MANAJEMEN TRAFIK DAN *BANDWIDTH* MENGGUNAKAN METODE CBQ (*CLASS BASSED QUEUE*) BERBASIS GNU/LINUX UNTUK OPTIMALISASI *CLOUD COMPUTING*

Andi Hasdiyansyah[1], Susmini Indriani Lestariningati[2], Taufiq Nuzwir Nizar[3]
[1,2,3] Jurusan Teknik Komputer Unikom, Bandung
[1]andi7vold@gmail.com

ABSTRAK

Cloud Computing merupakan layanan teknologi informasi yang bisa dimanfaatkan atau diakses oleh pelanggannya melalui jaringan Internet. Dalam penerapan Cloud Computing dibutuhkan trafik yang baik agar performa dan keandalan sistem tetap terjaga. Salah satu metode yang digunakan untuk pengendalian trafik yaitu Class Based Queue (CBQ). CBQ adalah disiplin antrian yang mengatur pemakaian bandwidth jaringan yang dialokasikan untuk tiap user, dapat menerapkan pembagian kelas dan share link bandwidth melalui struktur kelas-kelas secara hirarki. Pengaturan trafik dan bandwidth berdasarkan alamat IP, Port atau antarmuka yang dapat dikonfigurasi sesuai dengan keinginan. PC Router yang digunakan yaitu menggunakan sistem operasi Linux Ubuntu. Pengujian dilakukan pada Local Area Network dengan topologi jaringan Star. Pengujian dilakukan terhadap parameter kualitas layanan yaitu, delay, packet loss, throughput dan Bandwidth. Hasil pengujian delay dan packet loss yang diperoleh menurut standar TIPHON termasuk dalam kategori sangat bagus dan nilai Throughput yang baik. Dengan menerapkan manajemen trafik dan bandwidth menggunakan metode CBQ, dapat memberikan prioritas terhadap aplikasi-aplikasi yang kritis pada jaringan Cloud Computing seperti aplikasi server dan memaksimalkan penggunaan investasi pada jaringan seperti memaksimalkan pemakaian bandwidth.

Kata Kunci: Trafik, Bandwidth, Delay, Packet Loss, Throughput

1. PENDAHULUAN

Cloud computing merupakan layanan teknologi informasi yang bisa dimanfaatkan atau diakses oleh pelanggannya melalui jaringan internet dengan mudah sesuai kebutuhan diberbagai lokasi, dimana model ini memungkinkan untuk mengumpulkan sumberdaya komputasi dalam satu wadah seperti *network*, server, *storage, aplikasi*, dan *services*. *Cloud Computing* ini memberikan akses bagi para penggunanya dimana mereka mampu untuk membuat, mengubah, melihat, dan menyimpan data-data dari sejumlah komputer yang terdapat di tempat lain.

Dengan bertambahnya pelanggan pada *Cloud Computng* maka peningkatan trafik di jaringan semakin tinggi, sehingga dibutuhkan optimalisasi agar performa dan kehandalan sistem tetap terjaga. Salah satu metode optimalisasi pada jaringan adalah dengan pengendalian trafik jaringan dengan cara melakukan manajemen *bandwidth.* Terdapat beberapa metode untuk mengontrol *bandwidth* pada jaringan diantaranya adalah teknik antrian *CBQ (Class Based Queue). CBQ* adalah suatu algoritma penjadwalan yang di dasarkan pada pembagian paket ke dalam kelas-kelas dan menjadwalkan paket di dalam antrian dengan suatu transmisi *rate* tertentu. *CBQ* dapat mengalokasikan *bandwidth* untuk berbagai jenis trafik yang berbeda, sesuai dengan pembagiannya yang tepat untuk masing-masing trafik.

Dengan pengendalian trafik maka dapat memberikan jaminan kualitas suatu layanan jaringan / *QoS (Quality of services). QoS* mengacu kepada kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu melalui teknologi yang berbeda-beda.

Tujuan yang diharapkan adalah :

1. Mengatasi trafik yang tinggi sehingga tidak menyebabkan antrian dalam jaringan.
2. QoS *(Quality of Services)* yang dihasilkan oleh metode CBQ.
3. Membantu admin untuk mengontrol trafik dan *bandwidth* dalam jaringan *Cloud Computing.*

2. DASAR TEORI

2.1 *Cloud Computing*

Cloud computing merupakan layanan teknologi informasi yang bisa dimanfaatkan atau diakses oleh pelanggannya melalui jaringan internet. Kata-kata *"Cloud"* sendiri merujuk kepada simbol awan yang di dunia TI digunakan untuk menggambarkan jaringan internet (*internet cloud*). *Cloud computing* juga bisa diartikan sebagai suatu model yang memungkinkan jaringan dapat di akses dengan mudah sesuai kebutuhan diberbagai lokasi, dimana model ini memungkinkan untuk mengumpulkan sumberdaya komputasi seperti *network, server, storage, aplikasi*, dan *services* dalam satu wadah[1].

