

BAB II

TINJAUAN PUSTAKA

Manajemen *bandwidth* merupakan proses pengaturan *bandwidth* yang tepat untuk masing-masing *client* pada sistem jaringan internet yang mendukung kebutuhan aplikasi layanan internet. Manajemen *bandwidth* dengan prioritas alamat IP *client* dapat dilakukan secara *sentralisasi* menggunakan *mikrotik routerboard RB750* dengan administrasi pengaturan *mikrotik router* melalui *Windows Application* yang sering disebut dengan *WinBox*. Implementasi manajemen *bandwidth* pada *mikrotik routerboard RB750* menggunakan metode HTB (*Hierarchical Token Bucket*) dengan teknik antrian *simple queue* dan *queue tree*. Pengaturan sistem manajemen *bandwidth* terdapat beberapa hal yang diatur diantaranya adalah kecepatan *upload*, *download* pada masing-masing alamat IP *client*. Untuk mengamati performansi jaminan kualitas jaringan QoS (*Quality of Service*) digunakan aplikasi *Network Analyzer Wireshark*. Dengan menggunakan aplikasi *Network Analyzer Wireshark* dapat menggambarkan suatu tingkatan pencapaian di dalam sistem komunikasi data.

Pada Bab II ini akan dibahas tentang tinjauan pustaka yang dapat mendukung teori, rumus, dan standarisasi yang diperlukan dalam pembahasan sistem manajemen *bandwidth* dengan prioritas alamat IP *Client*. Secara khusus tinjauan pustaka untuk sistem manajemen *bandwidth* akan diuraikan dalam sub bab di bawah ini.

2.1 Konsep Dasar Jaringan Internet

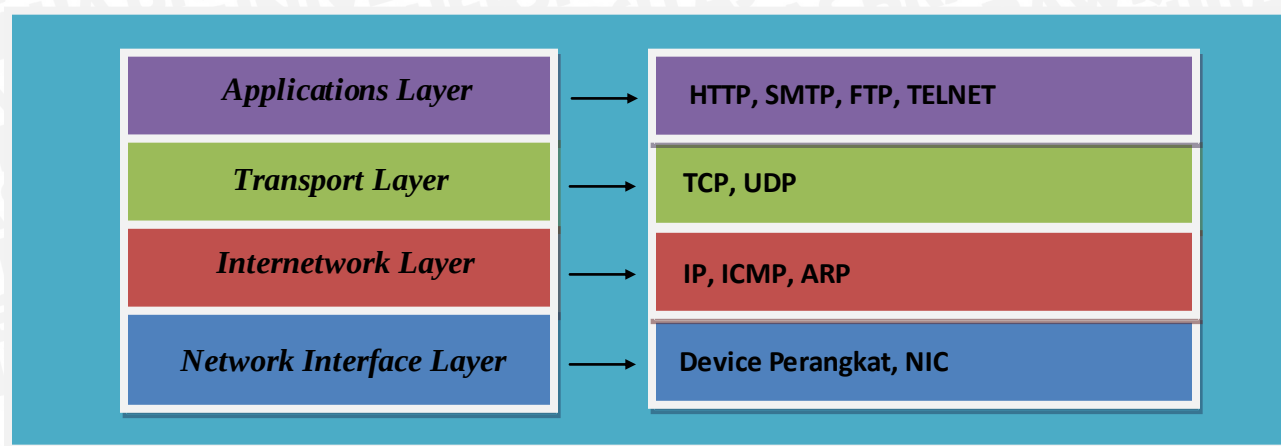
Jaringan internet adalah fasilitas transfer data melalui perangkat-perangkat yang terhubung ke jaringan internet. Jaringan internet dirancang dari perangkat keras dan perangkat lunak. Manfaat jaringan internet adalah sebagai berikut:

1. *Resource sharing* adalah berbagi program, data, dan peralatan pada satu jaringan.
2. *Realibilitas* adalah proses *back-up* data.
3. *Skalabilitas* adalah untuk meningkatkan kinerja sistem.

2.2 Konsep Dasar Protokol TCP/IP

Protokol adalah sejumlah aturan yang mengatur format *frame*, paket atau pesan dalam komunikasi data pada layer. Protokol TCP/IP didesain untuk melakukan fungsi-fungsi komunikasi data pada internet atau intranet (Onno W. Purbo, et al., 2001:22). Perangkat lunak jaringan protokol TCP/IP dimodelkan dalam bentuk layer. Protokol TCP/IP terdiri dari

4 susun *layer* yaitu *Application layer*, *Transport layer*, *Internetwork layer* dan *Network interface layer*.



Gambar 2.1 Model Arsitektur Protokol TCP/IP

Sumber: William Stalling, 2001, Komunikasi Data dan Jaringan Komputer

Keterangan:

1. *Network Interface Layer*

Network Interface Layer merupakan gabungan dari *physical* dan *data link layer* model OSI (*Open System Interconnection*). *Network Interface Layer* pada model TCP/IP berfungsi untuk mengirim dan menerima data ke dan dari media fisik. Media fisiknya dapat berupa kabel tembaga, koaksial, UTP, serat optik, atau gelombang radio. Protokol pada *layer* ini harus mampu menerjemahkan sinyal listrik menjadi data digital yang dapat dimengerti oleh komputer. Contoh: Ethernet, *wireless*, NIC.

2. *Internetwork Layer*

Internetwork Layer bertugas menentukan format dan protokol paket yang disebut *Internet Protocol* (IP) dan mengirimkan paket-paket IP yang berisi informasi ke alamat yang dituju. *Routing* paket dari sumber ke tujuan diatur di *internetwork layer*. Desain *routing* pengiriman paket dari sumber ke tujuan merupakan hal yang sangat penting. Terdapat 2 macam *routing*, diantaranya adalah sebagai berikut:

a. *Static Routing*

Static Routing merupakan metode *routing* yang tabel jaringannya dibuat secara manual oleh *administrator* jaringannya.

b. Dynamic Routing

Dynamic routing merupakan teknik *routing* dengan menggunakan beberapa aplikasi *networking* yang bertujuan menentukan *routing* secara otomatis. Tabel *routing* (ARP table) akan diatur oleh sebuah protokol *routing*.

Route pengiriman paket tergantung beban jaringan pada saat itu. Pengendalian antrian juga merupakan tugas *internetwork layer*. Protokol pada *internetwork layer* terdiri dari IP, ICMP, dan ARP. IP (*Internet Protocol*) berfungsi untuk menyampaikan paket data ke alamat yang tepat. ICMP (*Internet Control Message Protocol*) adalah protokol yang digunakan untuk mengirimkan pesan dan melaporkan kegagalan pengiriman data. ARP (*Address Resolution Protocol*) adalah protokol yang digunakan untuk menemukan alamat tujuan data akan dikirimkan dengan meresolusi alamat IP ke MAC address dari *hardware* tujuan data tersebut.

3. Transport Layer

Transport layer memberikan fungsi pengiriman data dan meningkatkan QoS atau kualitas jaringan dari *host* sumber ke *host* tujuan. Protokol pada *layer transport* adalah *Transmission Control Protocol* (TCP) dan *User Datagram Protocol* (UDP). Protokol TCP merupakan protokol yang memberikan fungsi pengiriman data secara *connectionoriented*, pencegahan duplikasi data, *congestion control* dan *flow control*. Contoh penggunaan protokol TCP adalah E-Mail. Protokol UDP memberikan fungsi pengiriman data secara *connectionless*, jalur yang tidak *reliable*. Contoh penggunaan protokol UDP adalah pada *video streaming* dan layanan *real-time*.

