

## BAB 2

### LANDASAN TEORI

#### 2.1 Jaringan Komputer

Menurut Sutedjo dan Oetomo (2006, p7), jaringan komputer adalah sekelompok komputer otonom yang saling menggunakan protokol komunikasi melalui media komunikasi sehingga dapat berbagi data, informasi, program aplikasi, dan perangkat keras seperti *printer*, *scanner*, *CD-Drive* ataupun *hard disk*, serta memungkinkan untuk saling berkomunikasi secara elektronik.

Sejumlah potensi jaringan komputer antara lain:

1. Komunikasi

Jaringan komputer memungkinkan terjadinya komunikasi antar pemakai komputer. Selain itu, tersedia aplikasi *teleconference* yang memungkinkan dilakukannya rapat atau pertemuan tanpa harus meninggalkan meja kerja.

2. Mengintegrasikan data

Jaringan komputer diperlukan untuk mengintegrasikan data antar komponen-komponen *client* sehingga dapat diperoleh suatu data yang relevan.

### 3. Perlindungan data dan informasi

Jaringan komputer memudahkan upaya perlindungan data yang terpusat pada *server*, melalui pengaturan hak akses dari para pemakai serta penerapan sistem *password*.

### 4. Berbagi peralatan

Jaringan komputer memungkinkan penggunaan bersama peralatan komputer berbagai merek, yang semula tersebar di berbagai ruangan, unit, dan departemen sehingga meningkatkan efektivitas dari penggunaan sumber daya tersebut.

### 5. Sistem terdistribusi

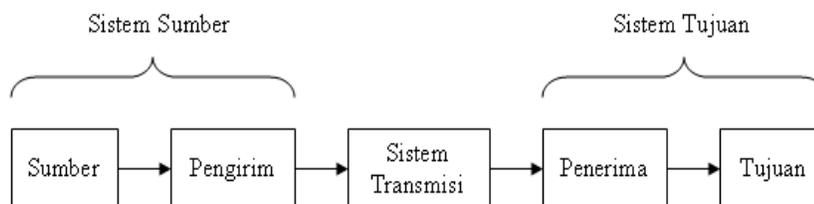
Jaringan komputer dimanfaatkan pula untuk mendistribusikan proses dan aplikasi sehingga dapat mengurangi terjadinya *bottleneck* atau tumpukan pekerjaan pada satu bagian.

### 6. Keteraturan aliran informasi

Jaringan komputer mampu mengalirkan data-data komputer *client* dengan cepat untuk diintegrasikan dalam komputer *server*. Selain itu, jaringan mampu untuk mendistribusikan informasi secara kontinu kepada pihak-pihak terkait yang membutuhkannya.

#### **2.1.1 Model Komunikasi**

Menurut Stallings (2001, p5), kegunaan sistem komunikasi ini adalah menjalankan pertukaran data antara dua pihak. Elemen-elemen dasar model komunikasi tersebut dapat digambarkan sebagai berikut:



Gambar 2.1 Model Komunikasi

(Sumber: Stallings (2001, p5))

Keterangan dari gambar di atas yaitu:

1. *Source* (Sumber)

Alat ini menghasilkan data untuk ditransmisikan atau dikirim. Contohnya telepon dan *Personal Computer* (PC).

2. *Transmitter* (Pengirim)

Biasanya data yang dihasilkan dari sistem sumber tidak ditransmisikan secara langsung dalam bentuk aslinya. Sebuah *transmitter* cukup memindahkan dan menandai informasi dengan cara yang sama seperti menghasilkan sinyal-sinyal elektromagnetik yang dapat ditransmisikan melewati beberapa sistem transmisi berurutan.

3. *Transmission System* (Sistem Transmisi)

Berupa jalur transmisi tunggal (*single transmission line*) atau jaringan kompleks yang menghubungkan antara sumber dengan tujuan.

#### 4. *Receiver* (Penerima)

*Receiver* menerima sinyal dari sistem transmisi dan menggabungkannya ke dalam bentuk tertentu yang dapat ditangkap oleh tujuan.

#### 5. *Destination* (Tujuan)

*Destination* berfungsi untuk menangkap data yang dihasilkan oleh *receiver*.

### 2.1.2 Peralatan Jaringan Komputer

Peralatan jaringan komputer adalah alat yang digunakan untuk menghubungkan *end-user device* ke jaringan, memperluas jangkauan jaringan, melakukan konversi format data, mengatur *transfer* data, dan fungsi jaringan lainnya (<http://ilmukomputer.org/2006/08/25/tutorial-komputer-dan-jaringan/>).

Contoh-contoh peralatan jaringan komputer yaitu:

#### 1. *Modem*

*Modem* (*modulator–demodulator*) digunakan untuk mengubah informasi *digital* menjadi sinyal *analog*. *Modem* mengubah tegangan bernilai biner menjadi sinyal *analog* dengan melakukan *encoding* data *digital* ke dalam frekuensi *carrier*. *Modem* juga dapat mengubah kembali sinyal *analog* yang termodulasi menjadi data *digital*, sehingga informasi yang terdapat di dalamnya dapat dimengerti oleh komputer. Proses ini disebut demodulasi. Berdasarkan peletakkannya, *modem* dibagi menjadi dua tipe, yaitu:

- *Modem* eksternal



Gambar 2.2 *Modem* Eksternal

(Sumber: <http://balicomp.indonetwork.co.id/433710/1456tr-external-modem.htm>)

- *Modem* internal



Gambar 2.3 *Modem* Internal

(Sumber:

<http://www.frontierpc.com/ProductDetails.aspx?eId=10217904>)

## 2. *Repeater*

*Repeater* merupakan peralatan jaringan komputer yang digunakan untuk memperkuat kembali sinyal komunikasi jaringan. Setelah melalui media transmisi, sinyal dapat melemah. *Repeater* bertugas untuk memperkuat kembali sinyal tersebut sehingga dapat ditransmisikan lebih jauh. *Repeater* tidak melakukan pengambilan keputusan apapun mengenai pengiriman sinyal. *Repeater* bekerja dengan menerima, memperkuat, kemudian meneruskan sinyal

yang diterima agar dapat melewati media jaringan dengan jangkauan yang lebih jauh.



Gambar 2.4 *Repeater*

(Sumber: <http://www.edgarsson.co.uk/lone.htm>)

### 3. *Hub*

*Hub* merupakan peralatan jaringan komputer yang berfungsi untuk menerima sinyal dari satu komputer dan mentransmisikannya ke komputer yang lain. *Hub* mengambil bit-bit yang datang dari satu *port* dan mengirimkan salinannya ke setiap *port* yang lain. Setiap *host* yang tersambung ke *hub* akan melihat paket ini, tetapi hanya *host* yang dituju saja yang akan memprosesnya. Hal ini dapat mengakibatkan masalah *network traffic* karena paket yang dituju ke satu *host* sebenarnya dikirim ke semua *host*.