Dilihat dari jenis layanan tersendiri, *Cloud Computing* terbagi dalam 3 jenis layanan secara umum, yaitu sebagai berikut:

1. SaaS *(Software as a Services)*. *SaaS* adalah istilah terhadap *software* atau aplikasi tertentu berbasis internet yang ditawarkan oleh *provider* kepada pengguna.
2. PaaS *(Platform as a Services)*. PaaS adalah layanan yang menyediakan modul-modul siap pakai yang dapat digunakan untuk mengembangkan sebuah aplikasi, yang tentu saja hanya bisa berjalan diatas *platform* tersebut.
3. IaaS *(Infrastructure as a Services)*. IaaS adalah sebuah layanan yang menyewakan sumberdaya teknologi informasi dasar, yang meliputi media penyimpanan, *processing power*, *memory*, sistem operasi, kapasitas jaringan dan lain-lain, yang dapat digunakan oleh pelanggannya untuk menjalankan aplikasi yang dimilikinya.

2.2 Router

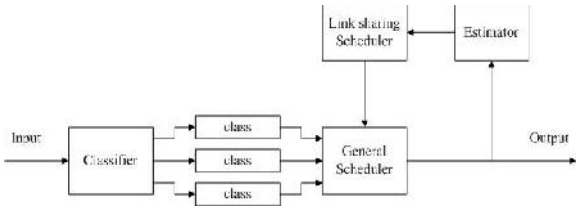
Router adalah perangkat jaringan komputer yang berfungsi untuk menghubungkan beberapa jaringan, dan meneruskan paket data dari satu jaringan ke jaringan lain, baik jaringan yang menggunakan teknologi yang sama atau yang berbeda. *Router* mengirimkan paket data melalui sebuah jaringan atau internet menuju tujuannya, melalui sebuah proses yang dikenal sebagai *routing*. Salah satu contoh perangkat *router* adalah PC *Router*.

PC *Router* adalah sebuah komputer dengan dua atau lebih interface jaringan yang dimodifikasi sedemikian rupa sehingga dapat digunakan sebagai *router*. PC *Router* pada umumnya memiliki fungsi seperti *router*, yaitu untuk mengirim paket data melalui sebuah jaringan atau internet menuju suatu tujuan sebagai penghubung dua buah jaringa atau lebih pada jaringan yang berbeda. Untuk membuat sebuah PC *router* harus memiliki *Ethernet* lebih dari satu atau minimal memiliki 2 *Ethernet*.

2.3 Teknik disiplin antrian CBQ

Class Based Queueing (CBQ) adalah disiplin antrian yang mengatur pemakaian *bandwidth* jaringan yang dialokasikan untuk tiap user, dapat menerapkan pembagian kelas dan *share link bandwidth* melalui struktur kelas-kelas secara hirarki[4]. Dengan CBQ, setiap kelas dapat mengalokasikan *bandwidth* miliknya untuk berbagai jenis trafik yang berbeda, sesuai dengan pembagiannya yang tepat untuk masing-masing trafik.

CBQ dapat digunakan untuk membatasi trafik Internet seperti trafik HTTP (web), mail, ftp, dan lain-lain. Selain itu CBQ juga memiliki fitur untuk mengatur *bandwidth* pada jam-jam tertentu.



Gambar 1 Diagram Blok Kerja CBQ

Ketika suatu paket datang, maka *classifier* memilih paket tersebut dan menempatkannya ke *class* yang bersesuaian, kemudian *class* tersebut akan mendapatkan *bandwidth* yang dibagikan oleh *general scheduler* kepada setiap *class* sesui dengan alokasinya masing-masing. sebelum paket dikeluarkan, bagian *estimator* akan

memeriksa apakah setiap *class* mendapatkan *bandwidth* sesuai dengan yang dialokasikan, jika suatu *class* kekurangan, maka *link sharing scheduler* akan meminjamkan *bandwidth* dari kelas yang memiliki *bandwidth* tidak terpakai ke kelas yang membutuhkan tambahan *bandwidth.*

2.4 *Quality of Service* (QoS)

