

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

Penelitian Fathoni (2011) mengenai “Analisis Kualitas Layanan Jaringan Intranet (Studi Kasus : Universitas Bina Darma)”. Kesimpulannya dari hasil survey yang dilakukan dengan mengambil 30 sampel bahwa QoS jaringan intranet cukup baik dengan total Qos rata-rata 3.67.

Fahrul Agus, dkk. (2010) melakukan studi tentang Optimalisasi Manajemen Bandwidth Pada Jaringan Intranet Universitas Mulawarman.

Berkembangnya jaringan LAN dan jaringan internet di Universitas Mulawarman yang semakin membesar membutuhkan adanya suatu system manajemen jaringan yang mengatur dan mengidentifikasi kebutuhan akan system komputer. Tujuan dari analisa efektifitas penggunaan bandwidth jaringan komputer di Universitas Mulawarman adalah untuk mengidentifikasi performa dan jumlah traffic yang terjadi setiap harinya, baik pada jaringan LAN maupun pada jaringan koneksi internet. Metode penelitian yang digunakan adalah pengamatan terhadap backbone jaringan komputer Universitas Mulawarman. Identifikasi serta analisa mengenai performa dan total traffic. Dari hasil pengamatan didapat bahwa dari dua jaringan yang ada, yaitu jaringan akses internet dan jaringan intranet, jaringan intranet tidak dimanfaatkan secara optimal sebagai jalur akses ke server Universitas Mulawarman yang seharusnya bisa diakses secara lokal. Pada penelitian ini menghasilkan beberapa topologi baru yang dapat mengatasi lambatnya koneksi

akses ke server lokal Universitas Mulawarman serta pembatasan bandwidth yang terpusat. Topologi ini diharapkan bisa memenuhi kebutuhan akan akses intranet yang cepat.

Konfigurasi dan Analisis Manajemen Bandwidth pada PC Router Menggunakan Metode HTB (*Hierarchy Token Bucket*) dan CBQ (*Class Based Queue*) Studi Kasus Kantor Badan Pertanahan Nasional Bukittinggi yang ditulis oleh Adrian Akmal, dkk. dari Politeknik Telkom Bandung.

Implementasi *Quality Of Service* Dengan Metode HTB (*Hierarchical Token Bucket*) Pada PT. Komunika Lima Duabelas yang ditulis oleh Yunus Arifin (2013) dari Universitas Udayana. *Hierarchical Token Bucket* merupakan teknik QoS yang mampu memaksimalkan *bandwidth* yang tidak terpakai, sehingga kualitas pelayanan menjadi lebih meningkat, berikut hasil yang dicapai:

- Setiap paket memperoleh *bandwidth* minimal pada CIR (*Committed Information Rates*).
- Setiap paket memperoleh *bandwidth* lebih dari CIR tetapi tidak melebihi MIR (*Maximum Information Rates*), selama traffic pada parentnya tidak penuh.
- Terjadi pemerataan *bandwidth* sesuai prioritasnya saat kondisi *traffic* seluruh paket penuh,

Wilmadi, dkk. (2013) Analisis *Management* Bandwidth Dengan Metode PCQ (*Per Connection Queue*) Dan HTB (*Hierarchical Token Bucket*) Dengan Menggunakan Router Mikrotik. Skripsi thesis, Universitas Muhammadiyah Surakarta. Pada hasil pengujian yang dilakukan didapatkan data *throughput* pada setiap metode tabel memberikan data *throughput* dan *packet loss* saat pengujian

berlangsung dengan menggunakan metode PCQ. Sedangkan pada table memberikan data *throughput* dan *packet loss*.

Aji Nugroho, Setyo (2013) mengenai *Custom Limitation Quality Of Service (QoS) Advance Hotspot* Menggunakan Mikrotik RB751U. Kesimpulan dari perancangan dan pengujian sistem secara keseluruhan :

- Implementasi *Custom Limitation* QoS dilakukan dengan cara membuat kategori koneksi, melakukan penandaan koneksi dan paket pada *Firewall Mangle* dan melakukan pengelompokan limitasi koneksi pada *Queue Tree*.
- Dengan mengimplementasikan *Custom Limitation* QoS, *bandwidth* dapat dibagi menurut kriteria koneksi yang beragam dan disalurkan menyesuaikan 16 dengan banyaknya *user* yang sedang aktif, sesuai dengan *rule* yang telah dibuat pada *Mangle* dan *Queue Tree*.

2.2 Landasan Teori

2.2.1. Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer beserta mekanisme dan prosedurnya yang saling terhubung dan berkomunikasi. Komunikasi yang dilakukan oleh komputer tersebut dapat berupa transfer berbagai data, instruksi, dan informasi dari satu komputer ke komputer lainnya (Ramadhan, 2006 :2).

Agar dapat mencapai tujuan yang sama, setiap bagian dari jaringan komputer meminta dan memberikan layanan (*service*). Pihak yang meminta layanan disebut klien (*client*) dan yang memberikan layanan disebut pelayan

(*server*). Arsitektur ini disebut dengan sistem *client-server*, dan digunakan pada hampir seluruh aplikasi jaringan komputer.

Tujuan utama dari tersedianya jaringan komputer adalah :

a. Membagi sumber daya

User yang saling terhubung dalam satu jaringan yang sama, dapat menggunakan perangkat yang telah terhubung dalam jaringan tersebut.

Contohnya: *Sharing printer, sharing scanner dan sharing mesin fax.*

b. Komunikasi

Dalam satu jaringan yang sama, *user* dapat saling bertukar data tanpa harus menggunakan media perantara seperti *USB drive* ataupun *Harddisk*.