Gambar 2.5 *Hub*

(Sumber: <http://www.global-b2b->

[network.com/b2b/96/530/page5/34489/ethernet\\_hub.html](http://www.global-b2b-network.com/b2b/96/530/page5/34489/ethernet_hub.html))

#### 4. *Bridge*

*Bridge* merupakan peralatan jaringan komputer yang digunakan untuk memisahkan suatu jaringan yang luas menjadi jaringan-jaringan yang lebih kecil. *Bridge* sangat berguna untuk menghubungkan beberapa LAN agar dapat mencakup daerah yang lebih luas atau membagi sebuah LAN besar menjadi beberapa LAN yang lebih kecil untuk mengurangi *traffic* yang melalui masing-masing LAN. Tugas *bridge* adalah melakukan pengambilan keputusan apakah paket harus diteruskan ke jalur yang berikutnya atau tidak. Ketika *bridge* menerima paket dari jaringan, *bridge* akan memeriksa *Media Access Control (MAC) address* tujuannya dan memeriksa *MAC address* tersebut pada *bridge table* yang dimiliki. *MAC address* adalah sebuah alamat jaringan yang mewakili node tertentu pada jaringan. *Bridge* kemudian melakukan proses pengambilan keputusan sebagai berikut:

- Jika tujuan berada pada jalur yang sama dengan jalur paket, *bridge* tidak akan mengirimkan paket ke jalur yang lain. Proses ini disebut *filtering*.
- Jika tujuan berada pada jalur yang berbeda, maka *bridge* akan meneruskan paket ke jalur yang dituju.
- Jika *MAC address* tujuan tidak diketahui, *bridge* akan meneruskan paket ke semua jalur kecuali jalur asal paket.



Gambar 2.6 *Bridge*

(Sumber:[http://www.learnthat.com/certification/learn.asp?id=432  
&index=56](http://www.learnthat.com/certification/learn.asp?id=432&index=56))

#### 5. *Switch*

*Switch* merupakan peralatan jaringan komputer yang bekerja pada *layer 2* model OSI, yang mampu melakukan manajemen *transfer* data yaitu hanya meneruskan data ke segmen yang dituju. *Switch* tidak melakukan konversi format data. *Switch* mempelajari *host* mana saja yang terhubung ke suatu *port* dengan membaca MAC *address* asal yang ada di dalam paket, kemudian *switch* membuka sirkuit *virtual* antara *node* sumber dengan *node* tujuan. Dengan demikian, komunikasi dua *port* tersebut tidak mempengaruhi *traffic* dari *port* lain. Hal tersebut membuat LAN menjadi lebih efisien.



Gambar 2.7 *Switch*

(Sumber:

[http://www.astrindo.co.id/Products/Networking/Images/71\\_m.jpg](http://www.astrindo.co.id/Products/Networking/Images/71_m.jpg))

## 6. Router

*Router* mempunyai semua kemampuan peralatan jaringan komputer lainnya. *Router* dapat memperkuat sinyal, mengkonsentrasikan beberapa koneksi, melakukan konversi format transmisi data, dan mengatur *transfer* data. Selain itu, *router* juga bisa melakukan koneksi ke WAN, sehingga dapat menghubungkan LAN yang terpisah jauh. *Router* bertugas melakukan *routing* paket data dari sumber ke tujuan pada LAN dan menyediakan koneksi ke WAN. Dalam lingkungan LAN, *router* membatasi *broadcast* dan membagi jaringan dengan menggunakan struktur *subnetwork*.



Gambar 2.8 Router

(Sumber:<http://www.lakewoodconferences.com>)

## 7. Access Point

*Access Point* (AP) berperan sebagai sentral *hub* pada infrastruktur WLAN (*Wireless LAN*). AP dilengkapi dengan antena dan menyediakan koneksi tanpa kabel pada daerah tertentu yang disebut *cell*.



Gambar 2.9 *Access Point*

(Sumber: <http://www.cibo-grupa.hr/price%20list/network/network.htm>)

### 2.1.3 Macam-Macam Jaringan

Menurut Setiawan (2003, p154), jaringan terdiri dari dua kategori utama, yaitu :

#### 1. *Local Area Network* (LAN)

LAN merupakan jaringan data yang memiliki kecepatan tinggi, memiliki sifat toleransi kesalahan dan mempunyai cakupan area geografis yang sempit.

Ciri-ciri LAN:

- Beroperasi pada area terbatas.
- Memiliki kecepatan *transfer* yang tinggi.
- Dikendalikan secara privat oleh *administrator* lokal.
- Menghubungkan secara fisik alat-alat yang berdekatan.

#### 2. *Wide Area Network* (WAN)

WAN merupakan gabungan dari beberapa LAN yang terhitung dalam jarak jangkauan yang jauh, misalnya antar kota dan antar negara.

Ciri-ciri WAN:

- Beroperasi pada area yang luas.
- Memiliki kecepatan *transfer* yang lebih rendah daripada LAN.
- Menghubungkan alat-alat yang terpisah dalam jarak jauh bahkan global.

## 2.2 Arsitektur Jaringan

Pada dasarnya suatu arsitektur jaringan adalah aturan-aturan atau *standarisasi* dari berbagai macam protokol dan format-pesan ([http://mti.ugm.ac.id/~yudhistira/ResourceMTI/Tutorial/Cawu01/Computer%20Network/ARSITEKTUR\\_indo.ppt](http://mti.ugm.ac.id/~yudhistira/ResourceMTI/Tutorial/Cawu01/Computer%20Network/ARSITEKTUR_indo.ppt)). Dengan kata lain, arsitektur jaringan merupakan gambaran umum yang berisi sekumpulan aturan yang menerangkan gambaran kerja jaringan untuk mengendalikan komunikasi data antara dua atau lebih komputer. Secara umum, terdapat dua arsitektur jaringan, yaitu model OSI dan model TCP/IP.

### 2.2.1 Model OSI

Menurut Stallings (2001, p21), model *Open Systems Interconnection* (OSI) yang dikembangkan oleh *International Organization for Standardization* (ISO) sebagai model untuk arsitektur komunikasi komputer, serta sebagai kerangka kerja bagi pengembangan standar-standar *protocol*.

Model OSI terdiri dari tujuh lapisan, yaitu:

### 1. *Physical Layer*

Lapisan *physical* mencakup *physical interface* di antara peralatan-peralatan yang terhubung ke jaringan dan aturan bit dilewatkan dari satu ke yang lain. Lapisan *physical* memiliki empat karakteristik penting, yaitu:

- Mekanis

Berkaitan dengan properti fisik dari *interface* ke media transmisi.