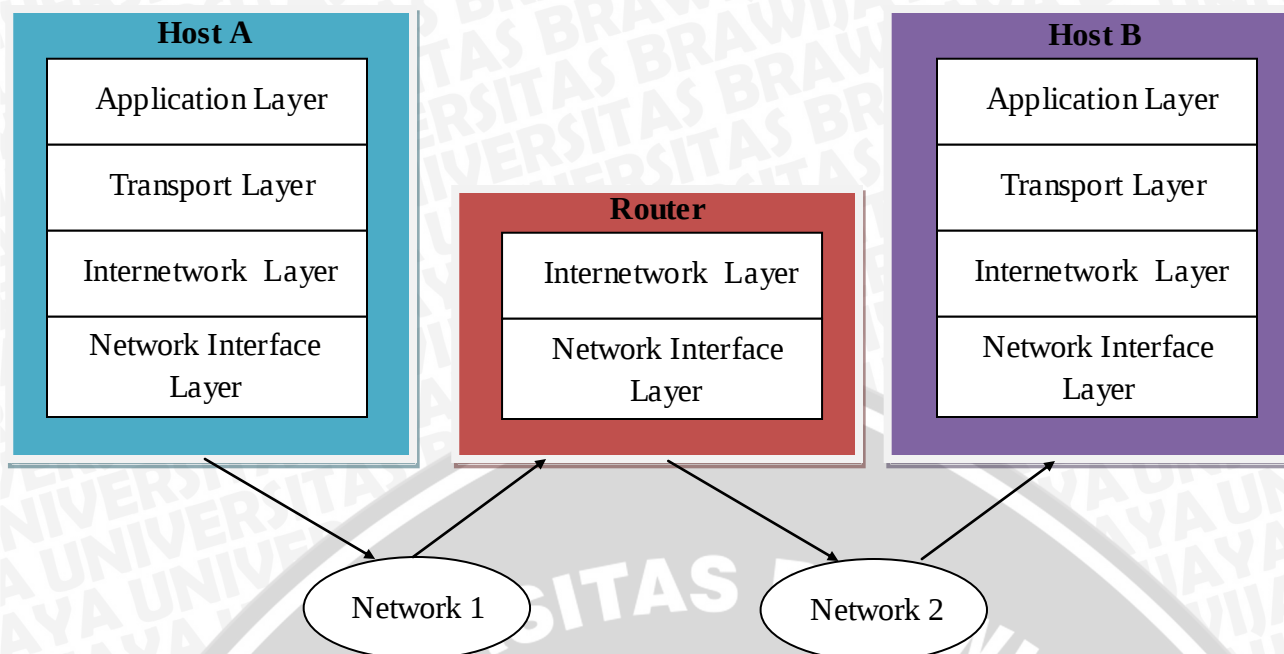
4. Application Layer

Application layer merupakan penyedia program atau aplikasi yang bekerja nyata bagi pengguna untuk berkomunikasi menggunakan TCP/IP. Terdapat tiga protokol pada *application layer* yang sangat berfungsi mendukung agar aplikasi-aplikasi dapat berfungsi diantaranya adalah sebagai berikut:

1. Keamanan untuk menjamin kerahasiaan, keaslian, pengakuan, dan kontrol bila diperlukan.
2. DNS untuk penamaan di internet.
3. Manajemen jaringan.

Contoh aplikasi pada *application layer* antara lain E-Mail, World Wide Web, dan multimedia.

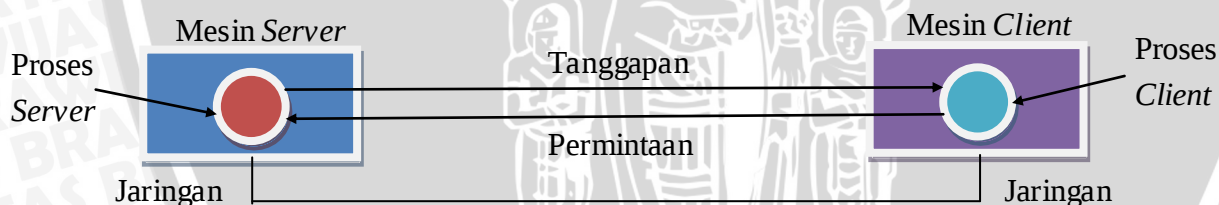
Interface yang digunakan untuk saling berkomunikasi adalah nomor *port* dan *socket*.



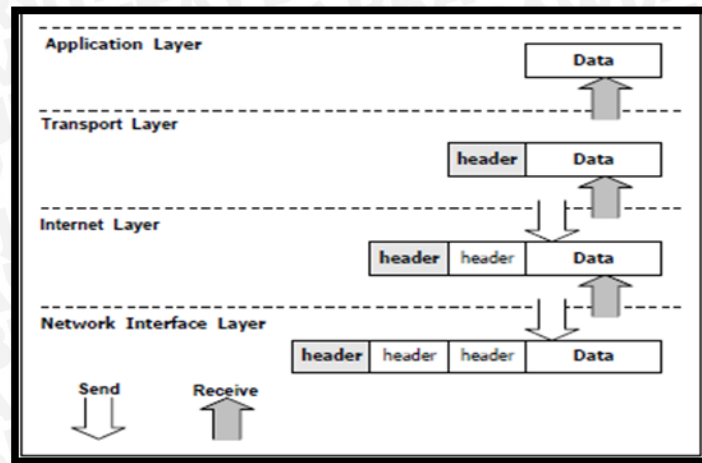
Gambar 2.2 Interkoneksi Layer TCP/IP Pada Jaringan
Sumber: Leon-Garcia, Alberto, 2001:60

2.3 Proses Komunikasi Data Pada TCP/IP

Komunikasi data TCP/IP bersifat *client-server*. *Server* adalah aplikasi yang memberikan pelayanan kepada *client* yang dapat menerima permintaan jaringan internet. *Client* adalah yang meminta pelayanan jaringan internet. Gambar 2.3 merupakan komunikasi data TCP/IP model *client-server*.



Gambar 2.3 Gambar Komunikasi Data Model Client-Server
Sumber: Buku Jaringan Komputer Edisi III, Andrew S. Tanenbaum



Gambar 2.4 Proses Aliran Komunikasi Data Pada Layer TCP/IP
Sumber: Craig Hunt, 1992:10

2.3.1 Proses Aliran Komunikasi Data Layer TCP/IP Pengiriman dari *Application Layer* ke *Network Interface Layer*

Proses aliran pengiriman komunikasi data pada layer TCP/IP dari *application layer* ke *network interface layer* melalui layer atas ke layer bawah. Berikut adalah proses aliran komunikasi datanya:

1. Data pada layer aplikasi dikirimkan ke lapisan transport dalam bentuk paket-paket dengan panjang tertentu.
2. Protokol menambahkan sejumlah bit pada setiap paket sebagai *header* yang berisi informasi mengenai urutan segmentasi untuk menjaga integritas data dan deteksi dan koreksi kesalahan.
3. Data dari lapisan transport, data yang telah diberi *header* tersebut diteruskan ke lapisan Internet. Pada lapisan internet terjadi penambahan *header* oleh protokol yang berisi informasi alamat tujuan, alamat pengirim dan informasi lain yang dibutuhkan untuk melakukan *header*. Kemudian terjadi pengarahannya ke *network* dan *interface* tujuan. Pada lapisan internet terjadi proses segmentasi data, karena panjang paket yang akan dikirimkan harus disesuaikan dengan kondisi media komunikasi pada *network* yang akan dilalui. Selanjutnya data menuju *Network Access Layer* dimana data akan diolah menjadi frame-frame, menambahkan informasi keandalan dan *address* pada *level link*.
4. Protokol pada lapisan *Network Access Layer* menyiapkan data dalam bentuk yang paling sesuai untuk dikirimkan melalui media komunikasi tertentu. Terakhir data akan sampai pada *Physical Layer* yang akan mengirimkan data dalam bentuk besaran-besaran listrik

atau fisik seperti tegangan, arus, gelombang radio maupun cahaya, sesuai media yang digunakan.

2.3.2 Proses Aliran Komunikasi Data Layer TCP/IP Penerimaan Dari *Network Interface Layer* Ke *Application Layer*

Proses aliran penerimaan komunikasi data layer TCP/IP dari *network interface layer* ke *application layer* melalui layer bawah ke layer atas. Berikut adalah proses aliran komunikasi datanya:

1. Sinyal yang diterima pada *Physical layer* akan diubah ke dalam data.
2. Protokol akan memeriksa integritas data dan jika tidak ditemukan *error*, *header* yang ditambahkan akan dilepas. Selanjutnya data diteruskan ke lapisan *network*.
3. Pada lapisan *network*, alamat tujuan dari paket data yang diterima akan diperiksa. Jika alamat tujuan merupakan *address host* yang bersangkutan, maka *header* lapisan *network* akan dicopot dan data akan diteruskan ke lapisan yang di atasnya. Namun jika tidak, data akan di-*forward* ke *network* tujuannya, sesuai dengan informasi *header* yang dimiliki.
4. Pada lapisan transport, kebenaran data akan diperiksa kembali, menggunakan informasi *header* yang dikirimkan oleh pengirim. Jika tidak ada kesalahan, paket-paket data yang diterima akan disusun kembali sesuai urutannya pada saat akan dikirim dan diteruskan ke lapisan aplikasi pada penerima.

2.4 Konsep Dasar *Internet Protocol Versi 4 (IPv4)*

Dalam arsitektur paket protokol TCP/IP, protokol yang digunakan adalah *Internet Protocol (IP)*. Alamat IPv4 merupakan representasi dari 32 bit yang dipisahkan oleh tanda titik setiap 8 bit. Bit ini terbagi menjadi 4 bagian, yang dikenal sebagai *octet* atau *byte* (8 bit). Penggunaan alamat IP adalah unik, artinya tidak diperbolehkan menggunakan alamat IP yang sama dalam satu jaringan.

2.5 Pembagian Kelas-Kelas Alamat IP

Pembagian kelas-kelas alamat IP didasarkan pada dua hal yaitu *network ID* dan *Host ID*. *Network ID* merupakan bagian alamat IP yang menunjukkan jaringan tempat komputer berada. *Host ID* merupakan bagian alamat IP yang digunakan untuk menunjukkan *server*, *router* dan semua *host* TCP/IP dalam jaringan tersebut. Dalam alamat IP terdapat alamat *network* dan alamat *broadcast*. Alamat *network* digunakan dalam *routing* untuk menunjukkan

pengiriman paket ke *remote network*. Alamat *broadcast* merupakan alamat yang digunakan oleh aplikasi dan *host* untuk mengirim informasi ke semua titik di dalam jaringan.

