

# BAB I

## PENDAHULUAN

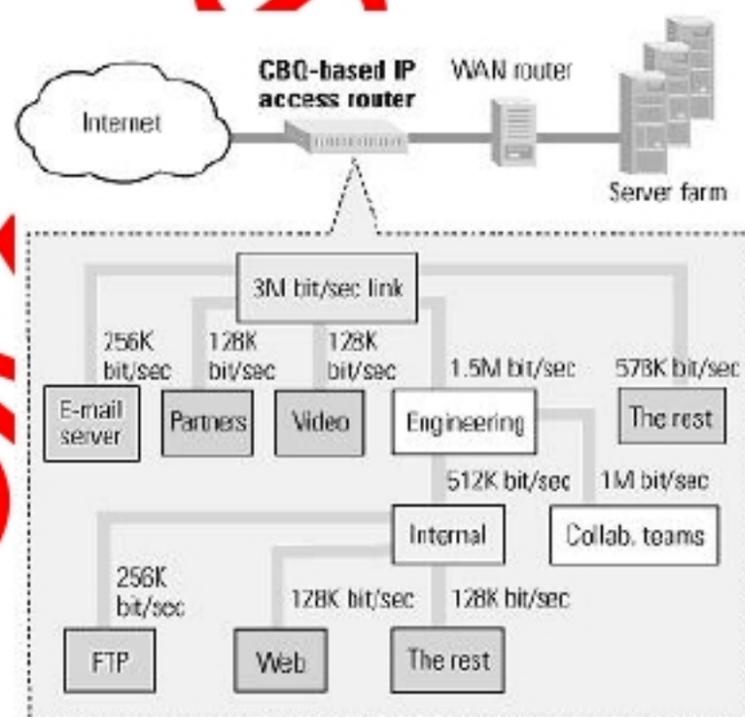
### 1.1 Latar Belakang Masalah

Perkembangan dunia komunikasi dan teknologi informasi dewasa ini, mengakibatkan lalu lintas aliran data di internet menjadi semakin bertambah dari hari ke hari, hal ini disebabkan meningkatnya jumlah para pengguna internet. Salah satu solusi yang paling efektif untuk mengatasinya adalah dengan mengelola pemakaian *bandwidth* sehingga menghasilkan suatu kualitas layanan lalu lintas aliran data yang baik dan berkualitas. (Bogi, Nyoman. Ardiansyah, Andi. Pratania, Eka, 2003:1)

Santosa, Budi (2003:6) menjelaskan bahwa “kualitas layanan aliran data pada jaringan komputer atau *QoS (Quality of Service)* dapat didefinisikan sebagai suatu komitmen dari semua elemen atau unsur jaringan komputer untuk memberikan pelayanan yang berkualitas dan memuaskan kepada para pengguna jaringan komputer, dengan kata lain memenuhi dan memberikan kepuasan terhadap permintaan dan kebutuhan pengguna jaringan dan menyediakan layanan jaringan yang nyata dan jelas kepada para pengguna jaringan”.

Ada beberapa metode yang sudah diaplikasikan untuk merepresentasikan konsep *QoS* pada sistem operasi Linux salah satunya adalah *CBQ (Class Based Queueing)*. *CBQ* diaplikasikan kedalam bentuk script (*cbq.init*) yang dikontrol secara langsung oleh *kernel* linux dan terintegrasi dengan beberapa algoritma antara lain algoritma *link sharing*, *classification*, dan *shaping*”. (Bogi, Nyoman. Ardiansyah, Andi. Pratania, Eka, 2003:2)

Shultz, Brian (2000:3) menjelaskan bahwa *CBQ script* berkerja dengan membatasi, membagi dan mengatur jumlah *bandwidth* pada masing-masing layanan yang disediakan oleh jaringan komputer, seperti pada gambar 1.1 dimana sebuah jaringan komputer yang memiliki *bandwidth* sebesar 3 Mbps (*Mega bit per second*) terdapat CBQ yang terintegrasi dengan IP (*Internet Protocol*) *Access Router* yang bertugas untuk membatasi dan mengatur pembagian *bandwidth* pada beberapa layanan jaringan yang disediakan antara lain untuk layanan Web sebesar 128 Kbps (*Kilo bit per second*), layanan e-mail dan FTP (*File Transfer Protocol*) masing-masing sebesar 256 Kbps, dan layanan-layanan lain seperti video serta pembagian *bandwidth* untuk kelompok pengguna seperti *internal*, *collabs team* dan *the rest*.