Quality of Service (QoS) didefinisikan sebagai suatu pengukuran tentang seberpa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari suatu layanan. QoS mengacu pada kemampuan jaringan untuk memberikan layanan yang lebih baik untuk jaringan lalu lintas yang dipilih melalui berbagai teknologi yang berbeda-beda, termasuk *Frame Relay, Asynchronous Transfer Mode* (ATM), *Ethernet* dan 802.1 *network*, SONET, dan *IP-routed network*. Tujuan QoS adalah untuk memenuhi kenutuhan-kebutuhan layanan yang bebeda, yang menggunakan infrastruktur yang sama. Pada jaringan berbasis IP, IP QoS mengacu pada performansi dari paket-paket IP yang lewat melalui satu atau lebih jaringan. Berikut ini merupakan Parameter QoS:

1. *Throughput*, kecepatan (*rate*) transfer data efektif, yang diukur dalam bps. *Throughput* adalah jumlah total kedatangan paket yang sukses yang diamati pada *destination* selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut.
2. *Packet Loss*, suatu parameter yang menggambarkan suatu kondisi yang menunjukan jumlah total paket yang hilang..

Tabel 1 *Packet Loss*

KATEGORI DEGREDASI	Packet Loss
Sangat Bagus	0%
Bagus	3%
Sedang	15%
Jelek	25%

[sumber : TIPHON]

3. *Delay/Latency,* total waktu tunda suatu paket yang diakibatkan oleh proses transmisi dari satu titik ke titik lain yang menjadi tujuannya.

Tabel 2 Delay/Latency

KATEGORI DEGREDASI	Delay/Latency
Excellent	< 150 ms
Good	150 s/d 300 ms
Poor	300 s/d 450 ms
Unacceptable	> 450 ms

[sumber : TIPHON]

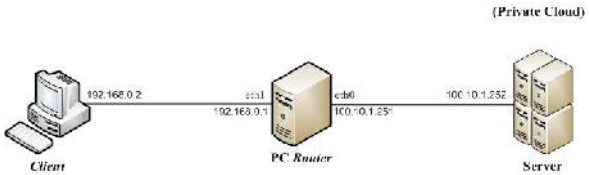
2.5 Sistem Operasi Linux

Linux adalah nama yang diberikan kepada sistem operasi komputer bertipe Unix. Linux merupakan salah satu contoh hasil pengembangan perangkat lunak bebas dan *open source*. Terdapat banyak distribusi Linux atau yang lebih dikenal dengan Distro Linux yang berkembang sampai menghasilkan Distro turunan, contohnya adalah Distro Debian. Ubuntu merupakan salah satu distribusi Linux yang berbasiskan Debian dan didistribusikan sebagai *software* bebas.

Ubuntu Server merupakan sistem operasi yang mendukung banyak sekali aplikasi-aplikasi server yang handal. Karena Ubuntu mempunyai sebuah *repository*. Sistem operasi Ubuntu mempunyai beberapa kelebihan, diantaranya yaitu adanya pemaketan yang memudahkan pengguna menambahkan maupun mengatur aplikasi apa saja yang ada pada server, adanya pilihan aplikasi yang sangat luas pada sistem operasi ubuntu, tersedianya pembaharuan yang dapat dipasang pada sistem operasi Ubuntu tersebut.

3. PERANCANGAN

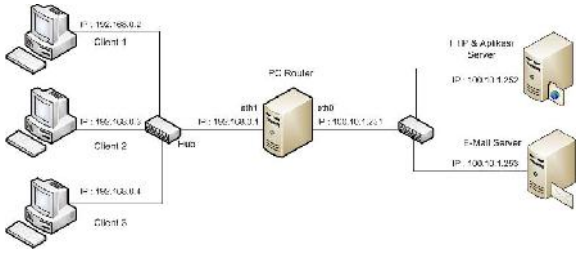
Pada topologi yang dibangun, jaringan *cloud computing* dirancang sebagai satu jaringan terpusat. Berikut ini adalah topologi sistem secara umum untuk jaringan *cloud computing*.



Gambar 2 Topologi Sistem Secara Umum

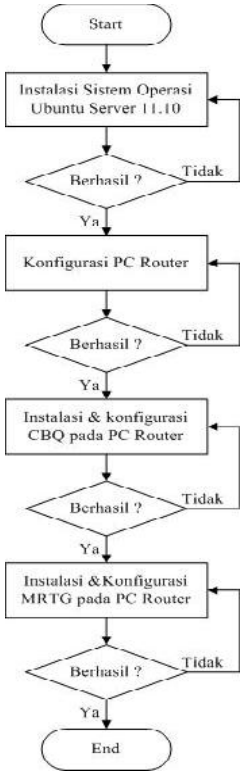
Untuk perancangan sistem ini dilakukan pada jaringan LAN yang menggambarkan sebagai jaringan *Cloud Computing* seperti pada

gambar 2. Gambar 3 adalah topologi sistem dalam jaringan lokal yang menunjukan koneksi jaringan *Cloud Computing*.