- Elektris

Berkaitan dengan tampilan bit. Misalnya dalam hal tingkatan-tingkatan voltase serta laju transmisi bit.

- Fungsional

Menentukan fungsi-fungsi yang ditampilkan oleh sirkuit tunggal dari *interface* fisik di antara sebuah sistem dengan media transmisi.

- Prosedural

Menentukan rangkaian kejadian dimana arus bit berpindah melalui medium fisik.

### 2. *Data Link Layer*

Lapisan *data link* mengupayakan agar *link* fisik cukup baik dan menyediakan alat-alat untuk mengaktifkan, mempertahankan dan menonaktifkan *link*. Layanan pokok yang disediakan oleh lapisan

*data link* untuk lapisan yang lebih tinggi adalah deteksi dan pengontrol kesalahan. Jadi, dengan suatu *protocol* lapisan *data link* berfungsi sepenuhnya, lapisan yang lebih tinggi berikutnya bisa menerima transmisi bebas kesalahan melalui *link*. Bagaimanapun juga, bila komunikasi terjadi di antara dua sistem yang tidak dihubungkan secara langsung, koneksi akan terdiri dari sejumlah *data link* ganda, masing-masing berfungsi secara terpisah. Jadi, lapisan yang lebih tinggi tidak dikurangi tanggung jawabnya untuk mengontrol kesalahan.

### 3. *Network Layer*

Lapisan *network* menyediakan *transfer* informasi di antara ujung sistem melewati beberapa jaringan komunikasi berurutan. Pada lapisan ini, sistem komputer berdialog dengan jaringan untuk menentukan alamat tujuan dan meminta fasilitas jaringan tertentu, misalnya prioritas.

### 4. *Transport Layer*

Lapisan *transport* menyediakan suatu mekanisme perubahan data di antara ujung sistem. Layanan *transport* menjamin bahwa data yang dikirim bebas dari kesalahan secara bertahap dengan tidak mengalami duplikasi atau hilang. Lapisan ini juga dapat dikaitkan dengan mengoptimalkan penggunaan layanan jaringan dan menyediakan mutu layanan yang dapat diminta untuk entiti sesi. Ukuran dan kelengkapan sebuah *protocol transport* tergantung

dari seberapa diandalkannya atau tidak jaringan yang mendasari serta tergantung pada lapisan *network* layanan.

#### 5. *Session Layer*

Lapisan ini menyediakan cara bagi pertukaran data dan menyediakan berbagai pilihan mutu layanan. Selain itu, lapisan ini menyediakan mekanisme untuk mengontrol dialog di antara aplikasi pada ujung sistem.

Layanan-layanan yang disediakan oleh lapisan sesi adalah:

- Disiplin dialog (*Dialogue discipline*)

Dapat berupa dua saluran simultan (*full duplex*) atau dua saluran pilihan (*half duplex*).

- Pengelompokan (*Grouping*)

Aliran data dapat ditandai dengan cara menentukan kelompok data.

- *Recovery*

Lapisan sesi dapat menyediakan suatu mekanisme pemeriksaan agar jika terjadi kegagalan di antara *checkpoint*, sesi entiti dapat mentransmisikan kembali seluruh data mulai dari *checkpoint* terakhir.

#### 6. *Presentation Layer*

Lapisan ini menentukan format data yang dipindahkan di antara aplikasi dan menawarkan program-program aplikasi serangkaian layanan transportasi data. Selain itu, lapisan ini menentukan

*syntax* yang digunakan di antara entiti aplikasi serta menyediakan modifikasi seleksi dan *subsequent* dari representasi yang digunakan. Contohnya adalah kompresi dan enkripsi data.

#### 7. *Application Layer*

Layanan *application* menyediakan cara bagi program-program aplikasi untuk mengakses lingkungan OSI. Lapisan ini berisi fungsi-fungsi manajemen dan mekanisme-mekanisme yang umumnya berguna untuk mendukung aplikasi-aplikasi yang didistribusikan.

### 2.2.2 Model TCP/IP (*Internet Protocol Suite*)

Menurut Lukas (2006, p21), model TCP/IP merupakan hasil penelitian yang dibuat dan dikembangkan oleh *Defense Advance Research Project Agency* (DARPA) yang digunakan pada jaringan paket. Model TCP/IP secara umum dikenal sebagai *Internet Protocol Suite*. *Internet Protocol Suite* ini terdiri atas sekumpulan protokol dalam jumlah besar yang dijadikan sebagai standar internet. TCP/IP memungkinkan terjadinya komunikasi antara jaringan yang saling berhubungan dan dapat digunakan baik dalam LAN maupun WAN.

Ada lima *layer* yang dikenal dalam model TCP/IP, yaitu :

#### 1. *Physical Layer*

Lapisan ini meliputi tampilan fisik suatu perangkat transmisi data, misalnya *workstation* dan komputer dengan sebuah media transmisi atau jaringan. *Layer* ini berkaitan dengan spesifikasi dari

media transmisi yaitu signal yang dapat dilewati, kecepatan transmisi dan lainnya yang berkaitan dengan karakteristik media.

## 2. *Network Access Layer*

Lapisan ini mengatur pertukaran data antara *end system* dengan jaringan dimana dihubungkan. Komputer pengirim harus menyediakan jaringan dengan alamat komputer yang dituju, sehingga jaringan dapat mengirimkan data ke tujuan secara tepat.

## 3. *Internet Layer*

Lapisan ini berfungsi untuk menentukan prosedur-prosedur tertentu agar data dapat melintasi jaringan yang bermacam-macam. *Internet Protocol* (IP) dipergunakan dalam lapisan ini agar fungsi *routing* untuk berbagai macam jaringan dapat dipakai.

## 4. *Transport Layer* atau *Host to Host*

Lapisan ini bertugas untuk menyediakan mekanisme-mekanisme yang diperlukan agar seluruh data tiba di aplikasi tujuan dan data yang tiba tersebut sesuai dengan yang diperintahkan pada saat data dikirim.

## 5. *Application Layer*

Lapisan ini berisi fungsi logika yang akan digunakan pada seluruh aplikasi *user*. Untuk setiap jenis aplikasi yang berbeda, misalnya pengiriman file, diperlukan modul terpisah yang sesuai dengan aplikasi tersebut.

## 2.3 Topologi Jaringan

Menurut Setiawan (2003, p162), topologi jaringan adalah struktur jaringan untuk mengidentifikasi cara bagaimana simpul atau pusat di dalam jaringan saling berhubungan. Hubungan dalam jaringan sangat bergantung jenis aplikasi yang digunakan. Topologi jaringan menjelaskan susunan peletakan node dalam suatu jaringan dan bagaimana cara untuk mengaksesnya. Topologi jaringan dibagi menjadi dua jenis, yaitu *physical topology* dan *logical topology*. Setiap topologi jaringan mempunyai kelebihan dan kekurangan masing-masing.