Pengalamatan dalam IPv4 memiliki 5 jenis kelas, A,B,C,D dan E. Tetapi hanya kelas A, B, dan C yang digunakan secara umum. Tabel 2.1 menggambarkan informasi tentang skema kelas-kelas yang ada pada IPv4.

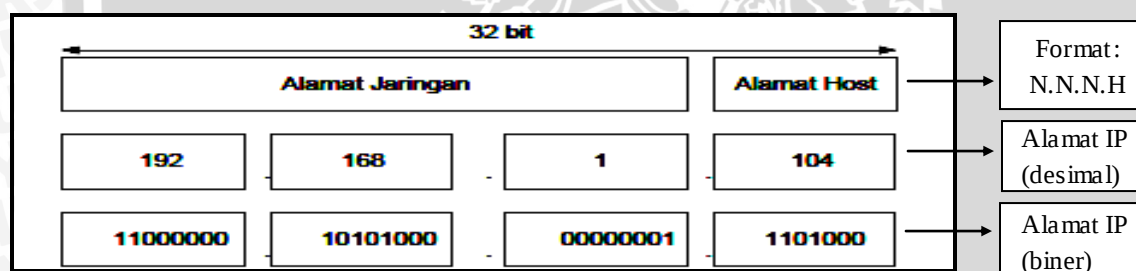
Tabel 2.1 skema kelas-kelas yang ada pada IPv4

Kelas	Format	Range Alamat	Jumlah Host Maksimal
A	N.H.H.H	0.0.0.0 – 127.255.255.255	$2^{24} - 2$
B	N.N.H.H	128.0.0.0 – 191.255.255.255	$2^{16} - 2$
C	N.N.N.H	192.0.0.0 – 223.255.255.255	$2^8 - 2$
D	-	224.0.0.0 – 239.255.255.255	-
E	-	240.0.0.0 – 254.255.255.255	-

Sumber: Onno W. Purbo, et al., 2001 : 80

Keterangan: N = Alamat Jaringan, H = Alamat Host

Gambar 2.5 merupakan contoh alamat IPv4 dengan 32 bit kelas C:



Gambar 2.5 Contoh alamat IPv4 dengan 32 bit

Sumber: Onno W. Purbo, et al., 2001 : 80

2.6 Subnetting

Alamat IP terdiri dari 2 bagian, yaitu alamat jaringan dan alamat *host*. *Subnet mask* untuk menunjukkan letak suatu *host*, apakah berada dalam satu jaringan atau tidak. *Subnet mask* direpresentasikan dengan nilai 1 dan 0 dimana bagian dengan nilai 1 merepresentasikan alamat jaringan sedangkan yang memiliki nilai 0 merupakan alamat *host*-nya Tabel 2.2 merupakan tabel *subnet mask* untuk tiap-tiap kelas alamat IP.

Tabel 2.2 Subnet Mask Untuk Tiap-Tiap Kelas Alamat IP

Kelas IP	Format	Bit Subnet Mask	Subnet Mask Dalam Desimal
A	N.H.H.H	11111111.00000000.00000000.00000000	255.0.0.0
B	N.N.H.H	11111111. 11111111. 00000000. 00000000	255. 255.0.0
C	N.N.N.H	11111111. 11111111. 11111111. 00000000	255. 255. 255.0

Sumber: Onno W. Purbo, et al., 2001 : 81

Keterangan: N = Alamat Jaringan, H = Alamat Host

Contoh Pembuktian Dua Alamat IP Berada Dalam Satu Jaringan Pada Kelas C• **PC 1**

Alamat IP : 192.168.0.2/24

Subnet Mask : 255.255.255.0

Network ID : 192.168.0.0

Broadcast ID : 192.168.0.255

• **PC 2**

Alamat IP : 192.168.0.130/24

Subnet Mask : 255.255.255.0

Network ID : 192.168.0.0

Broadcast ID : 192.168.0.255

Pembuktian:• **PC 1**

Alamat IP (biner) : 11000000.10101000.00000000.00000000

Subnet Mask (biner) : 11111111.11111111.11111111.11111111

Hasil (biner) : 11000000.10101000.00000000.00000000

Hasil (desimal) : 192 . 168 . 0 . 0

AND

→ Alamat jaringan

• **PC 2**

• Alamat IP (biner) : 11000000.10101000.00000000.10000010

• Subnet Mask (biner) : 11111111.11111111.11111111.11111111

• Hasil (biner) : 11000000.10101000.00000000.00000000

• Hasil (desimal) : 192 . 168 . 0 . 0

AND

→ Alamat jaringan

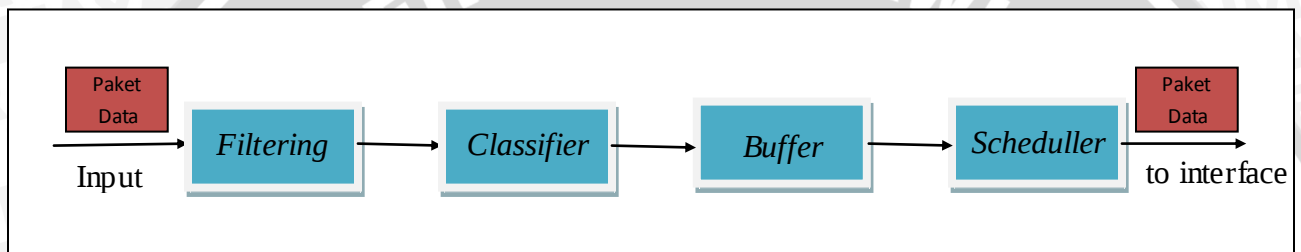
Maka dapat disimpulkan bahwa PC 1 dan PC 2 terletak pada jaringan yang sama karena memiliki alamat jaringan yang sama.

2.7 Konsep Dasar Manajemen *bandwidth*

Manajemen *bandwidth* merupakan pengalokasian *bandwidth* untuk mendukung kebutuhan aplikasi suatu layanan jaringan. Manajemen *bandwidth* diperlukan bagi jaringan multi layanan dengan menerapkan layanan *Quality of Service* (QoS) yang menggambarkan tingkat pencapaian pada suatu sistem komunikasi data.

2.7.1 Blok Diagram Penyusun Manajemen *Bandwidth* Mikrotik RouterOS Prioritas Alamat IP Client

Penyusun manajemen *bandwidth* pada mikrotik routerOS prioritas alamat IP *client* terdiri atas: *filtering*, *classifier*, *buffer*, *scheduler*. Gambar 2.6 merupakan blok diagram penyusun manajemen *bandwidth* Mikrotik RouterOS Prioritas alamat IP *Client*.

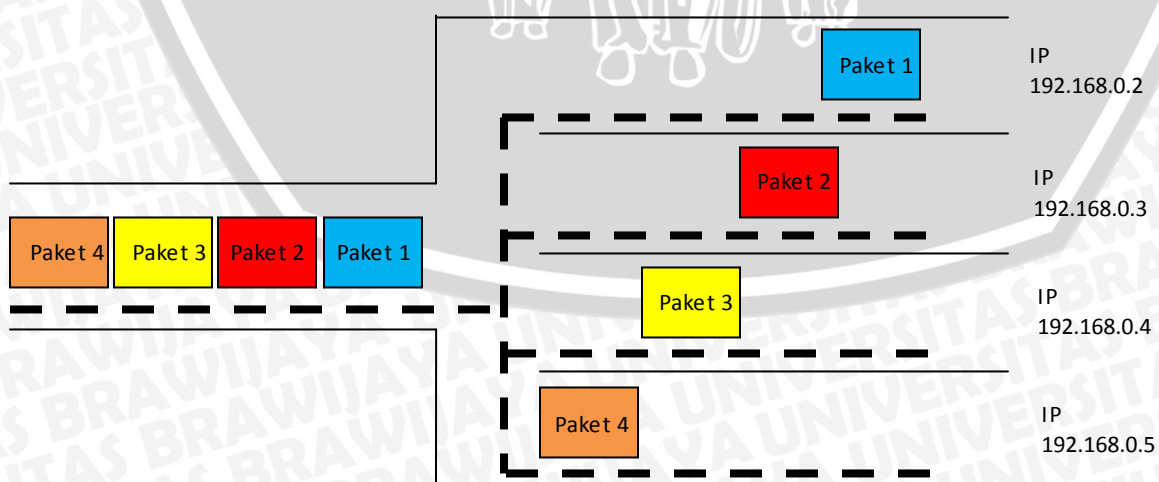


Gambar 2.6 Blok Diagram Penyusun Manajemen *Bandwidth* Mikrotik RouterOS Prioritas alamat IP *Client*
Sumber: Manajemen *Bandwidth* Internet dan Intranet, Budi Santosa, ST

Proses aliran manajemen *bandwidth* mikrotik routerOS sesuai dengan blok diagram adalah sebagai berikut:

1. *Filtering*

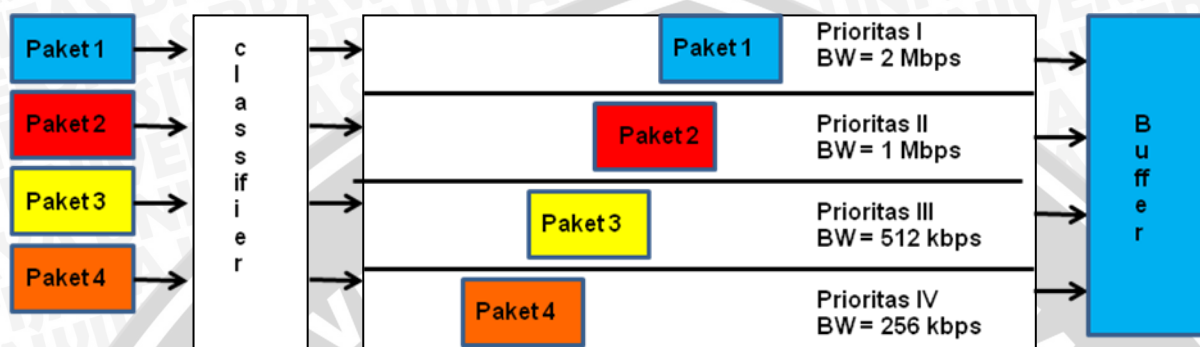
Filtering berfungsi untuk memfilter paket data berdasarkan alamat IP atau alamat *port* dan mengarahkan paket data ke tujuan yang benar. Gambar 2.7 merupakan *filtering* paket data.



Gambar 2.7 *Filtering* Paket Data Berdasarkan Alamat IP *Client*
Sumber: Manajemen *Bandwidth* Internet dan Intranet, Budi Santosa, ST

2. Classifier

Classifier bertugas untuk mengarahkan paket-paket yang datang ke kelas-kelas yang bersesuaian untuk mempermudah penanganan paket data menuju antrian atau *buffer*. Pada *classifier* terdapat *estimator* yang bertugas mengestimasi *bandwidth* yang digunakan oleh klasifikasi kelas. Jika sebuah kelas telah melampaui nilai *set* yang telah ditentukan, maka *estimator* akan menandai kelas tersebut sebagai kelas yang *overlimit*.

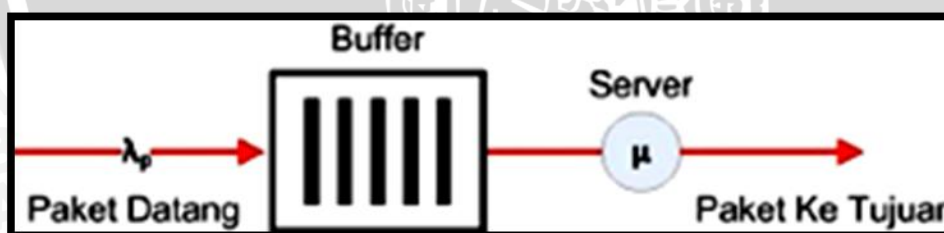


Gambar 2.8 Classifier Paket Data

Sumber: Manajemen Bandwidth Internet dan Intranet, Budi Santosa, ST

3. Buffer

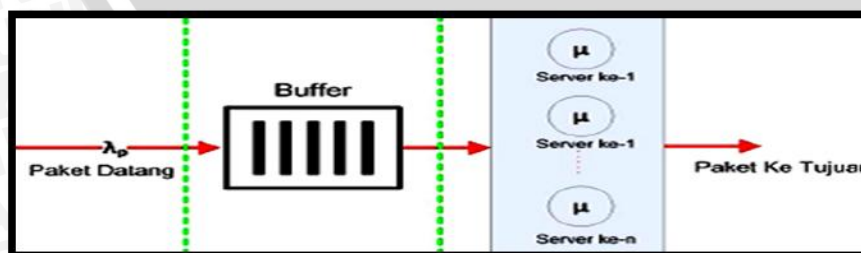
Buffer merupakan tempat penyimpanan paket data sementara. *Buffer* menyesuaikan waktu dengan menerapkan teknik antrian. Jika trafik pada *mikrotik* sedang tinggi maka prioritas tertinggi akan di proses terlebih dahulu, sedangkan yang lainnya akan di berikan ke antrian.



Gambar 2.9 Model Antrian M/M/1

Sumber: Mischa Schwartz, 1987:22

Gambar 2.10 menunjukkan model antrian M/M/S

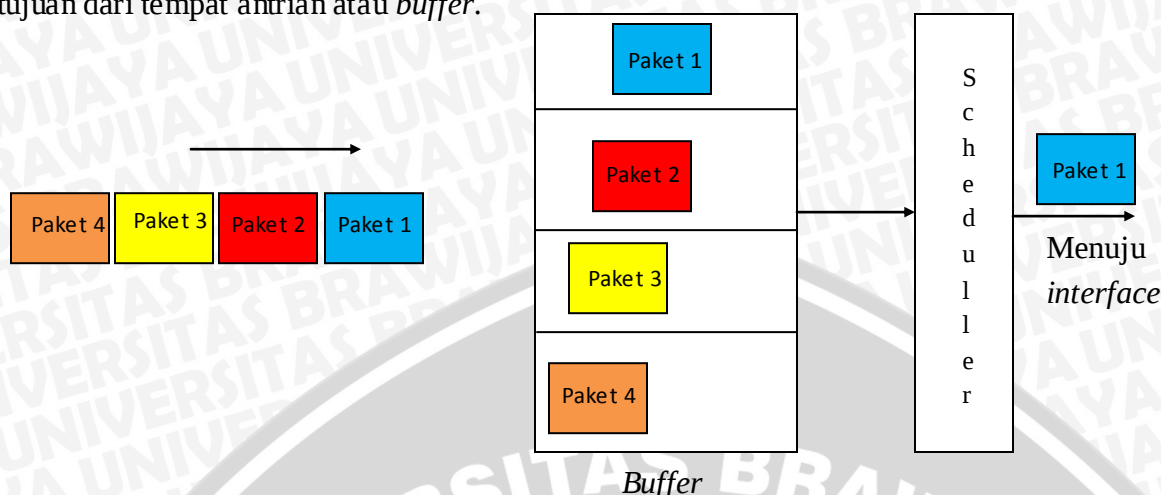


Gambar 2.10 Model Antrian M/M/S

Sumber: Misha Schwartz, 1987:22

4. Scheduler

Scheduler bertugas untuk menentukan penjadwalan paket data yang akan dikirim ke tujuan dari tempat antrian atau *buffer*.



Gambar 2.11 *Schedulling Paket Data Berdasarkan Alamat IP Client*
Sumber: Manajemen Bandwidth Internet dan Intranet, Budi Santosa, ST

2.8 Konsep Dasar HTB (*Hierarchical Token Bucket*)

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:

a. *Rate*

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

b. *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*.

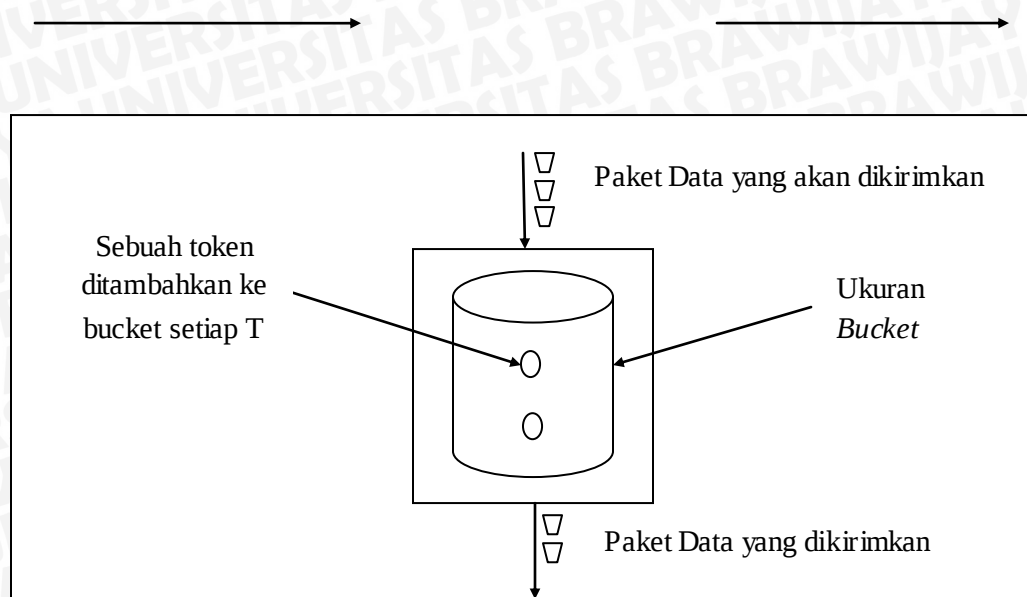
2.8.1 Algoritma HTB

Algoritma *Hierarchical Token Bucket* (HTB) dapat dianalogikan sebagai *token* dan *bucket*. *Token* merupakan *rate* data yang bertahan secara terus-menerus pada *bucket*. *Bucket* merupakan ukuran untuk menampung *token* hingga ukuran maksimum ember. Sebuah paket hanya dapat ditransmisikan apabila terdapat *token* yang cukup untuk menampung panjang paket tersebut. Jika *token* tidak tersedia maka paket yang dikirimkan harus menunggu sampai tersedia *token* yang cukup pada *bucket* untuk mengirimkan paket data.