Gambar 3 Topologi sistem dalam LAN

Pada gambar 3 digunakan dua buah komputer sebagai server dan tiga buah komputer sebagai *client*. Dalam hal ini komputer server yang terdiri dari FTP Server, Aplikasi Server dan *E-mail* Server akan berfungsi sebagai jaringan *Cloud*. Sedangkan untuk komputer *client* akan digunakan sebagai penguji. Komputer *Cilent* dan server tergabung dalam jaringan yang berbeda yang dihubungkan oleh sebuah PC *Router*, sehingga *client* dapat mengakses komputer server melalui PC *Router* tersebut. Berikut ini adalah diagram alir perancangan PC *router.*



Gambar 4 Diagram Alir Sistem PC *Router*

Pada diagram alir gambar 4 menjelaskan tentang perancangan sistem secara umum yang diwali dengan instalasi sistem operasi, konfigurasi PC Router, instalasi dan konfigurasi CBQ serta instalasi dan konfigurasi MRTG,

3.1 Instalasi dan Konfigurasi CBQ

Setelah *instalasi sistem operasi dan konfigurasi pada PC router,* selanjutnya adalah tahap instalasi dan konfigurasi CBQ. Berikut adalah instalasi CBQ

1. *Copy* program CBQ direktori */sbin.*
2. Mengubah mode kepemilikan file *cbq.init.*
3. Membuat direktori CBQ.
4. Membuat Aturan untuk manajemen *bandwidth* di */etc/sysconfig/cbq*.

Adapun konfigurasi trafik CBQ dan alokasi *bandwidth* pada sistem ini yaitu di tunjukan pada tabel 3 dan tabel 4.

Tabel 3 *Rule* Filter Trafik

Nama	Bandwidth (kbps)
Client 1	128
Client 2	64
Client 2	32

Tabel 4 Konfigurasi Trafik CBQ

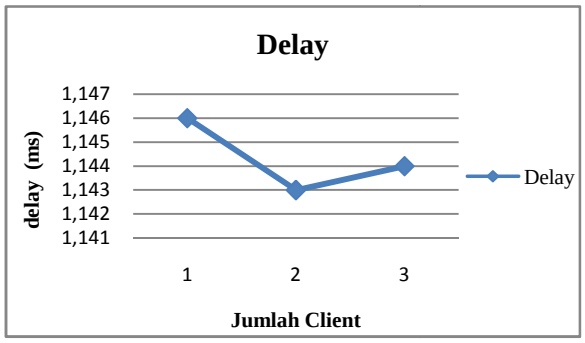
Nama	Filter Trafik
Client 1	DEVICE=eth1,10Mbit,1Mbit RATE=128Kbit WEIGHT=12.8Kbit LEAF=sfq RULE=192.168.0.3
Client 2	DEVICE=eth1,10Mbit,1Mbit RATE=64Kbit WEIGHT=6.4Kbit LEAF=sfq RULE=192.168.0.4
Client 2	DEVICE=eth1,10Mbit,1Mbit RATE=32Kbit WEIGHT=3.2Kbit LEAF=sfq RULE=192.168.0.5

4. HASIL PENGUJIAN

4.1 Tanpa Menggunakan Metode CBQ

Pada pengujian ini, kondisi yang diukur adalah trafik pada jaringan lokal tanpa menggunakan metode CBQ. Parameter yang diuji diantaranya *Delay* , *Packet Loss*, *Throughput*, dan *Bandwidth*. Berikut ini hasil pengukurannya:

1. *Delay.*



Gambar 5. Grafik *Delay* Terhadap Jumlah *Client*

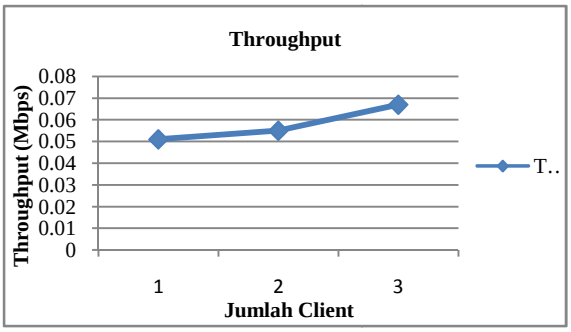
Dapat dilihat dari gambar 5, bahwa hasil pengukuran terhadap delay tanpa menggunakan CBQ hampir sama, baik itu terhadap 1 *client* ataupun beberapa *clinet.*

2. *Packet loss*

Tabel 5 *Packet Loss*

Jumlah *Client*	*Packet Loss* (%)
1	0
2	0
3	0

Dapat dilihat dari tabel 4.2, bahwa hasil pengukuran terhadap Packet Loss tanpa menggunakan CBQ adalah sama, meskipun jumlah *client* berubah-ubah.