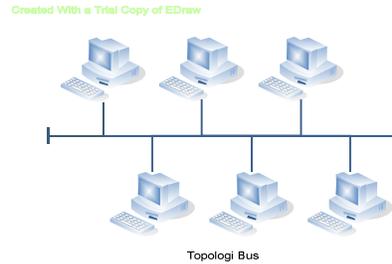
### 2.3.1 *Physical Topology*

*Physical topology* merupakan tata letak *node* dari sebuah jaringan dan bagaimana jaringan terhubung secara fisik ([http://en.wikipedia.org/wiki/Network\\_topology#Logical\\_topology](http://en.wikipedia.org/wiki/Network_topology#Logical_topology)).

Jenis-jenis dari topologi fisik (*physical topology*):

#### 1. *Bus Topology*

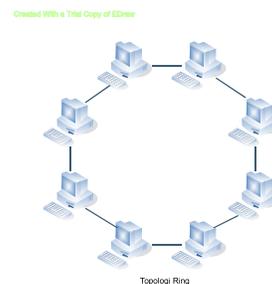
Topologi ini memiliki semua *node* yang terhubung langsung ke satu sambungan (*link*) dan tidak memiliki hubungan yang lain antar *node*. Sebuah topologi *bus* memungkinkan setiap perangkat jaringan untuk melihat semua sinyal dari perangkat-perangkat yang lain.



Gambar 2.10 *Bus Topology*

## 2. *Ring Topology*

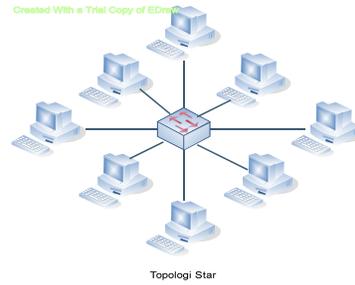
Topologi ini merupakan sebuah cincin tertutup (*closed ring*) yang terdiri dari *node-node* dan kabel (*link*), dimana masing-masing *node* terhubung hanya dengan dua *node* yang disebelahnya, sehingga pada akhirnya membentuk *loop* tertutup.



Gambar 2.11 *Ring Topology*

## 3. *Star Topology*

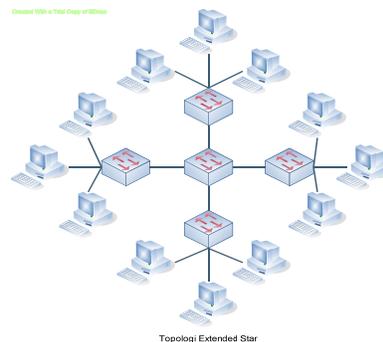
Topologi jaringan dimana *end point* pada sebuah jaringan dihubungkan dengan sebuah *central hub* atau *switch* oleh jaringan yang sedang berjalan. Topologi ini mempunyai sentral *node*, dimana semua hubungan ke *node* yang lain melalui sentral *node* tersebut.



Gambar 2.12 *Star Topology*

#### 4. *Extended Star Topology*

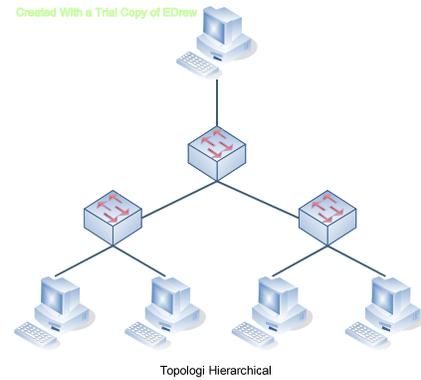
Topologi ini memiliki inti sebuah *star topology*, dengan masing-masing *end node* dari topologi inti bertindak sebagai pusat dari topologi *star*-nya sendiri.



Gambar 2.13 *Extended Star Topology*

#### 5. *Hierarchical topology*

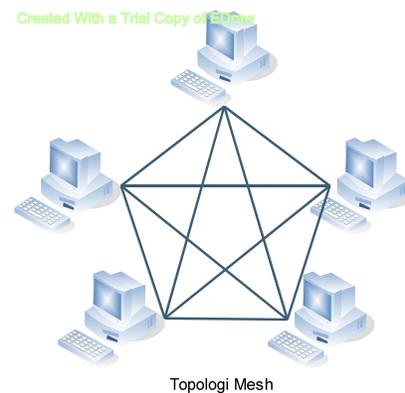
*Hierarchical topology* mirip dengan *extended star topology*. Perbedaan utamanya adalah topologi ini tidak menggunakan satu sentral *node*, melainkan menggunakan *trunk node* dengan masing-masing cabang ke *node* lainnya.



Gambar 2.14 *Hierarchical Topology*

### 6. *Mesh Topology*

Pada topologi ini, masing-masing *host* saling terhubung ke setiap *host* dalam *network*.



Gambar 2.15 *Mesh Topology*

### 2.3.2 *Logical Topology*

*Logical topology* merupakan topologi yang menggambarkan bagaimana *host* mengakses suatu media ([http://en.wikipedia.org/wiki/Network\\_topology#Logical\\_topology](http://en.wikipedia.org/wiki/Network_topology#Logical_topology)).

*Logical topology* terdiri dari 2 jenis, yaitu:

1. *Broadcasting*

*Broadcasting* artinya data yang dikirim oleh masing-masing *host* dalam jaringan tersebar ke setiap *host* dalam jaringan tersebut. *Broadcasting* menggambarkan secara logika pengiriman data yang terjadi dalam *physical topology bus*.

2. *Token passing*

*Token passing* mengontrol pengiriman data dengan mengirimkan *token* yang berupa sinyal elektronik ke setiap *host* secara bergiliran dan sekuensial. *Host* yang bisa mengirim data hanya *host* yang sedang menerima *token*. Jika saat *token* diterima oleh *host*, tapi *host* tersebut tidak mengirim data, maka *token* langsung dilempar ke *host* berikutnya (<http://www.cisco.com>).

## 2.4 Manajemen Jaringan

Menurut Subramanian (2000, p135), manajemen jaringan merupakan pengaturan, pengurusan atau pengelolaan elemen pada jaringan agar jaringan tetap dapat digunakan untuk ke depannya. Manajemen jaringan dapat dikategorikan menjadi lima macam, yaitu:

1. *Configuration Management* (Manajemen Konfigurasi)

Meliputi pengaturan alamat (*address*) dan perubahan konfigurasi dari jaringan dan komponen-komponen di dalamnya.

2. *Fault Management* (Manajemen Kesalahan)

Meliputi deteksi masalah, pengisolasian kesalahan, dan perbaikan sehingga jaringan dapat kembali ke operasi normal.