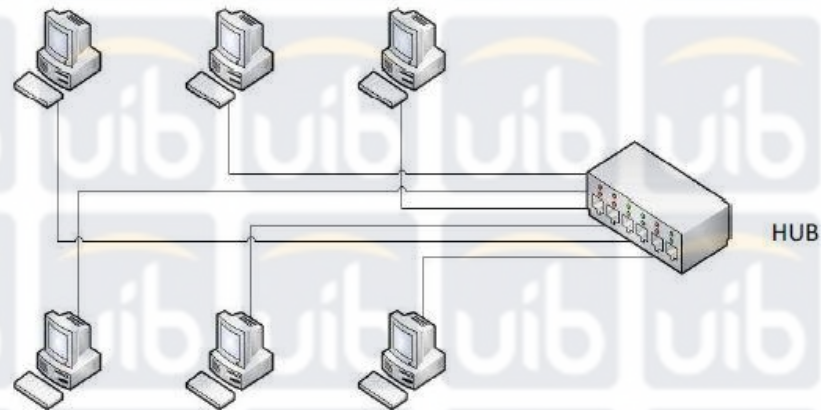
Pembentukan jaringan memiliki salah satu tujuan yaitu dalam komunikasi data, Menurut (Forouzan, 2007, p4) komunikasi data memungkinkan untuk melakukan pertukaran data diantara 2 perangkat atau lebih dengan sebuah media transmisi seperti kabel.

2.2.2. Klasifikasi Jaringan

Berdasarkan jaringnya, jaringan terbagi tiga, yaitu :

A. *Local Area Network* (LAN)

LAN merupakan jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantoran di sebuah gedung, atau sebuah sekolah, dan biasanya tidak jauh dari sekitar 1 km persegi.



Gambar 2.1 *Local Area Network*

B. Metropolitan Area Network (MAN)

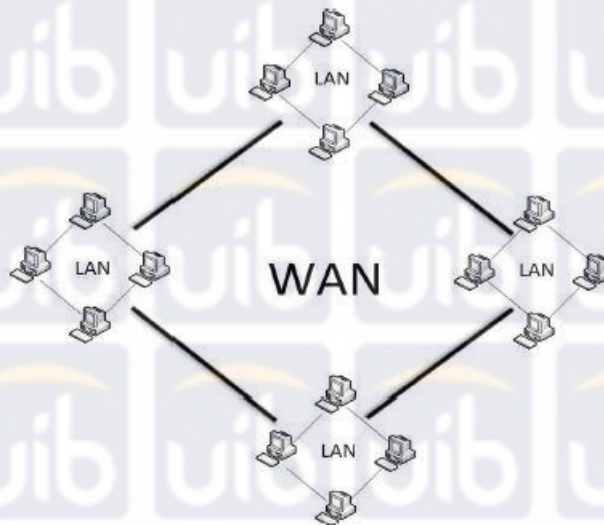
MAN biasanya meliputi area yang lebih besar dari LAN, misalnya antar wilayah dalam satu provinsi. Dalam hal ini jaringan menghubungkan beberapa buah jaringan – jaringan kecil dalam lingkungan area yang lebih besar, sebagai contoh yaitu jaringan Bank dimana beberapa kantor cabang sebuah Bank di dalam sebuah kota besar dihubungkan antar satu dengan lainnya.



Gambar 2.2 *Metropolitan Area Network*

C. *Wide Area Network (WAN)*

WAN adalah jaringan yang lingkungannya biasanya sudah menggunakan sarana satelit ataupun kabel bawah laut. Sebagai contoh keseluruhan jaringan Bank BNI yang ada di Indonesia ataupun yang ada di Negara-negara lain.

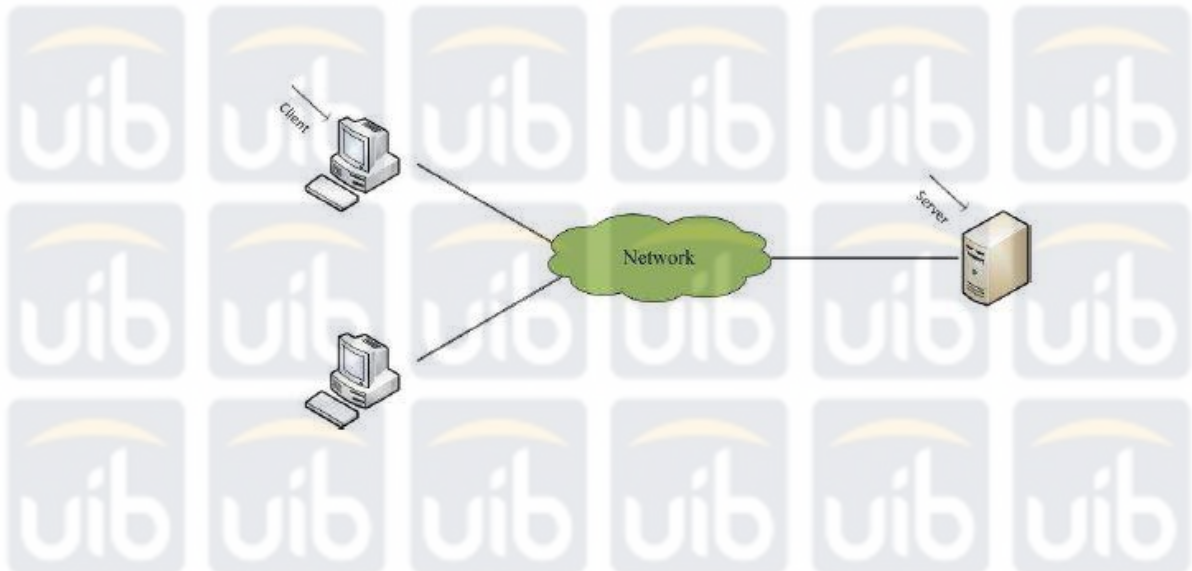


Gambar 2.3 *Wide Area Network*

Berdasarkan tipe pemrosesan data terbagi menjadi dua, yaitu :