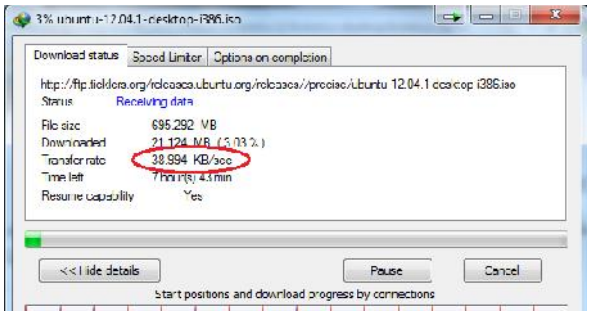


Gambar 6. Grafik *Throughput* Terhadap jumlah *client*

Berdaarkan Gambar 7, dapat dilihat bahwa untuk 1 *client* nilai rata-rata *throughput* tinggi, sedangkan ketika jumlah client bertambah, maka nilai dari throughput semakin menurun.

4. *Bandwidth*

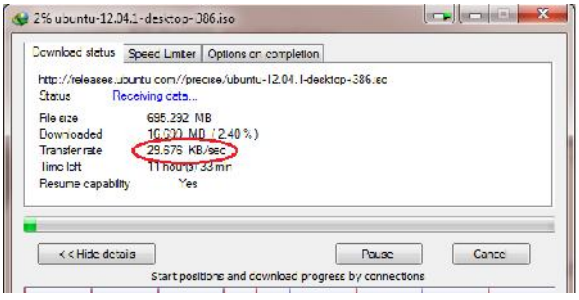
1) Pengukuran pada *Client* 1



Gambar 7. Pengamatan *Bandwidth* Pada *Client* 1

Pada gambar dapat dilihat kecepatan transfer pada saat *download* dapat mencapat 38,994 KB/sec.

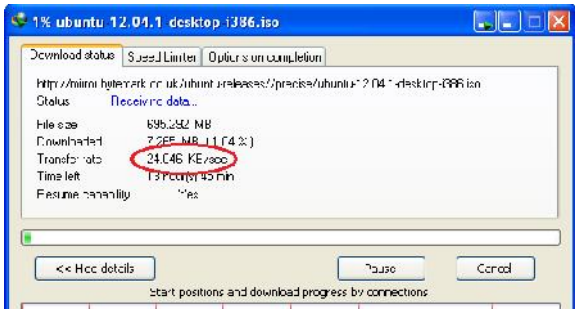
2) Pengukuran pada *Client* 2



Gambar 8 Pengamatan *Bandwidth* Pada *Client* 1

Pada gambar 8 dapat dilihat kecepatan transfer pada saat *download* dapat mencapai 29,676 KB/sec.

3) Pengukuran pada *Client* 3



Gambar 9 Grafik pemakaian *Bandwidth* pada *Client* 3

Pada gambar 9 dapat dilihat kecepatan transfer pada saat *download* dapat mencapat 24,046 KB/sec.

Berdasarkan pengujian, berikut ini adalah data yang diperoleh dari hasil pengukuran terhadap *bandwidth* dituliskan dalam bentuk tabel 8.

Tabel 6 Hasil Pengukuran Nilai *Bandwidth* Tanpa CBQ

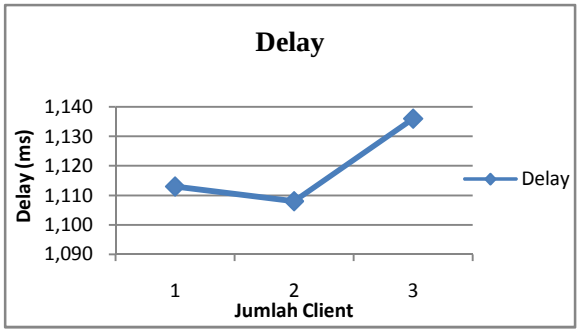
Nama	Transfer Rate (KB/sec)
Client 1	38,994
Client 2	29,676
Client 3	24,046

Dari hasil pengukuran dalam tabel 8 dapat dilihat bahwa setiap *client* memiliki *transfer rate* diatas 20 KB/sec, dan dari grafik pemakaian *bandwidth* dapat dilihat juga *transfer rate* yang tidak konstan.