3. *Performance Management* (Manajemen Performa)

Meliputi pengaturan tingkah laku performa dari jaringan yang sedang berjalan dimana performa jaringan ini ditampilkan dalam statistik jaringan seperti *traffic volume*, *network availability*, dan *network delay*.

4. *Security Management* (Manajemen Keamanan)

Mengatur keamanan fisik jaringan, akses ke sumber daya jaringan, dan keamanan komunikasi yang terjadi di dalam jaringan.

5. *Accounting Management* (Manajemen Akunting)

Meliputi manajemen keuangan yang dikeluarkan dalam pengelolaan jaringan.

## 2.5 *Internet Service Provider (ISP)*

ISP merupakan sebuah organisasi atau perusahaan yang menyediakan akses ke internet ([http://en.wikipedia.org/wiki/Internet\\_service\\_provider](http://en.wikipedia.org/wiki/Internet_service_provider)). ISP berskala kecil menyediakan jasa melalui *modem* dan ISDN sedangkan ISP yang lebih besar menawarkan pemasangan *private line*. Pada umumnya, pelanggan akan diberikan tagihan dengan biaya tetap per bulannya, namun mungkin saja terdapat biaya-biaya tambahan lainnya.

Selain melayani pelanggan individual, ISP juga melayani perusahaan-perusahaan besar dalam menyediakan koneksi langsung dari jaringan komputer di perusahaan tersebut ke internet. ISP sendiri terhubung satu dengan yang

lainnya melalui organisasi yang disebut dengan *Network Access Points* (NAP). Dalam hubungannya dengan menyediakan jasa internet, ISP juga disebut dengan *Internet Access Providers* (IAP). Contoh ISP yang ada di Indonesia, misalnya: IndosatM2, Centrin, CBN, FastNet, Speedy, dan lain-lain.

### 2.5.1 Layanan ISP

Contoh dari layanan-layanan yang diberikan oleh ISP misalnya:

1. Alamat *email*, *web hosting*, berita-berita terbaru melalui internet, akses ke internet, dan *File Transfer Protocol* (FTP) yang dibutuhkan untuk *upload* data.
2. *Value-added services* seperti VOIP, *web hosting*, *webmail*, IRC, dan *call center*. Bahkan ada juga ISP dengan layanan *online shopping*.
3. *Backup*, *update*, dan *upgrade* layanan-layanan yang telah diberikan dan *monitoring* jaringan untuk memastikan tidak ada gangguan pada koneksi.
4. Ada juga ISP yang menyediakan layanan penjualan dan juga perbaikan *hardware*, walaupun untuk itu *user* akan dikenakan biaya.

## 2.6 *Internet Protocol* (IP)

IP adalah deretan angka biner antar 32-bit sampai 128-bit yang dipakai sebagai alamat identifikasi untuk tiap komputer host dalam jaringan internet ([http://www.g-excess.com/id/study/pengertian\\_dari\\_ip.html](http://www.g-excess.com/id/study/pengertian_dari_ip.html)). IP mengatur

bagaimana suatu data dikirim dari suatu komputer ke komputer lain dalam suatu jaringan. Setiap komputer yang ada dalam jaringan internet setidaknya memiliki satu alamat IP yang bersifat unik dan berbeda dengan komputer lainnya.

IP mempunyai tiga fungsi utama dalam jaringan TCP/IP internet:

1. IP merupakan bagian dasar dalam pengiriman data di internet, mendefinisikan format data yang tepat untuk dikirimkan ke internet.
2. IP mendefinisikan fungsi *routing* dan mendefinisikan pengalamatan dari komputer.
3. IP mendefinisikan bagaimana suatu paket data harus diproses.

IP dapat dibagi menjadi dua macam, yaitu:

1. Pemakaian di internet

Berdasarkan kategori ini, IP dapat digolongkan menjadi dua macam:

- *Private IP address*

*Private IP address* adalah alamat IP yang digunakan oleh sebuah komunitas, baik rumah maupun perusahaan yang tidak tersambung langsung ke internet

([http://s3kitarkita.blogspot.com/2008\\_11\\_20\\_archive.html](http://s3kitarkita.blogspot.com/2008_11_20_archive.html)).

Alamat IP ini tidak bisa berkomunikasi langsung dengan komputer lain pada jaringan internet, sehingga untuk dapat berkomunikasi dibutuhkan perantara, yaitu ISP yang menyediakan jasa layanan internet. *Range private IP address* bergantung dari jenis kelas IP:

- ❖ *Class A* antara 10.0.0.0 sampai 10.255.255.255
- ❖ *Class B* antara 172.16.0.0 sampai 172.31.255.255

❖ *Class C* antara 192.168.0.0 sampai 192.168.255.255

- *Public IP address*

*Public IP address* adalah alamat IP yang digunakan untuk berkomunikasi antar komputer yang tersambung secara langsung dalam jaringan internet

([http://s3kitarkita.blogspot.com/2008\\_11\\_20\\_archive.html](http://s3kitarkita.blogspot.com/2008_11_20_archive.html)).

Orang lain dapat mengetahui dan mengakses komputer yang menggunakan *public IP address*. *Range* alamat yang dimiliki oleh *public IP address* adalah semua alamat IP selain yang berada dalam *range private IP address* dan *IP loopback* (127.0.0.1). *IP loopback* merupakan *IP default* dari suatu komputer.

## 2. Alamat yang mampu ditampung

Format penulisan *IP Address* dibagi menjadi dua, yaitu

(<http://ict.pontianak.go.id/ebook/suse/suse3.pdf>) :

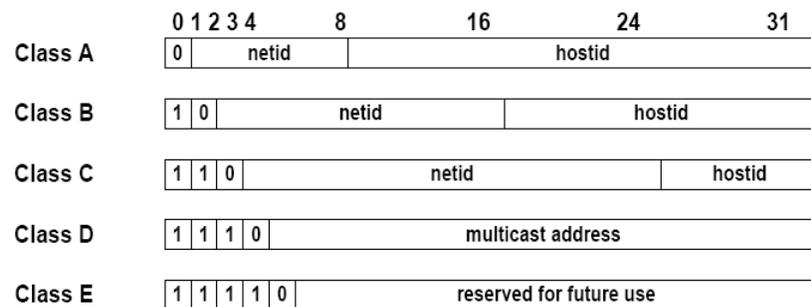
### a. Bentuk Biner

*IP address* merupakan suatu bilangan biner 32 bit yang dipisahkan oleh tanda pemisah berupa tanda titik setiap 8 bit (yang disebut sebagai oktet). Setiap bit diisi dengan angka biner, yaitu 0 dan 1.