A. *Client – Server*

Dalam arsitektur ini, sebuah komputer akan bertindak sebagai *server* yang bertugas melayani *client*. Jumlah *client* yang dilayani pada satu waktu dapat berjumlah lebih dari satu (*multi user*). Saat ini sebagian besar jaringan menggunakan arsitektur *client-server*, namun arsitektur ini memerlukan seorang *administrator* yang *professional* untuk memelihara *server* tersebut.

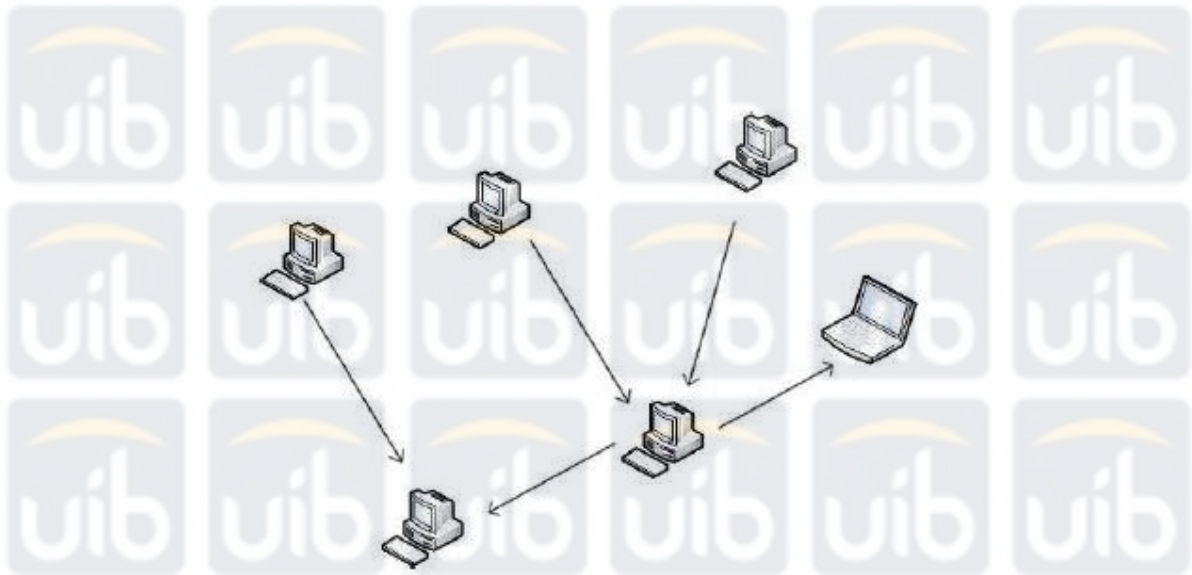


Gambar 2.4 *Client to Server*

B. *Peer to Peer*

Dalam arsitektur ini, tidak ada komputer yang mempunyai tugas khusus, semua komputer dapat menjadi *client* ataupun *server* pada saat yang bersamaan. Pengguna komputer bertanggung jawab terhadap sumber daya yang dimilikinya, misalnya dengan mengatur hak akses apa saja yang dapat dilakukan orang lain.

Jaringan *Peer-to-Peer* ini tidak memiliki sistem kontrol terpusat pada sumber daya yang terbagi. Semua peralatan dapat dibagikan menjadi sumber daya yang dengan komputer-komputer pada jaringan yang sama. Pada umumnya jaringan *peer-to-peer* terdiri dari sekumpulan komputer *workstation* yang biasa disebut dengan *client*, tetapi dalam jaringan *peer-to-peer* komputer tersebut dapat juga berperan sebagai *server* pada saat yang bersamaan.



Gambar 2.5 *Peer to Peer*

Berdasarkan media transmisi data, jaringan terbagi dua, yaitu :

A. Jaringan Berkabel (*Wired Network*)

Pada jaringan ini, untuk menghubungkan satu komputer dengan komputer lain diperlukan penghubung berupa kabel jaringan. Kabel jaringan berfungsi dalam mengirim informasi dalam bentuk sinyal listrik antar komputer jaringan. Berdasarkan pada buku (Subramanian, 2010, p 89) Macam - macam tipe kabel untuk jaringan komputer terdiri dari 3 macam :

1. Twisted-pair Cable
 - UTP (*Unshielded Twisted Pair*)
 - STP (*Shielded Twisted Pair*)
2. Coaxial cable
3. Optical Fiber

B. Jaringan Nirkabel (*Wireless Network*)

Jaringan nirkabel atau lebih sering kita dengar sebagai Wi-Fi yang merupakan singkatan dari Wireless Fidelity, memiliki pengertian yaitu kumpulan standar yang biasa digunakan untuk Jaringan Lokal Nirkabel (*Wireless Local Area Networks - WLAN*) yang didasari pada spesifikasi IEEE 802.11. Standar terbaru dari spesifikasi 802.11a atau b, seperti 802.11 g, saat ini sedang dalam penyusunan. Spesifikasi terbaru tersebut menawarkan banyak peningkatan mulai dari luas cakupan yang lebih jauh hingga kecepatan transfernya.

