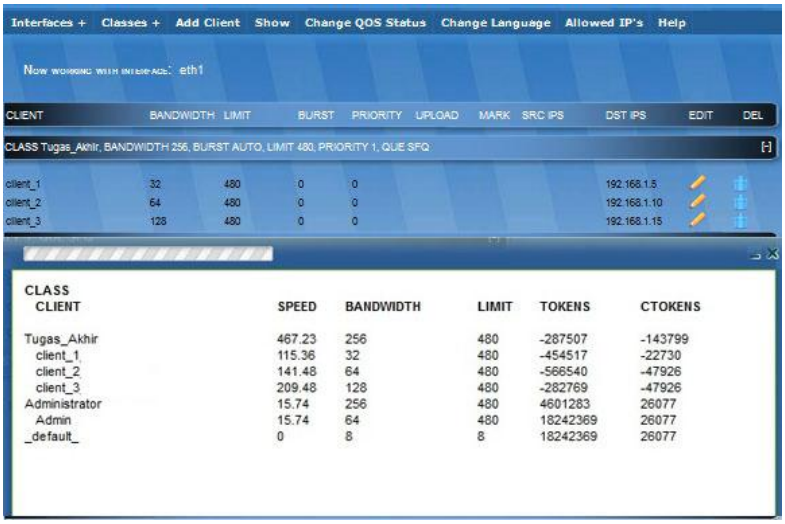
4.3.2 Pengujian dengan Bandwidth Berbeda Tanpa Prioritas

Pengujian 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 480 kbps serta tanpa menggunakan prioritas. Dapat dilihat pada gambar 4.3



Gambar 4.3 Skema Pembagian Bandwidth Berbeda Tanpa Prioritas

Berikut adalah tampilan webHTB setelah ketiga *client* di konfigurasi dengan pembagian *bandwidth* yang berbeda tanpa prioritas. Seperti terlihat pada gambar 4.4.



Gambar 4.4 Tampilan WebHTB Bandwidth Berbeda Tanpa Prioritas

Berdasarkan pada pengujian pengaturan *bandwidth* berbeda, tanpa menggunakan prioritas, berikut akan dituliskan dalam bentuk tabel 4.2. .

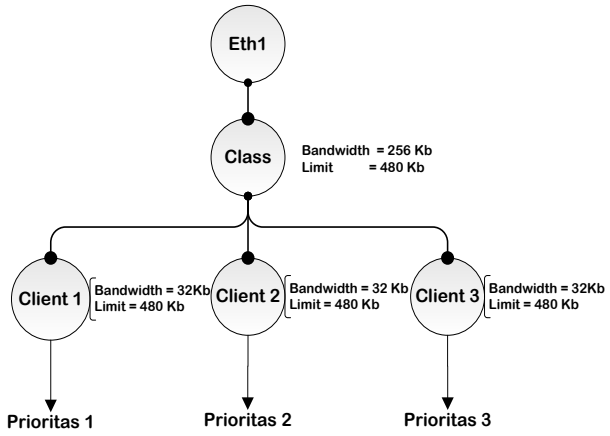
Tabel 4.2 Perbandingan Bandwidth Berbeda Tanpa Prioritas

Nama	*Bandwidth* (kbps)	Limit (kbps)	Prioritas	*Tranfer 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 4.2 bahwa pengujian *tranfer rate* dengan nilai limit yang sama pada ketiga *client*nya dan nilai *bandwidth* yang berbeda untuk masing masing *client Tranfer Rate* tertinggi dicapai pada *client* dengan nilai *bandwidth* yang tertinggi.

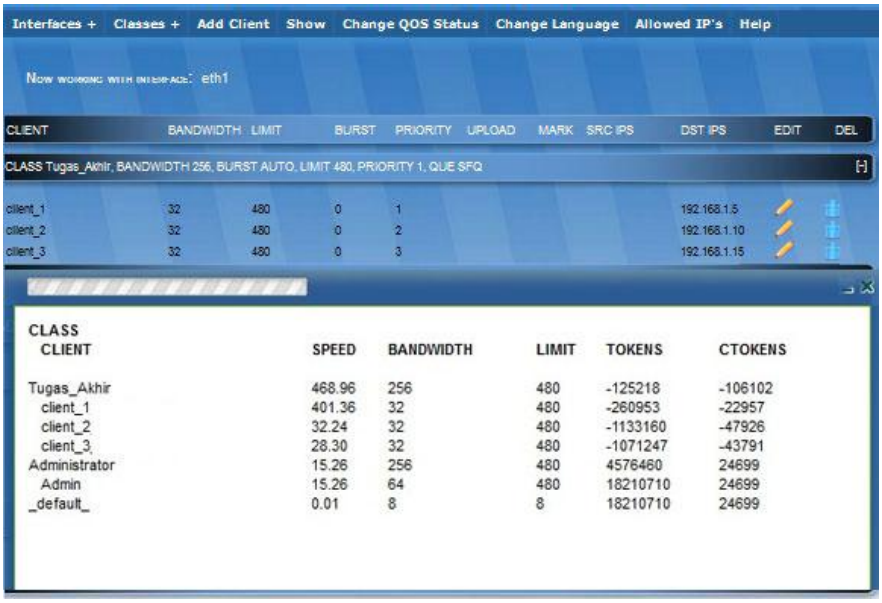
4.3.3 Pengujian dengan Bandwidth Sama Menggunakan Prioritas