4.2 Pengujian Trafik Dengan Menggunakan Metode CBQ

Pada pengujian ini dilakukan sama seperti pengujian sebelumnya, hanya saja untuk pengujian ini digunakan metode CBQ sebagai pengendali trafik jaringan. Pengujiannya yaitu sebagai berikut:

1. *Delay*



Gambar 10 Grafik *Delay* Terhadap Jumlah *Client*

Dari Gambar 11 nilai *delay* memiliki rata-rata 1 ms. Berdasarkan tabel versi THIPON maka kategori delay tersebut termasuk termasuk kategori *delay* sangat bagus karena besar *delay* masih dibawah 150 *ms*.

Dari Gambar 12 dan berdasarkan nilai *packet loss* sesuai dengan versi TIPHON sebagai standarisasi, hasil pengukuran setiap client yang mengakses aplikasi pada jaringan lokal termasuk kategori degredasi sangat bagus.

2. *Packet loss*

Tabel 7 Hasil pengukuran *Packet Loss*

Jumlah *Client*	*Packet Loss* (%)
1	0
2	0
3	0

Dari tabel di atas dan berdasarkan nilai *packet loss* sesuai dengan versi TIPHON sebagai standarisasi, hasil pengukuran setiap client yang mengakses aplikasi pada jaringan lokal termasuk kategori degredasi sangat bagus.

3. *Throughput*

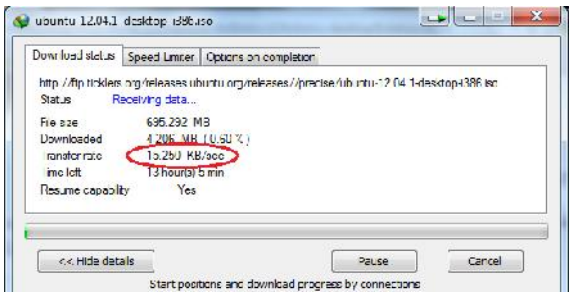


Gambar 11. Grafik *Throughput* Terhadap Jumlah *Client.*

Berdasarkan Gambar 11, dapat disimpulkan bahwa nilai rata-rata *throughput* tertinggi ketika jumlah *client* 1 dan nilai terendah ketika jumlah *client* 3. Hasil tersebut berdasarkan pembatasan *Bandwidth* pada setiap client dengan menggunakan CBQ.

5. Pengukuran *Bandwidth*

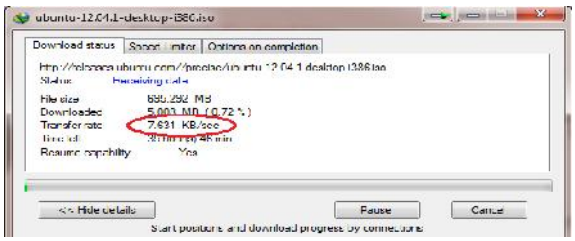
1) Pengukuran pada *Client* 1



Gambar 12 Pengamatan *Bandwidth* Pada client 2

Pada gambar 12 dapat dilihat kecepatan transfer pada saat *download* hanya mencapai 15,250 KB/sec. kecepatan tersebut sesuai dengan batasan *Bandwidth* yang telah dialokasikan sebelumnya

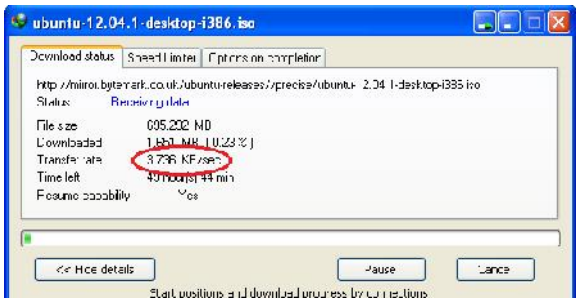
2) Pengukuran pada *Client* 2



Gambar 13. Pengamatan *Bandwidth* Pada *Client* 2

Pada gambar 13 dapat dilihat kecepatan transfer pada saat *download* hanya mencapai 7,631 KB/sec. kecepatan tersebut sesuai dengan batasan *Bandwidth* yang telah dialokasikan sebelumnya.

3) Pengukuran pada *Client* 3



Gambar 14 Pengamatan *Bandwidth* Pada *Client* 3

Pada gambar 14 dapat dilihat rata-rata kecepatan transfer pada saat *download* hanya mencapai 3,736 KB/sec. kecepatan tersebut sesuai dengan batasan *Bandwidth* yang telah dialokasikan sebelumnya.