Paket data yang akan dikirim

Metode HTB

Paket data yang terkirim



Gambar 2.12 ilustrasi sistem metode HTB (Hierarchical Token Bucket)
Sumber: William Stalling, Jaringan Komputer

Terdapat 2 kondisi hubungan antara paket yang datang dengan keadaan *token* pada *bucket*:

- **Link data yang datang sama dengan link masuknya *token* pada *bucket***

Jika *link* data yang datang sama dengan *link* masuknya *token* pada *bucket*, maka setiap paket yang masuk memiliki *token* masing-masing. Ini yang dimaksud dengan tanpa menunggu antrian.

- **Link data yang datang lebih besar dari link masuknya *token* pada *bucket***

Apabila *link* data yang datang lebih besar dari *link* masuknya *token* pada *bucket*, maka *bucket* akan segera kosong dari *token* dan *token bucket filter* akan menutup aliran untuk sementara, ini yang dimaksud dengan *overlimit*. Jika paket-paket data tetap datang, maka paket-paket akan dibuang.

1. Konsep Dasar Teknik Antrian Simple Queue

Simple Queue merupakan teknik antrian manajemen *bandwidth* pada mikrotik routerboard RB750. *Simple Queue* merupakan teknik antrian yang sederhana dan waktu pemrosesan antrian yang dibutuhkan juga cepat hanya memerlukan beberapa alamat IP untuk *client* dalam mengatur alokasi *upload* dan *download*. *Simple Queue* merupakan teknik antrian dengan metode FIFO (*First Input First Output*).

Teknik antrian FIFO adalah paket data yang pertama datang akan diproses terlebih dahulu dan dimasukkan ke dalam antrian, kemudian dikeluarkan sesuai dengan urutan

kedatangannya. Gambar 2.13 merupakan ilustrasi aliran paket data teknik antrian FIFO (*First Input First Output*).



Gambar 2.13 ilustrasi aliran paket data teknik antrian FIFO (*First Input First Output*)

Sumber: Manajemen Bandwidth Internet dan Intranet, Budi Santosa, ST

2. Konsep Dasar Teknik Antrian Queue Tree

Queue Tree merupakan teknik antrian manajemen *bandwidth* pada mikrotik routerboard RB750. Teknik antrian *queue tree* menggunakan metode PCQ (*Per Connection Queue*). Metode PCQ memiliki kemampuan membagi paket data yang diterima sesuai dengan koneksi setiap prioritas tujuannya. Setiap paket data yang terbagi setiap koneksinya dimasukkan ke dalam antrian dan dikeluarkan oleh penjadwalan (*scheduler*).

Algoritma Teknik Antrian Queue Tree Metode PCQ (*Per Connection Queue*)

Algoritma Teknik Antrian Queue Tree adalah metode PCQ (*Per Connection Queue*) sebagai berikut:



Gambar 2.14 Algoritma PCQ (*Per Connection Queue*)

Sumber: www.mikrotik.com/manual:queue

- **Mark Packet**

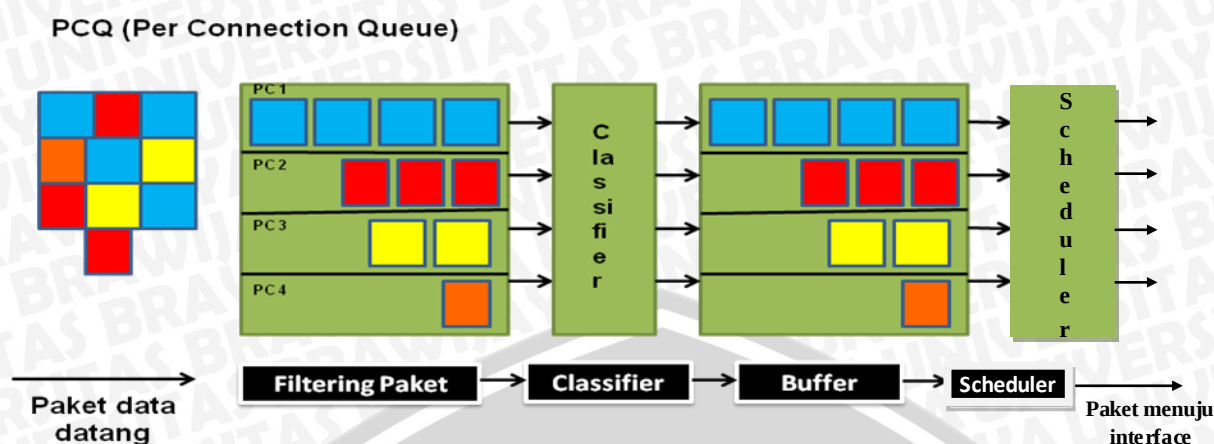
Mark packet bertugas untuk menandai paket data yang akan diproses ke antrian.

- **Firewall**

Firewall bertugas untuk menyeleksi paket sesuai dengan klasifikasi kelasnya.

- **Mangle**

Mangle bertugas untuk pembatasan *bandwidth*.



Gambar 2.15 Ilustrasi aliran paket data teknik antrian *queue tree* metode PCQ (Per Connection Queue)
Sumber: www.mikrotik.com/manual:queue

2.9 Konsep Dasar MikroTik RouterOS™

MikroTik RouterOS™ merupakan sistem operasi Linux yang digunakan sebagai *network router*. Administrasi pengaturan *mikrotik router* dapat dilakukan melalui *Windows Application* yang sering disebut dengan *WinBox*.

2.9.1 Jenis-Jenis Mikrotik RouterOS

Terdapat dua jenis mikrotik RouterOS, diantaranya adalah sebagai berikut:

1. MikroTik RouterOS yang berbentuk *software* yang dapat di-download di www.mikrotik.com yang dapat diinstal pada PC.
2. MikroTik RouterOS yang berbentuk *hardware* (perangkat keras) yang khusus dikemas dalam *router board* yang didalamnya sudah terinstal MikroTik RouterOS.

2.10 Parameter Performansi Quality of Service (QoS)

Quality of Service (QoS) merupakan kemampuan jaringan untuk menyediakan layanan yang lebih baik pada trafik jaringan tertentu melalui teknologi yang berbeda-beda. QoS dapat dijadikan sebagai ukuran untuk menentukan baik atau buruknya kinerja suatu jaringan internet. Parameter-parameter yang digunakan untuk mengukur kinerja QoS adalah *probabilitas packet loss*, *delay*, dan *throughput*.

1. Packet Loss

Packet loss merupakan parameter yang menunjukkan banyaknya jumlah paket yang hilang atau tidak sampai ke tujuan ketika melakukan pengiriman data dari sumber ke tujuan. Paket yang hilang dapat disebabkan oleh beberapa kemungkinan, antara lain adalah:

- Terjadinya *over load* dalam jaringan.
- Tabrakan atau tumbukan dalam jaringan.
- Error yang terjadi pada media fisik.
- Pengiriman data pada waktu yang bersamaan dengan menggunakan sebuah saluran secara bersama-sama.

Semakin kecil nilai *packet loss* dalam suatu jaringan maka semakin baik pula kinerja yang dimiliki jaringan tersebut. Secara umum terdapat empat kategori penurunan performansi jaringan berdasarkan nilai *packet loss* sesuai dengan versi *TIPHON-Telecommunications and Internet Protocol Harmonization Over Networks*, yaitu seperti tampak pada tabel berikut:

Tabel 2.3 Tabel Kategori Packet loss

Kategori	Packet loss (%)
Sangat Bagus	$x = 0$
Bagus	0 - 3
Sedang	3 - 15
Buruk	15 - 25

Sumber: TIPHON

packet loss dapat dihitung dengan rumus :

$$\text{Packet loss (\%)} = \frac{N_{\text{packet loss}}}{N_{\text{packet}} + N_{\text{packet loss}}} \times 100\% \quad (2-1)$$

Keterangan:

$N_{\text{packet loss}}$ = jumlah paket *multimedia* yang hilang (paket)

N_{paket} = jumlah paket *multimedia* rata-rata (paket)

$\text{Header}_{\text{TCP}}$ = 20 byte

$\text{Header}_{\text{IP}}$ = 20 byte

$\text{Header}_{\text{ETH}}$ = 14 byte

2. Delay end-to-end

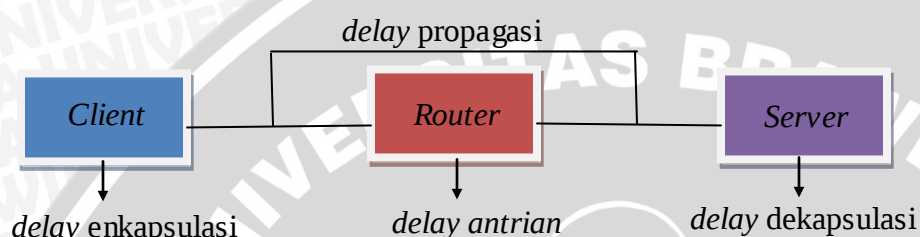
Delay end-to-end adalah waktu yang dibutuhkan untuk mengirimkan data dari sumber ke tujuan. Waktu tunda *end-to-end* yang terjadi dalam transmisi paket-paket data pada jaringan mengalami fluktuasi. Adanya variasi waktu tunda dalam transmisi layanan aplikasi internet menimbulkan masalah tersendiri, yaitu paket-paket yang datang terlambat yang dapat mengganggu data yang diterima oleh pelanggan. Kategori kinerja jaringan berdasarkan nilai *delay* dapat dilihat pada tabel berikut.

