

# MODUL 6

## MANAJEMEN BANDWIDTH DENGAN CBQ DAN HTB

### TUJUAN PEMBELAJARAN:

Setelah melaksanakan praktikum ini, mahasiswa diharapkan :

1. Mengerti dan memahami konsep manajemen *bandwidth*.
2. Mampu melakukan konfigurasi *static routing*.
3. Mampu melakukan instalasi dan konfigurasi *network traffic shapping*.

### DASAR TEORI

Akhir-akhir ini semakin berkembang layanan internet berkecepatan tinggi yang disediakan oleh ISP, mulai dari 256 kbps hingga 2 Mbps. Pada umumnya layanan ini digunakan oleh penyedia layanan internet (warung internet, dsb) serta perusahaan-perusahaan yang bergerak dibidang internet. *Bandwidth* sebesar itu dapat dimanfaatkan bersama-sama secara adil. Pada praktikum ini akan dibahas mengenai pembuatan sistem manajemen *bandwidth* yang dapat diimplementasikan pada jaringan yang telah ada.

#### a. CBQ

Teknik klasifikasi paket data yang paling terkenal adalah CBQ. Keunggulan dalam penggunaan CBQ adalah mudah dikonfigurasi, memungkinkan *sharing bandwidth* antar kelas (*class*) dan memiliki fasilitas *user interface*. CBQ mengatur pemakaian *bandwidth* jaringan yang dialokasikan untuk tiap *user*, dan pemakaian *bandwidth* yang melebihi nilai set akan dipotong (*shaping*). CBQ juga dapat diatur untuk *sharing* dan meminjam *bandwidth* antar *class* jika diperlukan.

*Class Based Queueing* (CBQ) adalah suatu mekanisme penjadwalan, bertujuan menyediakan *link sharing* antar kelas yang menggunakan jalur fisik yang sama, sebagai acuan untuk membedakan trafik yang memiliki prioritas-prioritas yang berlainan. Dengan CBQ, setiap kelas dapat mengalokasikan *bandwidth* miliknya untuk berbagai jenis trafik yang berbeda, sesuai dengan pembagiannya yang tepat untuk masing-masing trafik.

Pada jenis antrian CBQ, mempunyai beberapa parameter yang digunakan yaitu :

- **avpkt**, jumlah paket rata – rata saat pengiriman
- **bandwidth**, lebar *bandwidth* kartu *ethernet* biasanya 10 – 100Mbit
- **rate**, kecepatan rata – rata paket data saat meninggalkan *qdisc*, ini parameter untuk men-set *bandwidth*.
- **cell**, peningkatan paket data yang dikeluarkan ke kartu *ethernet* berdasarkan jumlah *byte*, misalnya 800 ke 808 dengan nilai *cell* 8.
- **isolated / sharing**, parameter *isolated* mengatur agar *bandwidth* tidak bisa dipinjam oleh klas (*class*) lain yang sama tingkatannya / *sibling*. Parameter *sharing* menunjukkan *bandwidth* kelas (*class*) bisa dipinjam oleh kelas lain.
- **bounded / borrow**, parameter *borrow* berarti kelas (*class*) dapat meminjam *bandwidth* dari klas lain, sedangkan *bounded* berarti sebaliknya.

#### b. HTB

*Hierarchical Token Bucket* (HTB) merupakan jenis aplikasi yang dikembangkan oleh Martin Devera pada tahun 2001 yang digunakan untuk membatasi akses menuju ke port/IP tertentu tanpa mengganggu trafik *bandwidth* pengguna lain. Aplikasi ini berfungsi sebagai pengganti aplikasi yang masih sering digunakan, yaitu CBQ. HTB diklaim mampu melakukan pembagian trafik yang lebih akurat.

Teknik antrian HTB mirip dengan teknik pada CBQ. Hanya perbedaannya terletak pada opsi, dimana pada HTB opsi yang digunakan jauh lebih sedikit dalam konfigurasinya, serta lebih presisi dalam penggunaannya. Teknik antrian HTB memberikan fasilitas pembatasan trafik pada

setiap level ataupun klasifikasinya, sehingga *bandwidth* yang tidak terpakai dapat digunakan oleh klasifikasi lain yang lebih rendah.

Pada antrian HTB mempunyai parameter yang menyusunnya dalam antrian yaitu :

1. Rate

Parameter *rate* menentukan *bandwidth* maksimum yang bisa digunakan oleh setiap *class*, jika *bandwidth* melebihi nilai “*rate*”, maka paket data akan dipotong atau dijatuhkan (*drop*).

2. Ceil

Parameter *ceil* di-set untuk menentukan peminjaman *bandwidth* antar *class* (kelas), peminjaman *bandwidth* dilakukan kelas paling bawah ke kelas di atasnya. Teknik ini disebut *link sharing*.