### b. Bentuk Desimal Bertitik

Notasi *IP address* dengan bilangan biner tidak mudah untuk dimengerti begitu saja. Untuk itu, *IP Address* lebih sering ditulis sebagai 4 bilangan desimal yang masing-masing dipisahkan oleh titik. Satu bilangan desimal mewakili satu oktet. Dalam sebuah *IP*

*address* terdapat dua buah identitas atau ID, yaitu *network-ID* dan *host-ID*. *Network-ID* adalah bagian dari *IP address* yang berfungsi untuk menunjukkan jaringan tempat komputer berada, sedangkan *host-ID* adalah bagian dari *IP address* yang berfungsi untuk menunjukkan komputernya atau *host*, *server*, *router*, dan semua *host* TCP/IP yang terdapat pada sebuah jaringan.



Gambar 2.16 Pembagian IP Address Class

(Sumber: [http://www.e-](http://www.e-dukasi.net/pengpop/datafitur/peng_populer/PP_286/images/7_2e.j)

[dukasi.net/pengpop/datafitur/peng\\_populer/PP\\_286/images/7\\_2e.j](http://www.e-dukasi.net/pengpop/datafitur/peng_populer/PP_286/images/7_2e.j)

[pg](http://www.e-dukasi.net/pengpop/datafitur/peng_populer/PP_286/images/7_2e.j))

Untuk memudahkan dalam pengorganisasian dan pendistribusiannya, *IP Address* diklasifikasikan ke dalam beberapa *class*:

#### 1. IP Address Class A

Format : Onnnnnnn.hhhhhhhh.

hhhhhhh.hhhhhh

Bit Awal : 0



*Range IP* : 192.0.0.0 – 223.255.255.255

Jumlah *Host Class C* : 254

#### 4. IP Address Class D

Format : 1110mmmm.mmmmmmmm.  
mmmmmmmm.mmmmmmmm

Bit Awal : 1110

Oktet Pertama : 224 - 239

Bit *Multicast* : 28 bit

Byte Inisial : 224 - 239 bit

*Range IP* : 224.0.0.0 – 239.255.255.255

Deskripsi : Kelas D adalah ruang alamat  
*multicast* (RFC 1112)

#### 5. IP Address Class E

Format : 11110rrr rrrrrrrr rrrrrrrr rrrrrrrr

Bit Awal : 11110

Oktet Pertama : 240 - 254

Bit Cadangan : 28 bit

Byte Inisial : 248 –255

*Range IP* : 240.0.0.0 – 254.255.255.255

Deskripsi : Kelas E adalah ruang alamat yang  
dicadangkan untuk keperluan  
eksperimental

Setelah melakukan pengalamatan IP, maka diperlukan suatu *subnet mask*.

Menurut Setiawan (2003, p147), *subnet* merupakan suatu bagian dari jaringan

yang digunakan untuk memecah *network-ID* menjadi beberapa jaringan yang lebih kecil. Tujuannya adalah agar beban kerja dari jaringan tidak terlalu besar, walaupun dalam jaringan mampu menjangkau lebih dari 100 *host* atau komputer namun beban yang besar tersebut jarang sekali digunakan. Biasanya perusahaan akan memecah jaringan menjadi beberapa jaringan kecil yang disebut *subnetting* dan dalam jaringan dinamakan dengan *subnet*.

*Subnet mask* digunakan untuk membedakan antara *host-ID* dan *network-ID* dan untuk menunjukkan letak suatu *host* apakah berada pada jaringan LAN atau WAN. *Subnet* ini terdiri dari angka biner sebesar 32 bit. Pada *subnet mask*, seluruh bit yang berhubungan dengan *network-ID* diatur dengan angka 1 dan yang berhubungan dengan *host-ID* diatur dengan angka 0.

*Subnet mask* diperlukan untuk melakukan operasi pengiriman IP dengan cara melakukan operasi *AND* antara *subnet mask* dengan *IP address* asal ke tempat *IP address* tujuan, serta membandingkan hasilnya untuk mengetahui arah tujuan *IP address* tersebut. Apabila hasil kedua operasi tersebut sama, maka *host* tujuan terletak dalam jaringan LAN. Sedangkan jika hasilnya berbeda, maka *host* tujuan berada dalam jaringan WAN.

Penggunaan *subnet mask* sangat ditentukan oleh banyaknya komputer atau *host* pada sebuah jaringan. Menurut Setiawan (2003, p198), untuk membedakan banyaknya *host* yang digunakan, *subnet mask* dibagi menjadi beberapa *class*, yaitu:

1. *Class A*

*Subnet Mask* : 11111111.00000000.00000000.00000000 = 255.0.0.0

Panjang *Network-ID* : 8

Panjang *Host-ID* : 24

Daerah *IP Address* : 1.0.0.0 – 127.255.255.255

Banyaknya *host* yang dapat ditampung : ± 16 juta

## 2. *Class B*

*Subnet Mask* : 11111111.11111111.00000000.00000000 = 255.255.0.0

Panjang *Network-ID* : 16

Panjang *Host-ID* : 16

Daerah *IP Address* : 128.0.0.0 – 191.255.255.255

Banyaknya *host* yang dapat ditampung : ± 65.000

## 3. *Class C*

*Subnet Mask* : 11111111.11111111.11111111.00000000 = 255.255.255.0

Panjang *Network-ID* : 24

Panjang *Host-ID* : 8

Daerah *IP Address* : 192.0.0.0 – 223.255.255.255

Banyaknya *host* yang dapat ditampung : ± 256

Dengan penggunaan *subnetting* ini, maka penggunaan *host-ID* menjadi lebih sedikit dan beban sebuah *server* menjadi lebih ringan. Semakin panjang sebuah *subnet*, maka jumlah *subnet* yang dapat dibentuk menjadi semakin banyak. Namun jumlah *host* dalam setiap *subnet* menjadi semakin sedikit.

## 2.7 *Network Address Translation (NAT)*

*Network Address Translation* atau yang lebih biasa disebut dengan NAT adalah suatu metode untuk menghubungkan lebih dari satu komputer ke jaringan internet dengan menggunakan satu alamat IP

([http://id.wikipedia.org/wiki/Network\\_Address\\_Translation](http://id.wikipedia.org/wiki/Network_Address_Translation)). Teknologi NAT memungkinkan alamat IP *private* terhubung ke jaringan *public* seperti internet. Sebuah *router* NAT ditempatkan antara jaringan lokal (*inside network*) dan jaringan *public* (*outside network*) serta mentranslasikan alamat lokal atau internal menjadi alamat IP global yang unik sebelum mengirimkan paket ke jaringan luar seperti internet. Dengan NAT, jaringan lokal tidak akan terlihat oleh dunia luar. IP lokal yang cukup banyak dapat dilewatkan ke internet hanya dengan melalui translasi ke satu IP *public*.