Tabel 2.4 Kategori kualitas *delay end-to-end*

Kategori	Delay (ms)
Bagus	$0 < t_{\text{end-to-end}} \leq 150$
Cukup, Dapat ditolerir	$150 < t_{\text{end-to-end}} \leq 400$
Buruk	$> 400 \text{ ms}$

Sumber: ITU G.114, 2011

Secara teori, *delay* pada jaringan dapat dibedakan dalam beberapa komponen yang berbeda. Agar lebih mudah memahami *delay end-to-end* berikut adalah blok diagram perhitungan *delay end-to-end*:



Gambar 2.16 Blok Diagram *delay end to end*

Total *delay end-to-end* adalah:

$$t_{\text{end to end}} = t_{\text{enkapsulasi}} + t_{\text{transmisi}} + t_{\text{propagasi}} + t_{\text{antrian}} + t_{\text{dekapsulasi}} \quad (2-2)$$

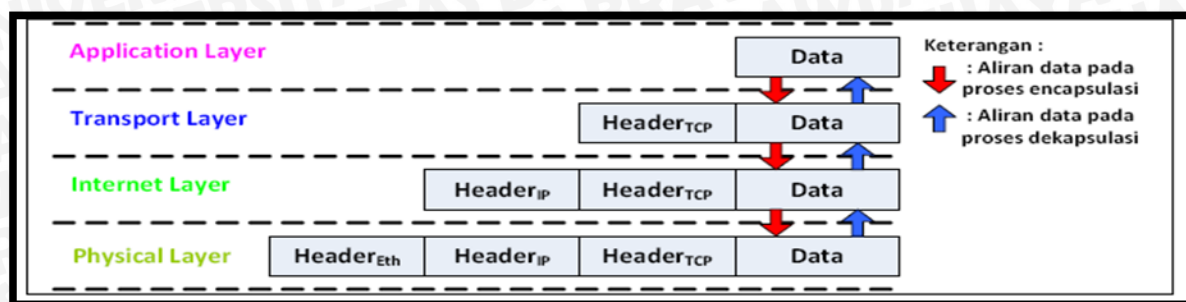
Keterangan :

- $t_{\text{end to end}}$ = *delay end-to-end* (s)
- $t_{\text{enkapsulasi}}$ = *delay enkapsulasi* (s)
- $t_{\text{transmisi}}$ = *delay transmisi* (s)
- $t_{\text{propagasi}}$ = *delay propagasi* (s)
- t_{antrian} = *delay antrian* (s)
- t_{dec} = *delay dekapsulasi* (s)

Delay end-to-end ditentukan berdasarkan arsitektur sistem dan merupakan penjumlahan *delay-delay* yang ada dalam perjalanan paket dari *host* sumber ke *host* tujuan. Penjumlahan *delay-delay* dari *host* ke sumber adalah sebagai berikut:

a. Delay Enkapsulasi

Enkapsulasi adalah proses pemaketan data sebelum dikirim ke *host* tujuan. *Delay enkapsulasi* adalah waktu yang dibutuhkan untuk proses pemaketan data sebelum dikirim ke *host* tujuan. Gambar 2.17 merupakan proses enkapsulasi dan dekapsulasi paket data.



Gambar 2.17 Proses Enkapsulasi dan Dekapsulasi Paket Data

Sumber: <http://www.citap.com/documents/tcp-ip/tcpip011.htm>

Persamaan delay enkapsulasi:

$$t_{enc} = \frac{L_{headerTCP} + L_{headerIP} + L_{headereth}}{C_{pros}} \quad (2-3)$$

$$C_{pros} = \frac{N_k}{T_k} \quad (2-4)$$

Keterangan:

C_{pros} = kecepatan pemrosesan pada terminal pengirim (bps);

N_k = jumlah total data yang dikirimkan (bit);

T_k = waktu pengiriman total data (s)

$Header_{TCP}$ = 20 byte;

$Header_{IP}$ = 20 byte;

$Header_{ETH}$ = 14 byte

b. Delay transmisi

Delay transmisi adalah waktu yang dibutuhkan untuk meletakkan sebuah paket multimedia ke media transmisi. Dipengaruhi ukuran paket data dan kapasitas kanal intranet.

Delay transmisi ditentukan dengan persamaan (2-9):

$$t_t = \frac{(L+L')}{c} = \frac{W_{frame}}{c} \quad (2-5)$$

Keterangan:

t_t = delay transmisi (s);

L = panjang paket data (byte)

L' = panjang header (byte);

C = kecepatan media transmisi (bps);

W_{frame} = panjang frame Ethernet (byte)

c. Delay Propagasi

Delay propagasi adalah waktu yang dibutuhkan untuk merambatkan paket *multimedia* melalui media transmisi dari *server* ke *client*. Tabel 2.5 merupakan kecepatan propagasi pada berbagai media.

Tabel 2.5 Kecepatan Propagasi pada Berbagai Media

Jenis Kabel	Kecepatan Propagasi	Keterangan
Thick Coax	0.77c	c merupakan kecepatan cahaya = 3.10^8 m/s
Thin Coax	0.65c	
Twisted Pair	0.59c	
Fiber Optic	0.66c	
UTP Cable	0.64c	

Sumber : <http://stason.org/TULARC/networking/lans-ethernet/3-11-What-is-propagation-delay-Ethernet-Physical-Layer.html>

Delay propagasi ditentukan dengan persamaan berikut:

$$t_p = \frac{d_{\max}}{v} \quad (2-6)$$

Keterangan :

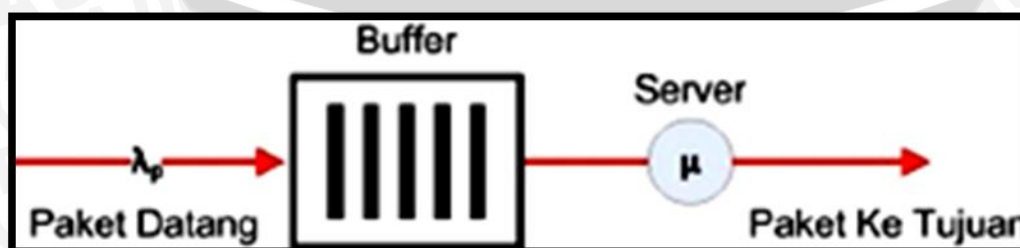
t_p = delay propagasi (s)

$d_{\max}$ = Panjang maksimum kabel UTP (m)

v = Kecepatan Propagasi (m/s)

d. Delay Antrian

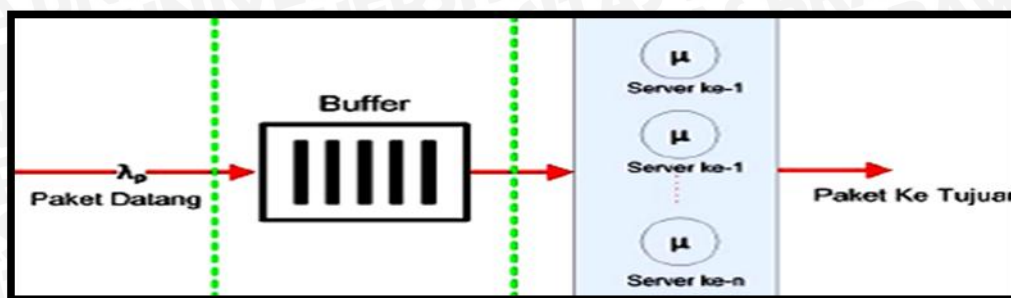
Delay antrian adalah waktu di mana paket data berada dalam antrian untuk diproses oleh *server*, lamanya waktu antrian bergantung pada kecepatan saluran dan kondisi antrian. Delay antrian dihitung dengan menggunakan model antrian M/M/1. M pertama menunjukkan distribusi kedatangan *Poisson*, M kedua menunjukkan bahwa jumlah *server* adalah tunggal yang diwakili oleh sebuah *node* yang dilengkapi *buffer*. Gambar 2.18 menunjukkan model antrian M/M/1. Gambar 2.19 Menunjukkan model antrian M/M/S.