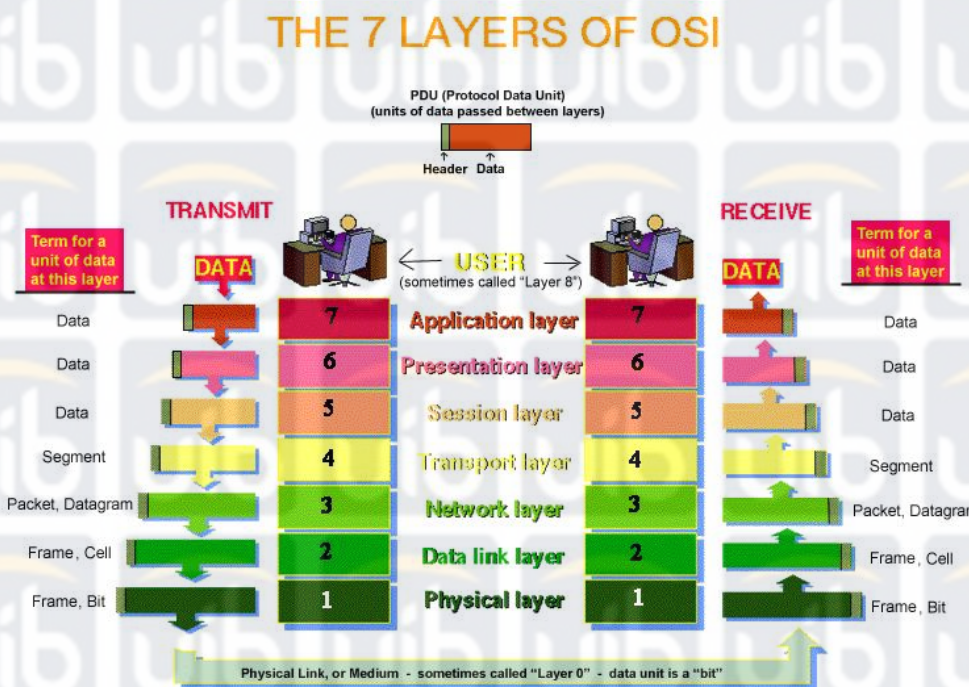
Awalnya Wi-Fi ditujukan untuk penggunaan perangkat nirkabel dan Jaringan Area Lokal (LAN), namun saat ini lebih banyak digunakan untuk mengakses internet. Hal ini memungkinkan seseorang dengan 16 komputer dengan kartu nirkabel (*wireless card*) atau Personal Digital Assistant (PDA) untuk terhubung dengan internet dengan menggunakan titik akses (atau dikenal dengan *hotspot*) terdekat.

2.2.3. Open System Interconnection (OSI) 7-Layer

Model OSI layer dikembangkan oleh International *Organization of Standardization* (ISO) pada tahun 1977 dengan tujuan untuk memfasilitasi *open interconnection* pada sistem komputer. ISO sebagai badan multinasional yang berfokus terhadap *worldwide agreements* atas standar-standar Internasional mengembangkan OSI karena *interconnection* dapat mendukung banyak vendor pada berbagai *environment*. Model OSI layer juga digunakan sebagai kerangka

kerja yang dipakai untuk memahami bagaimana informasi berjalan dalam sebuah jaringan.

Model OSI di bagi menjadi tujuh lapisan dan masing-masing lapisan mempunyai tugas sendiri-sendiri. Hal ini bukan berarti bahwa protokol yang dibuat berdasarkan model OSI mempunyai tujuh bagian yang terpisah atau hanya terdiri tujuh fungsi spesifik. Lapisan-lapisan ini lebih merepresentasikan tipe-tipe fungsi yang seharusnya didukung oleh protokol. Lapisan-lapisan ini disusun berdasarkan blok fungsi dalam model logik, dari atas ke bawah. Bagian atas merupakan fungsi paling dekat dengan pengguna (*user*) atau aplikasi, sedangkan bagian bawah mempunyai fungsi yang dekat dengan fisik jaringan atau antarmuka jaringan.



Gambar 2.6 OSI Layer

Lapisan Ke – 7 : *Application Layer*

Berfungsi sebagai antarmuka dengan aplikasi dengan fungsionalitas jaringan, mengatur bagaimana aplikasi dapat mengakses jaringan, dan kemudian membuat pesan-pesan kesalahan. Protokol yang berada dalam lapisan ini adalah HTTP, FTP, SMTP, dan NFS.

Lapisan Ke – 6 : *Presentation Layer*

Berfungsi untuk mentranslasikan data yang hendak ditransmisikan oleh aplikasi ke dalam format yang dapat ditransmisikan melalui jaringan. Protokol yang berada dalam *level* ini adalah perangkat lunak redirektor (redirector software), seperti layanan Workstation (dalam Windows NT) dan juga *Network shell* (semacam *Virtual Network Computing* (VNC) atau *Remote Desktop Protocol* (RDP)).

Lapisan Ke – 5 : *Session Layer*

Berfungsi untuk mendefinisikan bagaimana koneksi dapat dibuat, dipelihara, atau dihancurkan. Selain itu, di *level* ini juga dilakukan resolusi nama.

Lapisan Ke – 4 : *Transport Layer*

Berfungsi untuk memecah data ke dalam paket-paket data serta memberikan nomor urut ke paket-paket tersebut sehingga dapat disusun kembali pada sisi tujuan setelah diterima. Selain itu, pada *level* ini juga membuat sebuah tanda bahwa paket diterima dengan sukses (*acknowledgement*), dan mentransmisikan ulang terhadap paket-paket yang hilang di tengah jalan.