Pengujian 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. Untuk lebih jelas dapat dilihat pada gambar 4.3.



Gambar 4.5 Skema Pembagian Bandwidth Sama dengan Prioritas

Berikut adalah tampilan webHTB pada saat konfigurasi ketiga *client* dengan konfigurasi besar *bandwidth* yang sama tetapi dengan prioritas. Terlihat pada gambar 4.6.



Gambar 4.6 Tampilan WebHTB Bandwidth Sama dengan Prioritas

Berdasarkan pada pengujian bandwidth sama dengan prioritas, berikut akan dituliskan dalam bentuk tabel 4.3 Bagaimana hasil perbandingan antara pengaturan *bandwidth* sama, dengan menggunakan prioritas.

Tabel 4.3 Perbandingan Bandwidth Sama dengan Menggunakan Prioritas

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

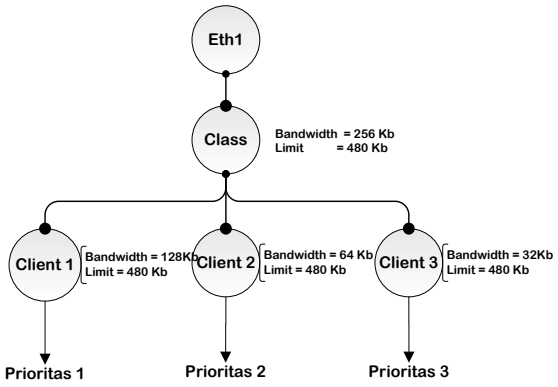
Dapat dilihat pada tabel 4.3 pengambilan data untuk pengujian *tranfer rate* dengan prioritas tertinggi didapat pada *client* dengan nilai prioritas satu.

4.3.4 Pengujian dengan Bandwidth Beda Menggunakan Prioritas

Selanjunya adalah mengatur *bandwidth* berbeda dengan prioritas berbeda. Pada pengujian *bandwidth* besar terdapat dua kondisi yaitu dengan prioritas tertinggi dan rendah.

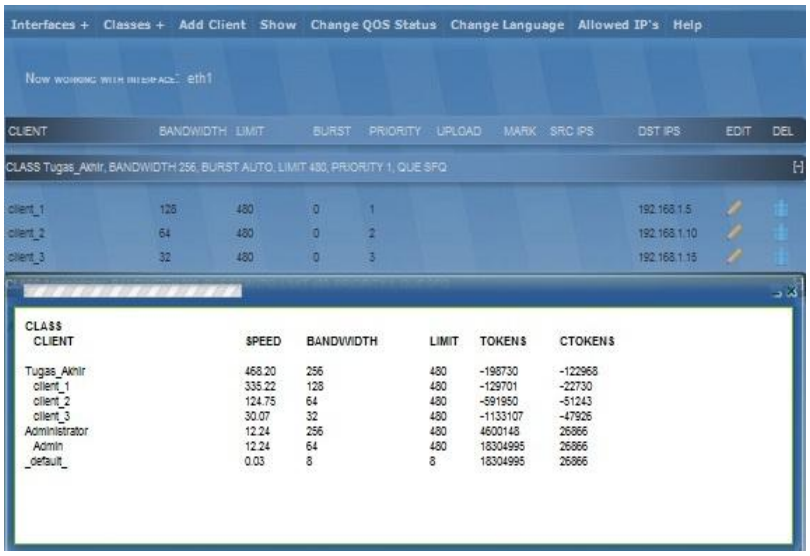
1. Pengujian prioritas tertinggi.