Gambar 2.18 Model Antrian M/M/1

Sumber : Mischa Schwartz, 1987:22

Gambar 2.19 Menunjukkan model antrian M/M/S



Gambar 2.19 Model Antrian M/M/S

Sumber: Misha Schwartz, 1987:22

Kecepatan kedatangan paket pada server ditentukan persamaan Little's Formula. Persamaannya adalah sebagai berikut:

$$\lambda_p = \frac{N}{T} \quad (2-7)$$

Keterangan:

λ_p = kecepatan kedatangan paket pada server (paket/s)

N = Total paket yang dikirim (paket)

T = Waktu pengiriman paket total (s)

Kecepatan pelayanan ditentukan oleh persamaan di bawah ini (Mischa Schwartz):

$$\mu = \frac{C}{L_t} \quad (2-8)$$

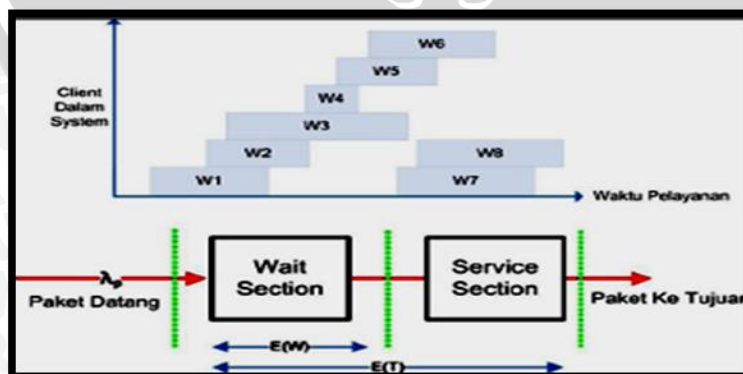
Keterangan:

μ = Kecepatan pelayanan server (paket/s)

C = Kapasitas kanal (bps)

L_t = Panjang paket data (bit/paket)

Gambar 2.20 merupakan implementasi Little's Formula pada system antrian



Gambar 2.20 Implementasi Little's Formula Pada Sistem Antrian

Sumber: Mischa Schwartz, 1987:42

Keterangan:

$E(W)$ merupakan waktu antrian atau waktu tunggu sebelum dilakukan pelayanan pada sisi server.

$E(T)$ merupakan waktu paket dalam system atau *delay* antrian. Berdasarkan gambar maka *delay* antrian rata-rata $E(T)$ dapat ditentukan dengan persamaan (Mischa Schwartz, 1987:42) di bawah ini:

$$E(T) = E(W) + \frac{1}{\mu} \quad (2-9)$$

dimana:

$$E(W) = \frac{1}{\mu - \lambda_p} - \frac{1}{\mu} \quad (2-10)$$

Sehingga besarnya *delay* antrian $E(T)$ dapat dinyatakan sebagai berikut:

$$E(T) = \frac{1}{\mu - \lambda_p} \quad (2-11)$$

Keterangan:

$E(T)$ = *delay* antrian pada server (s)

$E(W)$ = *delay* tunggu pada server (s)

λ_p = kecepatan kedatangan paket pada server (paket/s)

μ = kecepatan pelayanan server (paket/s)

3. *Throughput*

Throughput merupakan parameter yang digunakan untuk mengetahui jumlah data yang diterima dalam keadaan baik terhadap waktu total transmisi yang dibutuhkan dari sumber ke penerima. Berikut adalah perhitungan secara matematis *throughput*:

$$\lambda = \frac{L}{t_v} \quad (2-12)$$

dengan:

λ = *throughput* (bps)

t_v = waktu total transmisi untuk mengirimkan paket yang benar (s)

L = panjang paket data yang benar diterima (bit)

c. Pengujian Kualitas Jaringan Menggunakan Aplikasi *Wireshark*

Aplikasi *Wireshark* merupakan aplikasi yang dapat menangkap paket-paket jaringan pada sisi *client* yang berguna untuk menampilkan informasi secara detail pada paket tersebut.

Wireshark mempunyai banyak fitur dan kelebihan diantaranya adalah sebagai berikut:

1. Aplikasi *Wireshark* bersifat *open source* untuk menganalisis paket jaringan.
2. Mampu menangkap paket data secara langsung dari sebuah *network interface*.
3. Mampu menampilkan informasi secara detail mengenai hasil *capture* pada sebuah jaringan.
4. Mampu menampilkan hasil statistika dari hasil *capture* pada sebuah jaringan.
5. Tersedia untuk Linux dan Windows.

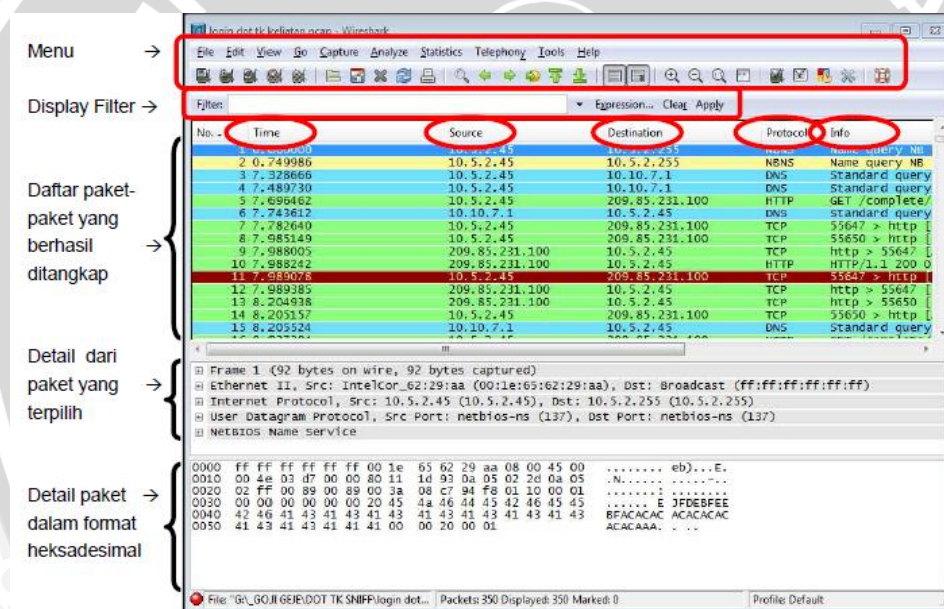
Gambar 2.21 merupakan logo dari aplikasi *Wireshark*:



Gambar 2.21 Logo Aplikasi *Wireshark*

Sumber: Tutorial *Wireshark*

Pada gambar 2.22 merupakan contoh tampilan *wireshark* yang sedang meng-*capture* paket-paket pada suatu sistem jaringan:



Gambar 2.22 Contoh Tampilan *Wireshark* yang Sedang Meng-*capture* Paket-Paket Pada Suatu Sistem Jaringan

Sumber: Tutorial *Wireshark*

Keterangan:

1. Menu : Menu-menu yang tersedia di *Wireshark*
2. *Display Filter* : Sebuah kolom yang dapat diisi dengan sintaks-sintaks untuk membatasi paket-paket apa saja yang akan ditampilkan pada list paket.
3. Daftar Paket : Menampilkan paket-paket yang berhasil ditangkap oleh aplikasi *Wireshark*, berurutan dari paket pertama yang ditangkap, dan seterusnya.
4. Detail Paket : Menampilkan detail paket yang terpilih pada daftar paket.

5. **Detail Heksa** : Menampilkan detail paket yang terpilih yang ditampilkan dalam bentuk heksa.

Pada daftar bagian daftar Paket, terdapat kolom-kolom seperti berikut ini:

1. **Time** : Menampilkan waktu saat paket-paket tersebut ditangkap.
2. **Source** : Menampilkan alamat IP sumber dari paket data tersebut.
3. **Destination** : Menampilkan alamat IP tujuan dari paket data tersebut.
4. **Protocol** : Menampilkan protokol yang digunakan pada sebuah paket data.
5. **Info** : Menampilkan informasi secara detail tentang paket data tersebut.

2.11 Komponen Jaringan Komputer

Komponen jaringan komputer yang digunakan dalam proses pengujian sistem, yaitu : *Mikrotik Routerboard750*, *Switch*, *Network Interface Card*, *Desktop Komputer*, Kabel UTP, dan RJ45.

a. Mikrotik Routerboard750

Mikrotik Routerboard750 merupakan *router indoor* yang berbentuk *hardware* (perangkat keras) yang didalamnya sudah terinstal *MikroTik RouterOS* level 4. Gambar 2.23 merupakan *mikrotik routerboard750*. Spesifikasi dari *Mikrotik Routerboard750* ditunjukkan pada Tabel 2.6.



Gambar 2.23 Mikrotik Routerboard 750
Sumber: wirelessconnect.eu

Tabel 2.6 Spesifikasi Mikrotik Routerboard 750

Spesifikasi	Keterangan
Product Code	RB750
LAN Ports	5 port Ethernet 10/100
Power Jack	10-28 V
Dimensions	113x89x28 mm
Operating System	RouterOS Level 4

Sumber: www.mikrotik.com

b. Switch

Switch merupakan perangkat keras yang berfungsi untuk menghubungkan komponen-komponen pada sistem jaringan dari *server mikrotik* ke *client*. Penggunaan *switch* dibutuhkan untuk melanjutkan informasi data secara tepat. Pada perancangan ini digunakan *Switch InfoSmart*. Gambar 2.24 merupakan *switch infoSmart*. Spesifikasi dari *Switch InfoSmart* tipe *INS800N* ditunjukkan pada Tabel 2.7.