Lapisan Ke – 3 : *Network Layer*

Berfungsi untuk mendefinisikan alamat-alamat IP, membuat header untuk paket-paket, dan kemudian melakukan *routing* melalui *internetworking* dengan menggunakan router dan switch layer-3.

Lapisan Ke – 2 : *Data-Link Layer*

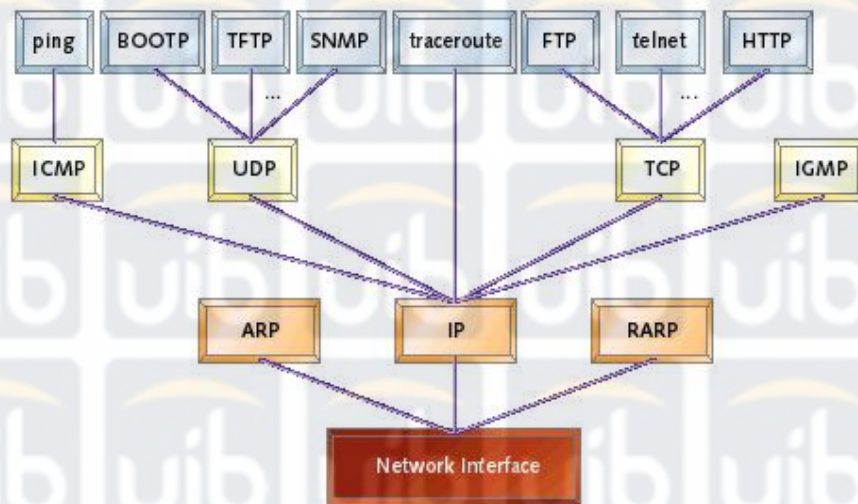
Befungsi untuk menentukan bagaimana bit-bit data dikelompokkan menjadi format yang disebut sebagai frame. Selain itu, pada level ini terjadi koreksi kesalahan, flow control, pengalamatan perangkat keras (seperti halnya *Media Access Control Address (MAC Address)*), dan menentukan bagaimana perangkat-perangkat jaringan seperti hub, bridge, repeater, dan switch layer 2 beroperasi. Spesifikasi IEEE 802, membagi *level* ini menjadi dua *level* anak, yaitu lapisan *Logical Link Control (LLC)* dan lapisan *Media Access Control (MAC)*.

Lapisan Ke – 1 : *Physical Layer*

Berfungsi untuk mendefinisikan media transmisi jaringan, metode pensinyalan, sinkronisasi bit, arsitektur jaringan (seperti halnya Ethernet atau Token Ring), topologi jaringan dan pengabelan. Selain itu, level ini juga mendefinisikan bagaimana *Network Interface Card (NIC)* dapat berinteraksi dengan media kabel atau radio.

2.2.4. TCP/IP

TCP/IP adalah sekumpulan protokol yang terdapat di dalam jaringan komputer (*network*) yang digunakan untuk berkomunikasi atau bertukar data antar komputer. TCP/IP merupakan standard protokol pada jaringan internet yang menghubungkan banyak komputer yang berbeda jenis mesin maupun sistem operasinya agar dapat berinteraksi satu sama lain.



Gambar 2.7 Beberapa protokol yang terdapat pada TCP/IP

TCP/IP protokol adalah jaringan dengan teknologi “packet *Switching*” yang berasal dari proyek DARPA (*development of Defense Advanced Research Project Agency*) ditahun 1970-an yang dikenal dengan nama ARPANET. TCP/IP sendiri sebenarnya merupakan suite dari gabungan beberapa protokol. Di dalamnya terdapat protokol TCP, IP, SMTP, POP, dan sebagainya TCP (*Transmission Control Protokol*) melakukan transmisi data per segmen, artinya paket data dipecah dalam jumlah yang sesuai dengan besaran paket, kemudian dikirim satu persatu hingga selesai. Agar pengiriman data sampai dengan baik,

maka pada setiap paket pengiriman, TCP akan menyertakan nomor seri (*sequence number*).

Komputer mitra yang menerima paket tersebut harus mengirim balik sebuah sinyal *ACK Knowledge* dalam satu periode yang ditentukan. Bila pada waktunya sang mitra belum juga memberikan ACK, maka terjadi “*time out*” yang menandakan pengiriman paket gagal dan harus diulang kemabali. Model protocol TCP disebut sebagai *connection oriented protocol* (Puspitasari, 2007).

2.2.5. IP Address

IP address merupakan alamat logika yang di berikan ke semua perangkat jaringan yang menggunakan protokol TCP/IP. IP address memungkinkan host pada jaringan yang berbeda maupun pada jaringan yang sama untuk bias saling berkomunikasi walaupun dlm platform yang berbeda.

IP address merupakan bilangan biner 32 bit yang terbagi menjadi empat kelompok, sehingga masing-masing kelompok terdiri dari bilangan biner 8 bit. Ini merupakan implementasi alamat IP yang disebut IPv4 (Wagito, 2005).

Internet Protocol (IP) merupakan sistem pengalamatan di *network* yang direpresentasikan dengan sederetan angka berupa kombinasi 4 deret bilangan antara 0 s/d 255 yang masing-masing dipisahkan oleh tanda titik (.), mulai dari 0.0.0.1 hingga 255.255.255.255. IP *address* memiliki 32 bit angka yang merupakan *logical address*. IP *address* bersifat *unique*, artinya tidak ada *device*, *station*, *host* atau *router* yang memiliki IP *address* yang sama dalam ruang lingkup satu *network*. Tapi setiap *host* komputer atau *router* dapat memiliki lebih dari satu IP *address*.

