Application-Level Quality of Service Management System

Project code: TD2-02

Supervisor: Dr. Danny H.K. Tsang

Members: Cheng Kwok Wing (00242892)
Kwan Kam Fai (00030316)
Kwok Chi Wai (00281264)
As the bandwidth available increases, Internet applications become more and more bandwidth demanding. Sometimes the performance of a network can be ruined by certain programs, which occupy almost all the bandwidth. These programs include peer-to-peer file sharing program, which will consume as much bandwidth as it can get. The bandwidth available is limited after all and other network applications, like database connection and web browsing, will suffer.

A solution to this problem is to classify the traffics passing through the network and treat them differently. Different classes of traffics are placed in queues of different priorities. Mission-critical and real-time traffics will be delivered with a higher priority than elastic traffics. The traffic classification and shaping are implemented in a packet-forwarding device in the network.

We built an Application-Level Quality of Service Management System. The whole system is a gateway running under Linux using tc as the traffic controller. In the system, the network administrator can assign certain bandwidth to each user in the network, and users can then allocate their shares of bandwidth to different applications by their needs. To make the Bandwidth management system easy to use, we built a web-based interface for configuration. To make evaluation on bandwidth allocation possible, we presented the network usage statistics on different classes of traffic using graphs dynamically.

In addition, we implemented Credit-based fair queue (CBFQ), which is a packet-scheduling algorithm similar to PGPS, as a module in Linux kernel. CBFQ solves the problems of infinite increment of time tags and heavy load of the system by using credit counters.
Step 1: create account for clients in a Local Area Network (LAN) and assign workstations to the clients.

Step 2: decide the bandwidth usage of clients and apply the policies to interfaces: inbound, outbound or both.

Step 3: specify the allocated bandwidth, ceiling and priority of the client.

Step 4: specify the bandwidth usage of applications for the client.

Step 5: check the statistics such as the bandwidth usage, packet rate, number of packets dropped and more.