Berdasarkan pengujian, berikut ini adalah data yang diperoleh dari hasil pengukuran terhadap *bandwidth* dituliskan dalam bentuk tabel 12.

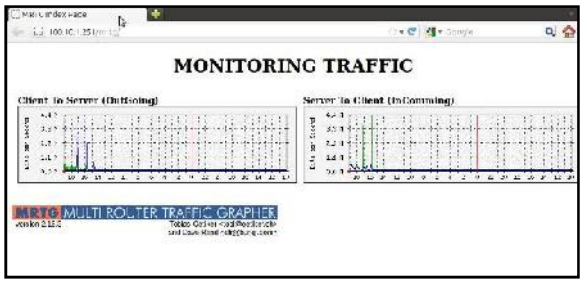
Tabel 8 Hasil Pengukuran Nilai *Bandwidth* Dengan CBQ

Nama	***Bandwidth* (Kbps)**	***Transfer rate* (KB/sec)**
Client 1	128	15,250
Client 2	64	7,631
Client 3	32	3,736

Dari hasil pengukuran dalam tabel 8 dapat dilihat bahwa setiap *client* memiliki *transfer rate* sesuai dengan batasan *bandwidth* yang telah dialokasikan.

4.2 Monitoring Trafik Dengan MRTG

untuk menampilkan hasil monitoring trafik pada jaringan, dapat dilihat dengan cara mengetikan alamat URL *100.10.1.251/mrtg* pada *webrowser*. Pada gambar 18 merupakan grafik monitoring.



Gambar 18. Monitoring Trafik Dengan MRTG

Pada gambar 18 adalah tampilan dari monitoring trafik menggunakan MRTG. Dari gambar tersebut dapat dilihat trafik yang terjadi dalam jaringan. Diantaranya yaitu trafik dari *client* ke server *(Outgoing)* dan trafik dari server ke *client (Incoming).*

4.4 ANALISA HASIL PENGUJIAN

Berdasarkan pengujian, analisa dari hasil pengujian tersebut yaitu sebagai berikut:

1. Pengamatan nilai *Delay*. Dari hasil pengujian, dapat diketahui bahwa rata-rata nilai *delay* yang diperoleh adalah 1 ms. Berdasarkan kategori degredasi untuk QoS versi THIPON, nilai *delay* <150 ms termasuk kedalam kategori sangat bagus sedangkan untuk *delay* >150 ms termasuk pada kategori jelek. dari hasil pengamatan tersebut, nilai *delay* yag di peroleh adalah <150 ms yang merupakan termasuk kedalam kategori sangat bagus.
2. Pengamatan nilai *Packet Loss*. Dari hasil pengujian diketahui nilai *packet Loss* yang dihasilkan sebesar 0%. Berdasarkan kategori degredasi untuk *packet Loss* versi THIPON, kategori degredasi untuk *Packet Loss sebesar* 0% adalah sangat bagus, dan jika melebihi dari 25% maka termasuk kedalam kategori jelek.
3. Pengamatan nilai *Throughput*. Dari hasil pengujian, nilai *throughput* yang diperoleh sebelum manggunakan CBQ dan setelah menggunakan CBQ adalah berbeda. Hal ini disebabkan karena dengan manggunakan CBQ terjadi peroses pembagian *bandwidth* pada setiap *client*. Dimana masing-masing *client* memperoleh *bandwidth* yang telah dialokasikan sebelumnya.
4. Pengamatan nilai *Bandwidth.*
   Dari hasil pengujian dengan melakukan proses *download*, *bandwidth* yang dihasilkan oleh setiap client sesuai dengan batasan yang diberikan. Pada *client* 1 diberikan batasan *bandwidth* sebesar 128 Kbps, dan *transfer rate* yang diperoleh mencapai 15,250 KB/sec, dimana maksimal *transfer rate* pada *client* 1 hanya sebesar 16 KB/sec. Pada *client* 2, diberikan batasan *bandwidth* sebesar 64 Kbps, dan *transfer rate* yang diperoleh mencapai 7,631 KB/sec, dimana maksimal *transfer rate* pada *client* 2 hanya sebesar 8 KB/sec. sedangkan untuk *client* 3, diberikan batasan *bandwidth* sebesar 32 Kbps, dan *transfer rate* yang diperoleh mencapai 3,736 KB/sec, dimana maksimal *transfer rate* pada *client* 1 hanya sebesar 4 KB/sec.