NAT sebaiknya digunakan pada kondisi jika membutuhkan koneksi internet dan *host* (komputer) tidak mempunyai alamat IP global. Selain itu, NAT juga digunakan pada kondisi jika berganti ke ISP baru dan diharuskan menggunakan alamat IP dari ISP baru tersebut untuk jaringan. Ada dua macam tipe NAT, yaitu *static* dan *dynamic* yang keduanya dapat digunakan secara terpisah maupun bersamaan. Penjelasannya adalah sebagai berikut:

1. *Static*

Translasi *static* terjadi ketika sebuah alamat lokal (*inside*) dipetakan ke sebuah alamat global atau internet (*outside*). Alamat lokal dan global dipetakan satu lawan satu secara *static*.

2. *Dynamic*

Translasi *dynamic* terjadi ketika *router* NAT diset untuk memahami alamat lokal yang harus ditranslasikan dan kelompok alamat global yang akan digunakan untuk terhubung ke internet. Proses NAT *dynamic* ini dapat memetakan beberapa kelompok alamat lokal ke beberapa kelompok alamat global.

Keuntungan menggunakan NAT adalah:

- Menghemat alamat IP legal (yang ditetapkan oleh *Network Interface Card* (NIC) atau *service provider*).
- Mengurangi terjadinya penggandaan alamat jaringan IP.
- Meningkatkan fleksibilitas untuk koneksi ke internet.
- Menghindarkan proses pengalamatan kembali pada saat jaringan berubah.

Kerugian menggunakan NAT adalah:

- Translasi menimbulkan *delay switching*.
- Menghilangkan kemampuan *trace end to end* IP.
- Aplikasi tertentu tidak dapat berjalan jika menggunakan NAT.

## 2.8 *Bandwidth*

Secara umum, *bandwidth* dapat diandaikan sebagai sebuah pipa air yang memiliki diameter tertentu. Semakin besar *bandwidth*, semakin besar pula diameter pipa tersebut sehingga kapasitas volume air (dalam hal ini air merupakan data dalam arti sebenarnya) dapat meningkat. Semakin besar *bandwidth* suatu media, semakin tinggi kecepatan data yang dapat dilaluinya.

Pengertian *bandwidth* menurut para ahli adalah sebagai berikut:

1. Menurut Norton dan Kearns (1999, p29), *bandwidth* ialah lebar komunikasi di antara saluran yang diukur dalam Hz.
2. Menurut Tanenbaum (2003, p88), *bandwidth* ialah jarak dari frekuensi yang ditransmisikan tanpa menyebabkan sinyal menjadi lemah.

*Bandwidth* dapat dikategorikan menjadi dua macam:

1. *Digital bandwidth*

*Digital bandwidth* merupakan jumlah atau volume data yang dapat dikirimkan melalui sebuah saluran komunikasi dalam satuan *bits per second* tanpa distorsi.

2. *Analog bandwidth*

*Analog bandwidth* merupakan perbedaan antara frekuensi terendah dengan frekuensi tertinggi dalam sebuah rentang frekuensi yang diukur dalam satuan *Hertz* (Hz) atau siklus per detik, yang menentukan berapa banyak informasi yang bisa ditransmisikan dalam satu saat.

Alokasi atau reservasi *bandwidth* adalah sebuah proses untuk menentukan besar *bandwidth* kepada pemakai dan aplikasi dalam sebuah jaringan. Termasuk di dalamnya menentukan prioritas terhadap berbagai jenis aliran data berdasarkan seberapa penting dan sensitif penundaan terhadap aliran data tersebut. Hal ini memungkinkan penggunaan *bandwidth* yang tersedia secara efisien dan apabila sewaktu-waktu jaringan menjadi lambat, aliran data yang memiliki prioritas yang lebih rendah dapat dihentikan, sehingga aplikasi yang penting dapat tetap berjalan dengan lancar.

*Bandwidth* merupakan salah satu faktor penting dalam jaringan. Beberapa hal yang menyebabkan *bandwidth* menjadi bagian penting yang harus diperhatikan adalah:

1. *Bandwidth* berdampak pada kinerja sebuah jaringan

Besarnya saluran atau *bandwidth* akan berdampak pada kecepatan transmisi. Data dalam jumlah besar akan menempuh saluran yang

memiliki *bandwidth* kecil lebih lama dibandingkan melewati saluran yang memiliki *bandwidth* yang besar. Kecepatan transmisi tersebut sangat dibutuhkan untuk aplikasi komputer yang memerlukan jaringan terutama aplikasi *real-time*, seperti *video conferencing*.

2. *Bandwidth* memiliki keterbatasan

Setiap medium yang digunakan untuk mentransmisikan data memiliki batas maksimal *bandwidth* yang dapat dicapai.

3. *Bandwidth* tidak didapatkan dengan gratis

Penggunaan *bandwidth* untuk LAN bergantung pada tipe alat atau medium yang digunakan. Umumnya semakin tinggi *bandwidth* yang ditawarkan oleh sebuah alat atau medium, semakin tinggi pula nilai jualnya. Sedangkan penggunaan *bandwidth* untuk WAN bergantung dari kapasitas yang ditawarkan dari pihak ISP. Perusahaan harus membeli *bandwidth* dari ISP dan semakin tinggi *bandwidth* yang diinginkan, semakin tinggi pula harganya.

4. Kebutuhan akan *bandwidth* akan selalu naik

Setiap sebuah teknologi jaringan baru dikembangkan dan infrastruktur jaringan yang ada diperbaharui, aplikasi yang akan digunakan umumnya juga akan mengalami peningkatan dalam hal konsumsi *bandwidth*.

Satuan dasar dari *bandwidth* adalah *bits per second* (bps). Walaupun satuan dasar yang dipakai bps, unit satuan yang lebih besar lebih umum dipakai. *Network bandwidth* biasanya dihitung dalam satuan *thousands bits per second* (Kbps), *millions bits per second* (Mbps), *billions bits per second* (Gbps), dan *trillions bits per second* (Tbps). Satuan ini umum digunakan dalam pemakaian

sehari-hari, terutama karena semakin meningkatnya kebutuhan *bandwidth* dan perkembangan teknologi informasi.

Besarnya *bandwidth* bervariasi tergantung dari tipe medium yang digunakan serta teknologi LAN atau WAN yang digunakan. Fisik dan medium yang digunakan juga turut mempengaruhi besarnya *bandwidth*. Sinyal data dapat melalui kabel *twisted-pair*, kabel koaksial, kabel serat optik, dan udara. Perbedaan dari bagaimana sinyal tersebut berjalan secara fisik mengakibatkan batasan mendasar terhadap besarnya kapasitas ditentukan oleh kombinasi dari medium fisik dan teknologi yang dipilih untuk bisa mendeteksi dan mengirimkan sinyal data dalam sebuah jaringan.