Pada pengujian *client* akan diberi bandwidth besar dan akan mendapatkan prioritas yang tinggi. Untuk lebih jelasnya dapat dilihat pada gambar 4.7



Gambar 4.7 Skema pengaturan Bandwidth Besar dengan Prioritas Tertinggi

Berikut ini tampilan webHTB pada saat konfigurasi *client bandwidth* besar dan prioritas tinggi di tunjukan pada gambar 4.8.



Gambar 4.8 Tampilan Webhtb Bandwidth Besar dengan Prioritas Tertinggi

Berdasarkan pada pengujian dengan *bandwidth* besar dan prioritas tinggi, berikut akan di tuliskan dalam bentuk tabel 1.4.

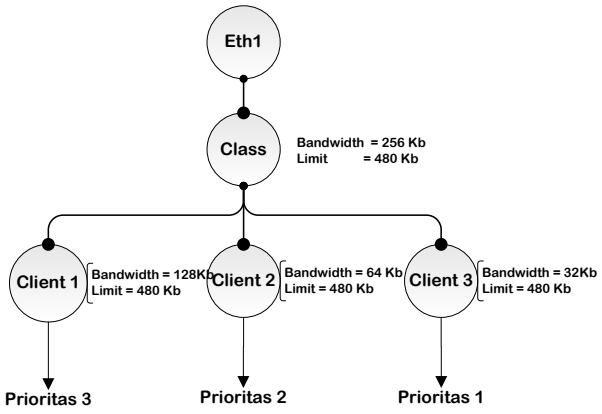
Tabel 4.4 Bandwidth Besar dan Prioritas Tertinggi

Nama	*Bandwidth* (kbps)	Limit (kbps)	Prioritas	*Tranfer 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 *bandwitdh* 128 dan prioritas satu.

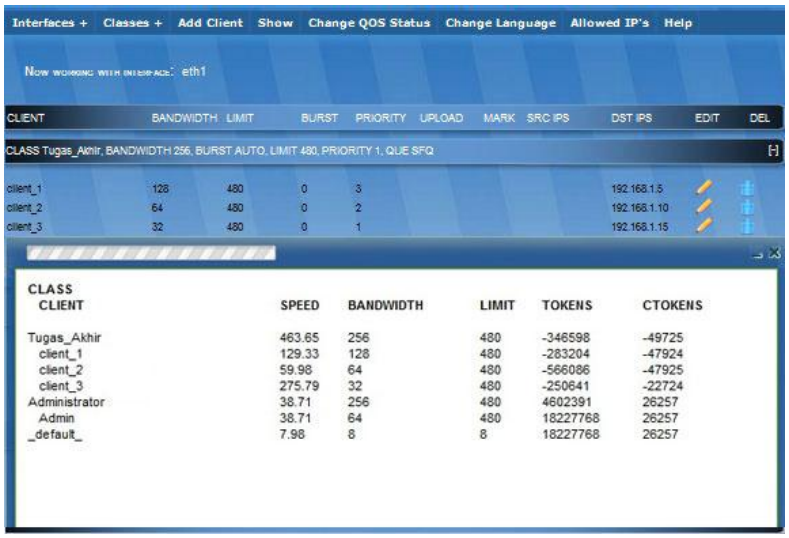
2. Pengujian prioritas rendah.

Client yang mempunyai *bandwidth* besar akan mendapatkan prioritas yang rendah dan *client* yang mempunyai *bandwidth* kecil akan mendapatkan prioritas yang rendah. Untuk lebih jelasnya dapat dilihat pada gambar 4.9



Gambar 4.9 Skema Pengaturan Bandwidth Besar dengan Prioritas Rendah

Berikut adalah tampilan webHTB pada saat konfigurasi ketiga *client* dengan konfigurasi besar *bandwidth* yang berbeda dengan prioritas yang berbeda. ditunjukan pada gambar 4.10



Gambar 4.10 Tampilan Webhtb Bandwidth Besar dengan Prioritas Rendah

Berdasarkan pada pengujian tahap ketiga, berikut akan dituliskan dalam bentuk tabel. Bagaimana hasil perbandingan antara pengatura *bandwidth* berbeda, dengan menggunakan prioritas yang berbeda juga.

Tabel 4.5 Perbandingan Bandwidth Berbeda dengan Prioritas Rendah

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

Dapat dilihat pada tabel 4.5 bahwa nilai *transfer rate* tertinggi dicapai pada *client* tiga sedangkan pada *client* satu dan dua *transfer rate* mendekati daripada nilai *bandwidth*.

4.4 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 *baserate* nilai

*rate*nya. 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 meningktkan *bandwidth* ke aplikasi prioritas yang lebih tinggi. Dimana nilai *bandwidth* menentukan *tranfer rate* minimal pada suatu *client*.