3. Random Early Detection (RED)

*Random Early Detection* atau bisa disebut *Random Early Drop* biasanya digunakan untuk *gateway/router backbone* dengan tingkat trafik yang sangat tinggi. RED mengendalikan trafik jaringan sehingga terhindar dari kemacetan pada saat trafik tinggi berdasarkan pemantauan perubahan nilai antrian minimum dan maksimum. Jika isi antrian dibawah nilai minimum, maka mode ‘*drop*’ tidak berlaku, saat antrian mulai terisi hingga melebihi nilai maksimum, maka RED akan membuang (*drop*) paket data secara acak sehingga kemacetan pada jaringan dapat dihindari. Pada antrian RED juga mempunyai parameter yang menyusunnya, yaitu :

a. Min

Yaitu nilai rata-rata minimum antrian (*queue*).

b. Max

Nilai rata-rata maksimum antrian, biasanya dua kali nilai minimum atau dengan rumus :

$$\text{Max} = \text{bandwidth (bps)} * \text{latency (s)}$$

c. Probability

Jumlah maksimum probabilitas penandaan paket data. Nilainya berkisar antara 0.0 sampai dengan 1.0.

d. Limit

Batas paling atas antrian secara riil, jumlah paket data yang melewati *limit* pasti dibuang. Nilai limit harus lebih besar daripada ‘*max*’ dan dinyatakan dengan persamaan :

$$\text{limit} = \text{max} + \text{burst}$$

e. Burst

Digunakan untuk menentukan kecepatan perhitungan nilai antrian mempengaruhi antrian riil (*limit*). Bisa dihitung dengan persamaan :

$$\text{Burst} = (\text{min} + \text{min} + \text{max}) / 3 * \text{avpkt}$$

f. Avpkt

Nilai rata – rata paket data/grafik yang melintasi *gateway* RED, sebaiknya diisi 1000.

g. Bandwidth

Yaitu lebar *bandwidth* kartu Ethernet.

h. Ecn (*Explicit Congestion Notification*)

Parameter ini memberikan fasilitas *gateway* RED untuk memberitahukan kepada *client* jika terjadi kemacetan.

PERALATAN :

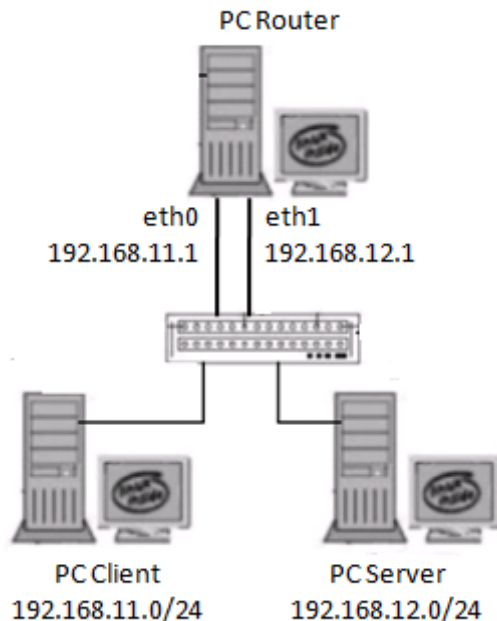
- Sebuah komputer sebagai *server*, dengan 1 NIC Card
- Sebuah komputer sebagai *client*, dengan 1 NIC Card
- Sebuah komputer sebagai router, dengan 2 NIC Card
- *Hub/switch* sebagai penghubung jaringan
- Kabel jaringan secukupnya

## TUGAS PENDAHULUAN

1. Apa tujuan dengan adanya bandwidth management ?
2. Jelaskan perbedaan yang paling mendasar antara CBQ dan HTB ?

## PERCOBAAN

Bangun jaringan sederhana seperti pada gambar berikut :



Gambar 1. Konfigurasi Jaringan

NB :

Aplikasi bandwidth management akan dipasang di PC Router.

### A. Setting pada perangkat jaringan

1. Pada PC Router, setting sebagai berikut :
  - a. Setting dengan IP Aliasing pada eth0 dan eth1

```
# ifconfig eth0 192.168.11.1 netmask 255.255.255.0
# ifconfig eth1 192.168.12.1 netmask 255.255.255.0
```
  - b. Aktifkan ip\_forward pada PC Router untuk melakukan proses routing :

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
```
2. Setting pada PC Server
  - a. Setting IP

```
# ifconfig eth1 192.168.12.2 netmask 255.255.255.0
# route add -net default gw 192.168.12.1
```
  - b. Install aplikasi FTP server

```
# apt-get install proftpd
```
3. Setting pada PC Client

```
# ifconfig eth0 192.168.11.2 netmask 255.255.255.0
# route add -net default gw 192.168.11.1
```
4. Lakukan tes koneksi dengan ftp dari PC Client

```
# ftp 192.168.12.2
ftp> get file1Mb          => untuk download dari server ke client
ftp> put file1Mb         => untuk upload dari client ke server
```

Catat hasilnya untuk waktu akses dan throughputnya seperti contoh di bawah ini.