Gambar 2.24 Switch InfoSmart tipe INS800N
Sumber: www.alibaba.com

Tabel 2.7 Spesifikasi Switch InfoSmart tipe INS800N

Spesifikasi	Keterangan
Port	8 Port Ethernet 10/100 Mbps
Flow Control	Half duplex and Full Duplex IEEE 802.3
Input Voltage	100-240 V AC
Input Frequency	50-60 Hz
Temperature	0 - 50°C
Packet Buffer	1536 bytes

Sumber: www.indonetwork.co.id

c. Network Interface Card

Network Interface Card (NIC) adalah sebuah perangkat keras yang berfungsi sebagai *interface* dari komputer ke sebuah jaringan computer sehingga dapat mentransmisikan informasi ke jaringan dengan media transmisi berupa kabel UTP Cat.5. Prinsip kerja NIC adalah mengubah aliran data paralel dalam *bus* komputer menjadi bentuk data serial kemudian dipaketkan menjadi beberapa *frame*. Proses pembuatan *frame* ini, akan menambahkan *header* terhadap data yang hendak dikirimkan, yang mengandung alamat, pensinyalan, dan layanan yang digunakan. *Frame-frame* tersebut kemudian diubah menjadi pulsa-pulsa elektronik. Spesifikasi *Network Interface Card* ditunjukkan pada tabel 2.8.



Gambar 2.25 Network Interface Card
Sumber: everylearning.blogspot.com

Tabel 2.8 Spesifikasi Network Interface Card

Spesifikasi	Keterangan
Standards and Protocols	Standards IEEE 802.3, Protocol TCP/IP
Interface	32-bit PCI Slot
Transmission Rate	10-100 Mbps (Half-Full Duplex Mode)
Port	1x10/100 Mbps RJ45
LED Indicator	Link/Act
Network Media	10Base-T(UTP CAT 3/4/5), 100 Base-Tx(UTP 5/5e)

Sumber: www.alnect.com

d. Desktop Komputer

Desktop komputer adalah perangkat keras sistem elektronik yang berfungsi untuk memanipulasi data dengan cepat dan tepat serta dirancang dan diorganisasikan agar secara otomatis menerima dan menyimpan data *input*, memprosesnya, dan menghasilkan output di bawah pengawasan suatu langkah-langkah instruksi program yang tersimpan di didalam penyimpanannya (Sanderes, 1985). Pada proses perancangan komputer sistem *manajemen bandwidth* memiliki beberapa fungsi diantaranya sebagai :

- Perangkat untuk menyediakan layanan *system manajemen banadwidth*, yang meliputi pembentukan proses teknik antrian *simple queue* dan *queue tree* menuju *client*, proses pengunduhan data menggunakan IDM, remote mikrotik menggunakan WinBox.

Spesifikasi *Desktop* komputer ditunjukkan pada tabel 2.9.



Gambar 2.26 Desktop Komputer

Sumber: sajakan.blogspot.com

Tabel 2.9 Spesifikasi Desktop Komputer

Spesifikasi	Keterangan
Processor	Intel Celeron Dual Core
RAM	2 GB
Operating system	Windows 7 Ultimate
NIC	10/100 Mbps Ethernet
Port Connector USB	2 port

Sumber: www.kliknklik.com

e. Kabel UTP dan Konektor RJ-45

Kabel UTP dengan konektor RJ-45 merupakan perangkat keras yang berfungsi sebagai sistem pengkabelan pada system manajemen *bandwidth* yang digunakan secara umum dengan standart IEEE 802.3. Pengurutan kabel UTP dapat dibedakan menjadi 2 macam:

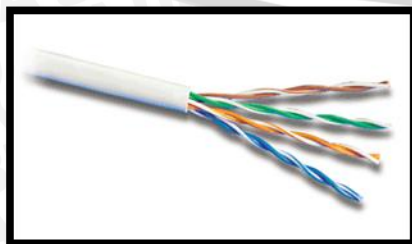
- Kabel *straight*

Digunakan untuk hubungan antara PC-PC, hub-hub, switch-switch, router-router.

- Kabel *cross*

Digunakan untuk hubungan antara PC-hub, PC-Switch, Switch-Router, PC-Modem.

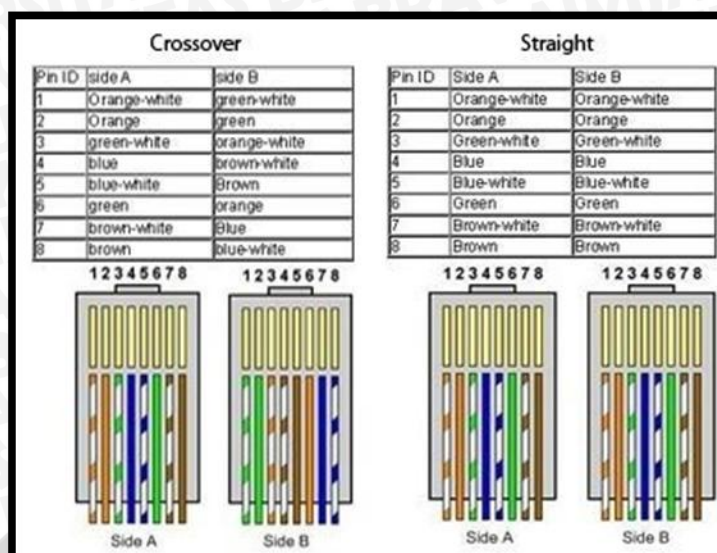
Pada perancangan ini digunakan kabel UTP Cat.5 dengan konfigurasi *straight* ditunjukkan pada Gambar 2.21. Spesifikasi kabel UTP ditunjukkan pada tabel 2.10.



(a)



(b)



(c)

Gambar 2.27 (a) Kabel UTP (b) Konektor RJ-45 (c) Konfigurasi Kabel Crossover dan Straight

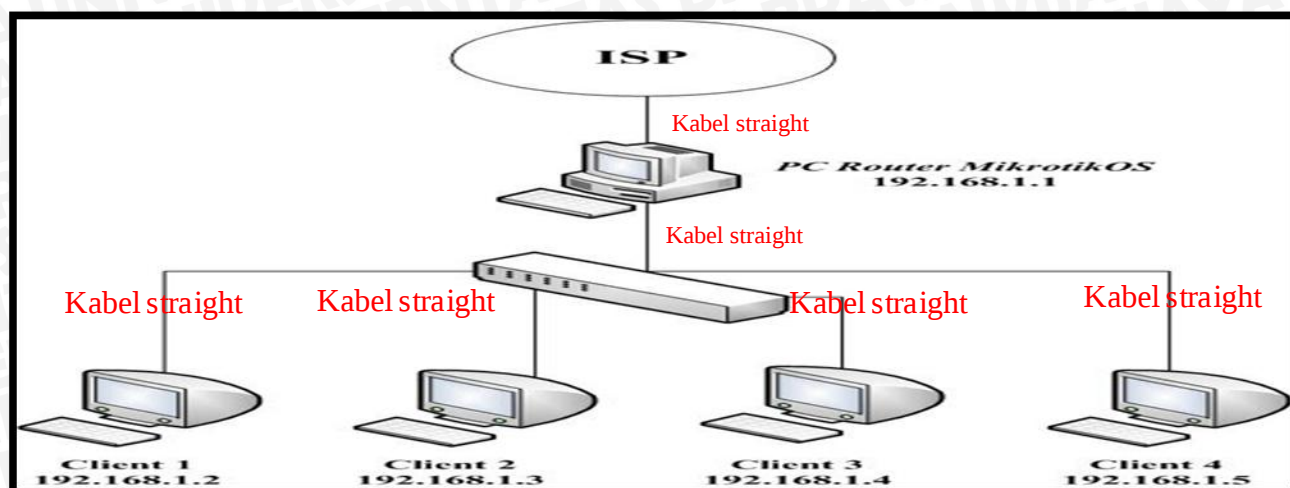
Sumber : <http://www.beldenapac.com/catalogbrochures.html>

Tabel 2.10 Spesifikasi Kabel UTP

Spesifikasi	Keterangan
Impedansi Karakteristik	100 $\Omega \pm 15\%$
Kecepatan Propagasi	0.64c m/s
Delay Propagasi	4.8 – 5.3 ns/m
Kapasitansi	5.2 pF/m
Induktansi	525nH/m
Diameter Kabel	0.51054 mm
Ketebalan Isolasi	0.245 mm
Arus Perkonduktor	0.577 A
Temperatur Kerja	-55°C sampai 60°C

Sumber : <http://www.beldenapac.com/catalogbrochures.html>

Gambar 2.28 merupakan konfigurasi pengkabelan sistem manajemen *bandwidth*.



Gambar 2.28 Konfigurasi Pengkabelan Sistem Manajemen *Bandwidth*