Dari hasil pengukuran tersebut, rata-rata secara keseluruhan nilai yang diperoleh pada setiap parameter dituliskan dalam bentuk tabel. Pada tabel 4.9 merupakan perbandingan hasil pengukuran tanpa CBQ dan menggunakan CBQ.

Tabel 4.9 Perbandingan Hasil Pengukuran

No	Parameter QoS	Tanpa CBQ	Dengan CBQ	Standar THIPON
1	*Delay*	1,14 ms	1,13 ms	Sangat Bagus
2	*Packet Loss*	0 %	0 %	Sangat Bagus
3	*Throughput*	0.057 Mbps	0.072 Mbps	-
4	*Bandwidth*	30,90 Kbps	8,87 Kbps	-

Berdasarkan tabel 4.9 dapat dilihat perbandingan hasil pengukuran dari setiap parameter QoS. Dimana dari hasil pengukuran terhadap parameter *Delay*, dan *Throughput* tersebut nilai yang diperoleh tanpa menggunakan metode CBQ dan dengan

menggunakan metode CBQ tidak memiliki perbedaan yang signifikan, begitu juga dengan nilai *Packet loss* yang tidak terlihat. Hal tersebut disebabkan karena pengujian dilakukan pada jaringan lokal, dimana server berada pada jaringan lokal, serta ukuran paket data yang diakses sangat kecil, sehingga trafik yang terjadi pada jaringan tidak terlalu padat.

5. KESIMPILAN DAN SARAN

5.1 Kesimpulan

Berdasarkan pengujian yang telah dilakukan**,** maka diambil kesimpulan sebagai berikut :

1. Berdasarkan hasil pengujian, layanan *cloud computing* bisa dikatakan baik untuk digunakan oleh pelanggan, karena tidak memiliki jumlah *delay*, *packet loss* dan *throughput* yang besar. Untuk penggunaan *Bandwidth* dengan CBQ memiliki nilai rata-rata yang kecil, hal ini cukup baik karena dapat memberikan kesempatan pada pengguna lain untuk mengakses jaringan atau menggunakan layanan *cloud computing.*
2. Dengan melakukan manajemen trafik dan *bandwidth*, dapat memberikan prioritas terhadap aplikasi-aplikasi yang kritis pada jaringan *cloud computing* seperti aplikasi server dan memaksimalkan penggunaan investasi pada jaringan seperti memaksimalkan pemakaian *bandwidth*

5.2 Saran

Saran yang diajukan supaya menjadi masukan untuk pengembangan selanjutnya yaitu sebagai berikut:

1. Pengujian ini dilakukan pada jaringan skala kecil dengan jumlah dengan jumlah *client* sebanyak 3 *client*. Untuk penelitian selanjutnya bisa diimplementasikan pada jaringan yang lebih luas dengan lebih banyak *client.*
2. Pada penelitian selanjutnya, diharapkan metode CBQ yang digunakan berbasis web untuk mempermudah melakukan pengaturannya.

DAFTAR PUSTAKA

[1] Anggeriana, H., & Susilo, y. V. *Cloud Computing.*

[2] Gerald Kaefer Corporate Research and Technologies. (n.d.). Cloud Computing Architecture. Munich, Germany.

[3] Sofana, I. (2008). *Membangun Jaringan Komputer.* Bandung: Informatika Bandung.

[4] Akmal, A., Susanti, F., & Iskandar, M. I. (n.d.). Konfigurasi dan Analisis Manajemen Bandwidth pada PC Router Menggunakan Metode HTB (Hierarchy Token Bucket) dan CBQ (Class Based Queue). 2-4.

[5] Abdillah, M. R., Darwiyanto, E., & Iskandar, M. I. (n.d.). Implementasi High Availability Mail Server dengan Menggunakan Postfix, Heartbeat, dan DRBD. 3-4.

[6] Rabah, K. (2010). Global Open Versity. In *Build & Deploy Desktop Personal Cloud using eyeOS Cloud Computing Operating System on Linux* (p. 1). Canada.

[7] Mansfield, N. (2004). *Pratctical TCP/IP : mendesain, menggnakan, dan Troubleshooting Jaringan TCP/IP di Linux dan Windows* (1 ed.). (D. Prabantini, Trans.) Yogyakarta: ANDI.

[8] CISCO. Quality of Service Networking. In *Internetworking Technologies Handbook* (pp. 49-1).

[9] European Telecommunications Standards Institute (ETSI). (1998). Telecommunications and Internet Protocol

[10] Harmonization Over Networks (TIPHON). In *General aspects of Quality of Service (QoS)* (pp. 24-27). FRANCE.

[11] Wahono, R. S. (2003). In *Cepat Mahir Linux* (p. 1). Ilmukomputer.com.