IP yang digunakan untuk kegiatan antar jaringan yaitu kelas A, B dan kelas C. Sedangkan IP kelas D digunakan untuk pengalamatan *multicast*, dan IP kelas E digunakan untuk percobaan. Berikut ilustrasi bagian-bagian alamat IP untuk kelas A, B, dan kelas C.

IP address kelas A terdiri atas 8 bit untuk network ID dan sisanya 24 bit digunakan untuk host ID, sehingga IP address kelas A digunakan untuk jaringan dengan jumlah host sangat besar. Pada bit pertama diberikan angka 0 sampai dengan 127.

IP address kelas B terdiri atas 16 bit untuk network ID dan sisanya 16 bit digunakan untuk host ID, sehingga IP address kelas B digunakan untuk jaringan dengan jumlah host tidak terlalu besar. Pada 2 bit pertama, diberikan angka 10.

IP address kelas C terdiri atas 24 bit untuk network ID dan sisanya 8 bit digunakan untuk host ID, sehingga IP address kelas C digunakan untuk jaringan berukuran kecil. Kelas C biasanya digunakan untuk jaringan Local Area Network atau LAN. Pada 3 bit pertama, diberikan angka 110.

Tabel 2.1

Jumlah Network ID & Host ID

Kelas	Antara	Jumlah Jaringan	Jumlah Host Jaringan
A	1 s.d. 126	126	16.777.214
B	128 s.d. 191	16.384	65.534
C	192 s.d. 223	2.097.152	254

2.2.6. *Bandwidth*

Bandwidth adalah kapasitas atau daya tampung kabel ethernet agar dapat dilewati trafik paket data dalam jumlah tertentu. *Bandwidth* juga bisa berarti jumlah konsumsi paket data per satuan waktu dinyatakan dengan satuan bit *per second* (bps) (Santosa, 2004). *Bandwidth* internet disediakan oleh *provider* internet dengan jumlah tertentu tergantung sewa pelanggan. *Bandwidth* adalah banyaknya ukuran suatu data atau informasi yang dapat mengalir dari suatu tempat ke tempat lain dalam sebuah *network* di waktu tertentu. *Bandwidth* dapat dipakai untuk mengukur baik aliran data analog maupun data digital. Sekarang sudah menjadi umum jika kata *bandwidth* lebih banyak dipakai untuk mengukur aliran data digital (Pandypata, dkk, 2009). *Bandwidth* dapat didefinisikan sebagai kapasitas atau daya tampung suatu *channel* komunikasi (medium komunikasi) untuk dapat dilewati sejumlah *traffic* informasi atau data dalam satuan waktu tertentu (mutiara, dkk, 2012).

2.2.7. *Quality of Service (QoS)*

Quality of Service (QoS) merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu usaha untuk mendefinisikan karakteristik dan sifat dari suatu servis (Ferguson & Huston, 1998). QoS digunakan untuk mengukur sekumpulan atribut kinerja yang telah dispesifikasikan dan diasosiasikan dengan suatu servis.

QoS didesain untuk membantu end user menjadi lebih produktif dengan memastikan bahwa user mendapatkan kinerja yang handal dari aplikasi-aplikasi

berbasis jaringan. Parameter-parameter yang digunakan untuk mengukur kinerja QoS adalah *packet loss*, *delay*, *throughput*.

2.2.7.1. Packet Loss

Jumlah paket yang hilang saat pengiriman paket data ke tujuan, kualitas terbaik pada saat LAN/WAN jika jumlah losses paling kecil (Santosa, 2004). Di dalam implementasi jaringan IP, nilai packet loss ini diharapkan minimum. Secara umum terdapat empat kategori penurunan performansi jaringan dengan versi TIPHON (*Telecommunications and internet protocol harmonization over networks*) (Joesman, 2008), yaitu sebagai berikut :

$$\text{Packet loss} = \frac{\text{Jumlah paket yang dikirim} - \text{jumlah paket yang diterima}}{\text{jumlah paket yang dikirim}} \times 100\%$$

Gambar 2.8 Rumus *Packet Loss*

Tabel 2.2

Performansi Jaringan IP Berdasarkan Packet Loss (Sumber TIPHON)

Kategori	Besar <i>Paket Loss</i>
Sangat Bagus	0 %
Bagus	3 %
Sedang	15 %
Jelek	25 %

2.2.7.2. Delay

Delay merupakan Waktu yang dibutuhkan untuk sebuah paket untuk mencapai tujuan, karena adanya antrian yang panjang, atau mengambil rute yang lain untuk menghindari kemacetan. Delay dapat dicari dengan membagi antara

panjang paket (L , packet length (bit)) dibagi dengan link *bandwidth* (R , link bandwidth (bit/s)).

$$Delay = \frac{\text{Panjang paket yang diterima (L)bit}}{\text{Link Bandwidth yang tersedia (R)bit/s}} \text{ detik}$$

Gambar 2.9 Rumus *Delay*

Tabel 2.3

Performasi Jaringan Berdasarkan Delay (Sumber TIPHON)

Kategori	Besar Delay
Sangat Bagus	<150 ms
Bagus	150 s/d 300 ms
Sedang	300 s/d 450 ms
Jelek	>450 ms

2.2.7.3. Throughput

Menurut Dewo (2003. 2), throughput adalah *bandwidth* aktual yang terukur pada suatu ukuran waktu tertentu dalam satu hari menggunakan rute internet yang spesifik ketika sedang mendownload suatu file.

$$Throughput = \frac{\text{Jumlah data yang dikirim (bit,Byte)}}{\text{Waktu yang dibutuhkan untuk mengirimkan data (s)}} \text{ bps atau Bps}$$

Gambar 2.10 Rumus *Throughput*

Tabel 2.4

Performasi Jaringan Berdasarkan Delay (Sumber TIPHON)

Kategori	Besar Throughput
Sangat Bagus	100
Bagus	75
Sedang	50
Jelek	< 25

2.2.8 *Queue Tree*

Queue Tree berfungsi untuk mengimplementasikan fungsi yang lebih kompleks dalam limit bandwidth pada mikrotik dimana penggunaan *mark packet* nya memiliki fungsi yang lebih baik. Digunakan untuk membatasi satu arah koneksi saja baik itu download maupun *upload*.