```
226 Transfer complete
911455 bytes received in 5.33 secs (167.1 kB/s)
ftp> █
```

## B. Bandwidth Management dengan CBQ

### 1. Lakukan instalasi aplikasi CBQ

- a. Download aplikasi CBQ  
cbq.init-v0.7.1
- b. Pindahkan file cbq.init ke direktori /sbin/  
# mv cbq.init-v0.7.1 /sbin/cbq.init
- c. Ubah mode kepemilikan file cbq.init  
# chmod 755 /sbin/cbq
- d. Buat directory CBQ  
# mkdir /etc/sysconfig/cbq
- e. Buat aturan / rule di CBQ yang membatasi download di sisi client hanya sebesar 64Kbps  
# vim /etc/sysconfig/cbq/cbq-028.luar-dalam

```
DEVICE=eth0,100Mbit,10Mbit
RATE=64Kbit
WEIGHT=6Kbit
PRIO=5
LEAF=sbq
RULE=192.168.11.0/24
```

- f. Buat aturan / rule di CBQ yang membatasi download di sisi client hanya sebesar 64Kbps  
# vim /etc/sysconfig/cbq/cbq-040.dalam-luar

```
DEVICE=eth2,100Mbit,10Mbit
RATE=128Kbit
WEIGHT=12Kbit
PRIO=5
LEAF=sbq
RULE=192.168.20.0/24
```

- g. Masuk ke /sbin, ketikkan :

```
# cbq.init
# cbq.init compile
# cbq.init start
# cbq.init stats => untuk melihat statistik
```

- h. Lakukan cbq.init compile dan start setiap kali melakukan perubahan konfigurasi.

### 2. Lakukan pengetestan seperti langkah A.4 dan bandingkan hasilnya.

3. Lakukan pengambilan data dengan menggunakan ftp, hanya untuk proses download saja. Catat hasilnya.

Bandwidth	Waktu (s)	Throughput (Kbps)
64Kbps		
128Kbps		
256Kbps		

## C. Bandwidth Management dengan HTB

### 1. Instalasi aplikasi HTB

#### a. Download paket HTB

```
HTB-tools-0.3.0a-i486-1.tgz
```

#### b. Ekstrak file tersebut

```
# tar -zxvf HTB-tools-0.3.0a-i486-1.tgz -C /usr/local/src
```

#### c. Pindahkan semua hasil ekstrak ke folder aslinya

```
# cp /usr/local/src/sbin/* /sbin
# cp /usr/local/src/usr/* /usr
# mkdir /etc/htb
# cp /usr/local/src/etc/htb/* /etc/htb
# cp /usr/local/src/etc/rc.d/rc.htb.new /etc/init.d/rc.htb
# chmod 755 /etc/init.d/rc.htb
```

#### d. Copykan file konfigurasi berikut ini:

```
# cd /etc/htb/
# mv eth0-qos.cfg.new eth0-qos.cfg
# mv eth1-qos.cfg.new eth1-qos.cfg
```

#### e. Konfigurasi HTB Untuk proses download dari server ke client : eth0-qos.cfg

```
# vim /etc/htb/eth0-qos.cfg
```

```
class class_1 {
    bandwidth 10000;
    limit 10000;
    burst 2;
    priority 1;

    client client_1 {
        bandwidth 64;
        limit 128;
        burst 2;
        priority 1;
        dst {
            192.168.11.0/24;
        };
    };
};
class default { bandwidth 8; };
```

#### f. Untuk proses upload dari client ke server : eth1-qos.cfg

```
# vim /etc/htb/eth0-qos.cfg
```

```
class class_1 {
    bandwidth 10000;
    limit 10000;
    burst 2;
    priority 1;

    client client_1 {
        bandwidth 256;
        limit 512;
        burst 2;
        priority 1;
        src {
            192.168.12.0/24;
        };
    };
};
class default { bandwidth 8; };
```

- g. Untuk menjalankan HTB, masuk ke /etc/htb, jalankan perintah berikut :
- ```
# /etc/init.d/rc.htb stop
# /etc/init.d/rc.htb start
# /etc/init.d/rc.htb stats      => untuk melihat statistik
```

Atau

```
# /sbin/htb eth1 stop
# /sbin/htb eth1 start
```

2. Lakukan pengetesan seperti langkah A.4 dan bandingkan hasilnya.
3. Lakukan pengambilan data dengan menggunakan ftp, hanya untuk proses download saja. Catat hasilnya.

| <b>Bandwidth</b> | <b>Waktu (s)</b> | <b>Throughput (Kbps)</b> |
|------------------|------------------|--------------------------|
| 64Kbps           |                  |                          |
| 128Kbps          |                  |                          |
| 256Kbps          |                  |                          |

Bandingkan hasilnya dengan langkah B.3.

## LAPORAN RESMI

### Daftar Pertanyaan

1. Berikan kesimpulan hasil praktikum yang anda lakukan.
2. Tugas akan diberikan pada waktu praktikum.