Gambar 1.1 Implementasi CBQ pada Jaringan Komputer

## 1.2 Perumusan Masalah

Dari latar belakang masalah yang telah diuraikan di atas, maka rumusan masalahnya adalah :

1. Bagaimana melakukan pembagian bandwidth untuk masing-masing layanan (*services*) yang disediakan oleh jaringan komputer yang terhubung ke internet kepada seluruh pengguna baik user maupun kelompok user sesuai dengan tingkat kebutuhan layanan yang digunakan dengan menerapkan salah satu metode dari konsep *QoS (Quality of Service) Class Based Queueing (CBQ)* pada sistem operasi Linux berbasis web
2. Bagaimana melakukan *monitoring* (pengontrolan) pemakaian bandwidth agar dapat dilakukan pembatasan pemakaian dari masing-masing layanan yang dipakai oleh user pada komputer *client* yang berada didalam lingkup jaringan komputer.

## 1.3 Batasan Masalah

Sesuai dengan latar belakang dan perumusan masalah yang telah diuraikan di atas maka pembatasan masalahnya adalah sebagai berikut :

1. Sistem ini dibuat dengan mengintegrasikan beberapa *software* yang mempunyai *open source license* dan berjalan pada sistem operasi Linux antara lain *cbq.init script (cbq.init-v0.7.3)* sebagai *tool* utama yang digunakan, *squid proxy* dengan fitur *delay pools*-nya untuk meningkatkan kinerja performa layanan HTTP (*Hyper Text Transfer Protocol*), *iptables* yang berfungsi sebagai *firewall*, MySQL sebagai tempat penyimpanan data konfigurasi dan data *history*, serta Perl-CGI dan PHP sebagai bahasa pemrogramannya.

2. Penelitian juga membahas algoritma-algoritma yang terdapat pada konsep dasar CBQ (*Class Based Queueing*). Algoritma yang dibahas antara lain algoritma *Shaping*, algoritma *Link Sharing*, dan algoritma *Classification*.
3. Distribusi (*distro*) serta versi dari sistem operasi linux yang digunakan adalah Linux RedHat versi 9.0.

#### 1.4 Tujuan

Berdasarkan rumusan masalah serta batasan masalah yang telah disampaikan pada sub bab sebelumnya, penulis menetapkan tujuan sebagai berikut :

1. Memudahkan pengelola jaringan untuk membagi pemakaian *bandwidth* sesuai dengan kebutuhan layanan (*service*) yang digunakan oleh user atau kelompok user dengan menerapkan salah satu metode dari konsep QoS (*Quality of Service*) yakni *Class Based Queueing* (CBQ).
2. Memungkinkan pengelola jaringan untuk melakukan *monitoring* (pengontrolan) pemakaian *bandwidth* pada masing-masing layanan yang digunakan oleh user pada komputer *client* yang berada pada lingkup jaringan komputer.

#### 1.5 Sistematika Penulisan

Di dalam penulisan Tugas Akhir ini secara sistematika diatur dalam lima bab, yaitu :

##### **BAB I : PENDAHULUAN**

Pada bab ini diuraikan secara umum tentang latar belakang masalah, tujuan, perumusan masalah yang menjelaskan mengapa aplikasi sistem

ini dibuat, serta sistematika penulisan tugas akhir sebagai ringkasan materi dari masing-masing bab.

## **BAB II : LANDASAN TEORI**

Bab ini menjelaskan konsep-konsep dan teori-teori yang mendukung penyelesaian tugas akhir ini antara lain definisi tentang jaringan komputer, model dan arsitektur dari TCP/IP, konsep dasar dan definisi dari disiplin antrian, mekanisme konsep serta algoritma-algoritma yang terdapat pada metode *Class Based Queueing* (CBQ), Kernel Linux, Squid, Iptables, Apache web server, OpenSSL, dan landasan teori tentang pembuatan web antara lain CGI-Perl, PHP, dan MySQL.

## **BAB III : PERANCANGAN SISTEM**

Dalam Bab ini akan dijelaskan tentang tahap-tahap yang akan dikerjakan dalam proses perancangan sistem dari Tugas Akhir ini. Dimulai dari analisa permasalahan dengan sub bab proses perancangan sistem, perancangan Sistem Flow, perancangan Data Flow Diagram, pembuatan Entity Realtional Diagram, Struktur Database, dan Desain Input Ouput dari sistem.

## **BAB IV : IMPLEMENTASI SISTEM**

Bab ini membahas tentang kebutuhan perangkat lunak yang diperlukan untuk melakukan instalasi aplikasi ini, dan langkah-langkah proses instalasi dari aplikasi ini. Selanjutnya pada bab ini dijelaskan bagaimana implementasi dari aplikasi ini dimana terdapat tampilan-tampilan interface dari aplikasi ini dan hasil dari testing dan evaluasi dari aplikasi ini.

**BAB V : PENUTUP**

Bab ini membahas kesimpulan yang didapat dari pembuatan tugas akhir ini secara keseluruhan termasuk hasil testing dan evaluasi dan saran-saran yang diharapkan terhadap pengembangan dari karya yang ada serta kemungkinan-kemungkinan dilakukannya pengembangan dan perbaikan apabila masih terdapat kesalahan yang telah dilakukan.

STIKOMMP SURABAYA