Digunakan untuk membatasi satu arah koneksi saja baik itu download maupun *upload*. Secara umum *Queue Tree* ini tidak terlihat berbeda dari *Simple Queue*. Perbedaan yang bisa kita lihat langsung yaitu hanya dari sisi cara pakai atau penggunaannya saja. Dimana *Queue Simple* secara khusus memang dirancang untuk kemudahan konfigurasi sementara *Queue Tree* dirancang untuk melaksanakan tugas antrian yang lebih kompleks dan butuh pemahaman yang baik tentang aliran trafik.

2.2.9. **HTB (Hierarchical Token Bucket)**

Hierarchical Token Bucket (HTB) merupakan teknik penjadwalan paket yang baru-baru ini diperkenalkan bagi router berbasis Linux, dikembangkan pertama kali oleh Martin Devera pada akhir 2001 untuk diproyeksikan sebagai pilihan (atau pengganti) mekanisme penjadwalan yang saat ini masih banyak dipakai yaitu CBQ.

Hierarchical Token Bucket (HTB) merupakan metode manajemen *bandwidth* yang digunakan untuk membatasi akses menuju alamat IP tertentu tanpa mengganggu trafik *bandwidth* pengguna lain.

Pada antrian HTB mempunyai parameter-parameter penyusun antrian adalah sebagai berikut:

1) *Rate*

Rate menentukan *bandwidth* maksimum yang dapat digunakan oleh setiap *class*, jika *bandwidth* melebihi nilai “*rate*” maka paket data akan dipotong.

2) *Ceil*

Ceil diatur untuk menentukan peminjaman *bandwidth* antar *class* (kelas), peminjaman *bandwidth* dilakukan kelas paling bawah ke kelas di atasnya.

Teknik ini disebut dengan *link sharing*.

Diambil dari algoritma *Token Bucket*. Analogi dari algoritma ini yaitu penetapan kapasitas keranjang (*Bucket*) pada pengambilan ke sekian kali (*Token*). Ketika pengecekan isi keranjang hampir penuh maka akan ditentukan kapan mengakhiri pengambilan. Itulah singkat gambaran dari algoritma *Token Bucket* yang selanjutnya digunakan pada *Hierarcial Token Bucket*. Pada konsep HTB disini yang merupakan bucket adalah paket data, selanjutnya akan dapat ditentukan berapa *token bucket* yang akan dijalankan. Dengan menjalankan HTB didapat sebuah sistem kontrol *bandwidth* yang akan digunakan oleh sebuah komposisi jaringan.

HTB diklaim menawarkan kemudahan pemakaian dengan teknik peminjaman dan implementasi pembagian trafik yang lebih akurat. Teknik antrian HTB memberikan fasilitas pembatasan traffic pada setiap *level* maupun klasifikasi, *bandwidth* yang tidak terpakai bisa digunakan oleh klasifikasi yang lebih rendah.

Penerapan *Queue* didasarkan pada *Hierarchical Token Bucket* (HTB). HTB (*Hirarchical Token Bukcet*) adalah metode pengelompokan *queue* (antrian) yang berguna untuk menangani berbagai jenis traffik. HTB memungkinkan kita

untuk membuat struktur antrian berjenjang serta kaitannya antar antrian seperti *parent-child* atau *child-child*.

Setidaknya ada 3 langkah dasar yang harus kita ikuti sebelum membuat

HTB:

- Klasifikasi dan menandai trafik - yaitu mengklasifikasikan trafik untuk digunakan lebih lanjut.
- Membuat aturan untuk menandai trafik - yaitu menempatkan trafik ke dalam kelompok tertentu, antrian tertentu, kemudian menentukan tindakan yang akan diambil untuk masing-masing kelompok tadi.
- Menerapkan aturan untuk antarmuka tertentu - aturan yang bisa ditambahkan (*global-in*, *global-out*, *global-total*) dapat diterapkan untuk seluruh antarmuka, atau antarmuka tertentu, atau jenis *parent* antrian tertentu.

Di Mikrotik RouterOS, struktur hirarki/berjenjang dapat dipasang pada 4 tempat yang berbeda yang biasanya digunakan sebagai pilihan *parent* antrian:

- *global-in*: mewakili antarmuka input secara umum. Maksudnya, antrian yang masuk ke *global-in* berlaku untuk trafik sebelum difilter seperti trafik *upload*.
- *global-out*: mewakili semua antarmuka output secara umum. Maksudnya, antrian yang keluar dari *global-out* berlaku untuk trafik setelah difilter seperti trafik *download*.
- *global-total*: mewakili semua antarmuka *input* dan *output* bersama-sama (dengan kata lain itu adalah penyatuan *global-in* dan *global-out*).

Digunakan dalam kasus ketika pelanggan memiliki batas untuk *upload* dan *download*.

- *<interface name>*: merupakan salah satu *outgoing interface* tertentu.

Hanya trafik yang keluar dari antarmuka ini yang bisa diqueue/masuk ke dalam daftar antrian.

Namun untuk versi 6 keatas *parent global-in*, *global-out*, dan *global-total* untuk *Queue Tree* diganti dengan *global* yang setara dengan *global-total* untuk versi sebelumnya.

Antrian pada HTB dibatasi oleh 2 rate:

- CIR (*Comitted Information Rate*) - batas bawah atau minimal trafik (*limit-at*) yang dapat diperoleh antrian. *Limit-at* membatasi minimal trafik suatu antrian, tidak peduli walau dalam kondisi apapun antrian tidak akan mendapat trafik dibawah batas ini.
- MIR (*Maximal Information rate*) - batas atas atau maksimal trafik (*max-limit*) yang bisa diperoleh antrian. *Max-limit* membatasi maksimal trafik suatu antrian, dan setiap antrian akan mencapai batas ini jika *parent* masih memiliki cadangan *bandwidth*.

Sederhananya, pertama-tama trafik yang lewat akan mendapat batas minimal trafik (*limit-at*) kemudian setiap antrian mencoba mendekati batas maksimal (*max-limit*) dengan meminjam *bandwidth* yang tersisa dari *parent*.

Oleh karenanya untuk mendapatkan hasil terbaik penggunaan fitur limitasi diatas sangat dianjurkan. Jika kita menggunakan *parent* antrian, maka total jumlah *limit-at* untuk *child* antrian pada semua tingkatan harus kurang dari atau sama dengan jumlah trafik yang tersedia pada *parent*.

$$CIR (parent) \geq CIR (child1) + CIR (child2) + CIR (child3) + \dots$$

Dan *max-limit* masing-masing *child* antrian pada semua tingkatan harus kurang dari atau sama dengan trafik maksimal *parent*.

$$MIR (parent) \geq MIR (child1) \ \& \ MIR (parent) \geq MIR (child2) \ \& \ MIR (parent) \geq \dots$$

Sudah kita ketahui bahwa semua antrian tidak akan mendapatkan trafik dibawah nilai *limit-at* pada kondisi apapun. Maka nilai prioritas bertanggungjawab untuk mendistribusikan *bandwidth* yang tersisa dari *parent* antrian kepada *child* antrian sehingga mereka mampu mencapai nilai *max-limit*.

Antrian dengan prioritas lebih tinggi akan mencapai nilai *max-limit* daripada antrian dengan prioritas yang lebih rendah. 8 adalah prioritas terendah dan 1 adalah prioritas tertinggi. Perlu dicatat bahwa prioritas akan bekerja jika nilai *max-limit* bukan nol (0). Untuk antrian yang melewati nilai *limit-at* dan *max-limit* *parent* antrian (*leaf queue*), prioritas di dalam (*inner*) antrian tidak akan memiliki arti.

Name	Parent	Packet Marks	Limit At (b...	Max Limit ...
Parent	global		30k	40k
Child	Parent	Download	10k	20k
Child2	Parent	Browsing	20k	30k
Child3	Parent	Streaming	15k	25k

Gambar 2.11 Contoh *Queue Tree* HTB

2.2.10. PCQ (*Per Connection Queue*)

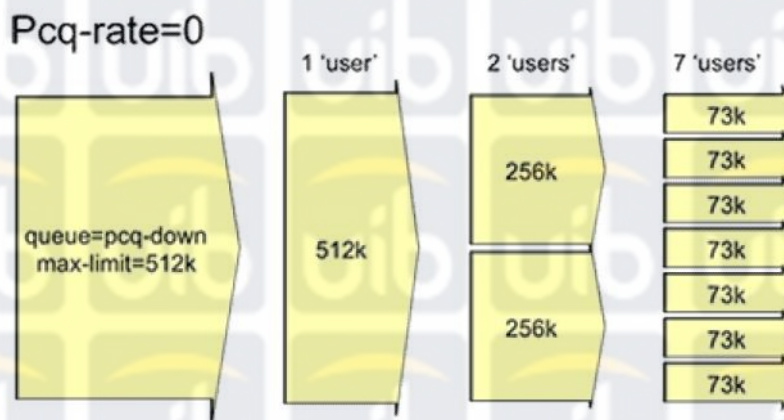
PCQ merupakan salah satu cara melakukan manajemen *bandwidth* yang cukup mudah dimana PCQ bekerja dengan sebuah algoritma yang akan membagi *bandwidth* secara merata ke sejumlah client yang aktif. PCQ ideal diterapkan

apabila dalam pengaturan *bandwidth* kita kesulitan dalam penentuan *bandwidth* per client.

Misalnya, sebelumnya kita bisa melakukan *bandwidth management* dengan system HTB dimana jumlah client sedikit, maka masih mudah bagi admin jaringan dalam menentukan parameter limit-at. Tetapi bagaimana jika *bandwidth* 1 Mbps namun ingin dibagi rata ke 200-an client. Jika menggunakan model HTB, akan sulit untuk menentukan limit-at. Dengan kondisi seperti ini, akan lebih mudah jika kita serahkan perhitungan *management bandwidth* ke router, agar Router yang akan membagi *bandwidth* secara otomatis ke *client*.

Cara kerja PCQ adalah dengan menambahkan *sub-queue*, berdasar classifier tertentu. Berikut gambaran cara kerja PCQ dengan parameter PCQ-Rate

= 0.



Gambar 2.12 PCQ-rate=0

PCQ *rate* adalah dasar perhitungan router. Seberapa besar *rate-limit* yg akan diberikan ke user yg aktif. Cara setting PCQ sebenarnya cukup mudah. Kita hanya perlu menambahkan *Queue Type* PCQ, kemudian tentukan nilai *classifier* dan nilai *rate*. Untuk management *traffic download*, centang opsi *classifier dst.address*. Dan untuk management *traffic upload*, centang opsi *src.address*.