## 2.9 *Load balancing*

Dalam jaringan komputer, *load balancing* adalah sebuah konsep yang digunakan untuk menyeimbangkan beban atau muatan pada sebuah perusahaan agar seluruh jaringan dapat dimanfaatkan secara maksimal dan optimal ([http://www.pc24.co.id/article/category40\\_1.htm](http://www.pc24.co.id/article/category40_1.htm)). Kondisi menggunakan *load balancing* terjadi apabila ISP menggunakan lebih dari satu *gateway* untuk terhubung ke internet.

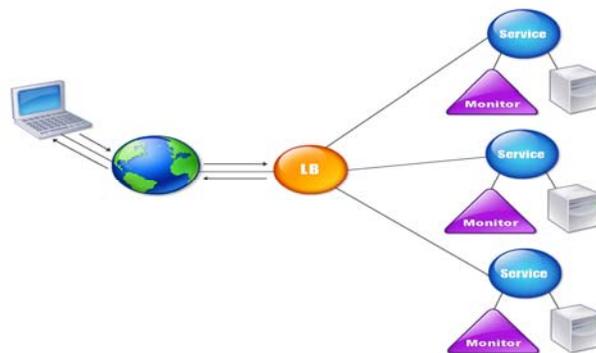
Dengan menggunakan komponen-komponen *load balancing* seperti *single component*, akan meningkatkan reliabilitas karena akan mengurangi beban pada sistem ataupun jaringan. Layanan *load balancing* biasanya disediakan oleh program yang dijalankan atau perangkat *hardware* seperti *multilayer switch*.

*Load balancing* sangat penting jika jumlah permintaan proses ke *server* sulit diramalkan. Situs web yang sibuk atau banyak dikunjungi biasanya

menggunakan dua atau lebih web *server*, sehingga jika salah satu *server* kelebihan beban, permintaan akan dialihkan ke *server* lain yang masih mempunyai kapasitas untuk memproses permintaan tersebut. Untuk lebih memaksimalkan *load balancing*, maka diperlukan suatu pemisah *IP address list* antara IIX dan Internasional. *IP address list* yang memuat IIX dinamakan NICE (*National Inter Connection Exchange*) yang dapat di-*download* di internet. NICE di-*update* setiap hari karena *IP address* tersebut setiap hari akan berubah.

Kelebihan *load balancing*:

1. Mengatur agar tidak terjadi penumpukan di salah satu ISP, sehingga dapat dipakai semua jalur atau *gateway* yang disediakan.
2. Mengatur agar dapat memisahkan antara jaringan internasional dan *Indonesia Internet Exchange* (IIX) supaya tidak terjadi tarik-menarik *bandwidth* dalam pemakaian internet.
3. Memperkecil *deadlock* yang sering terjadi pada perusahaan.



Gambar 2.17 *Load Balancing*

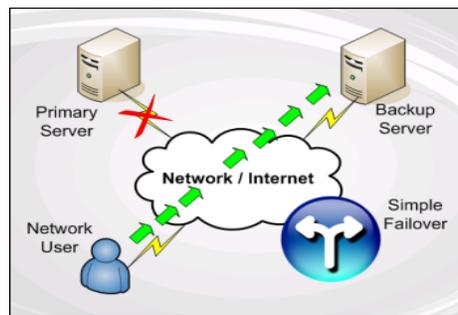
(Sumber:

<http://www.ciscoimm.com/mikrotik/lb/loadbalancing.jpg>)

## 2.10 Failover

*Failover* ialah kemampuan untuk memindahkan komputer *server* secara otomatis apabila pada sistem atau jaringan yang sedang dipakai terjadi kegagalan ataupun penghentian yang abnormal dari *server* yang sedang aktif tersebut.

*Failover* biasanya terjadi tanpa campur tangan manusia dan umumnya tanpa pemberitahuan, contohnya seperti *switchover*. Sistem *designer* biasanya menyediakan kemampuan *failover* pada *server*, sistem ataupun jaringan yang harus tersedia terus-menerus dan membutuhkan tingkat kemampuan yang tinggi.



Gambar 2.18 *Failover*

(Sumber: <http://www.simplefailover.com/>)

## 2.11 Alert

*Alert* atau sinyal adalah suatu bunyi untuk mendefinisikan suatu komunikasi mesin ke orang yang penting dan biasanya bersifat mendesak. *Alert* yang dimaksud dalam skripsi ini adalah suatu bunyi penanda bagi *administrator* jaringan untuk mengetahui kesalahan yang ada dalam jaringan seperti peringatan yang penting dan pemberitahuan yang mendesak. Misalnya pada saat baterai *notebook* tinggal sedikit, atau adanya peringatan *error*, dan lain sebagainya.

*Alert* atau sinyal secara khas dikirimkan melalui suatu sistem pemberitahuan dan aplikasi paling umum dari jasa komunikasi mesin ke *user*. Konfigurasi *alert* merupakan suatu pengaturan bunyi untuk mengetahui koneksi jaringan yang terputus dengan menggunakan *script* tertentu.

Untuk menghasilkan *alert* dalam MikroTik, maka diperlukan suatu frekuensi musik. Frekuensi ini dituangkan dalam bentuk *script-script* yang akan menghasilkan bunyi *alert* sesuai yang diinginkan.

Tabel 2.1 *Musical Note Frequency*

OCT #	C	C#/Db	D	D#/Eb	E	F
0	16.3515978313	17.3239144361	18.3540479948	19.4454364826	20.6017223071	21.8267644646
1	32.7031956626	34.6478288721	36.7080959897	38.8908729653	41.2034446141	43.6535289291
2	65.4063913251	69.2956577442	73.4161919794	77.7817459305	82.4068892282	87.3070578583
3	130.8127826503	138.5913154884	146.8323839587	155.5634918610	164.8137784564	174.6141157165
4	261.6255653006	277.1826309769	293.6647679174	311.1269837221	329.6275569129	349.2282314330
5	523.2511306012	554.3652619537	587.3295358348	622.2539674442	659.2551138257	698.4564628660
6	1046.5022612024	1108.7305239075	1174.6590716696	1244.5079348883	1318.5102276515	1396.9129257320
7	2093.0045224048	2217.4610478150	2349.3181433393	2489.0158697766	2637.0204553030	2793.8258514640

OCT #	F#/Gb	G	G#/Ab	A	A#/Bb	B
0	23.1246514195	24.4997147489	25.9565435987	27.5000000000	29.1352350949	30.8677063285
1	46.2493028390	48.9994294977	51.9130871975	55.0000000000	58.2704701898	61.7354126570
2	92.4986056779	97.9988589954	103.8261743950	110.0000000000	116.5409403795	123.4708253140
3	184.9972113558	195.9977179909	207.6523487900	220.0000000000	233.0818807590	246.9416506281
4	369.9944227116	391.9954359817	415.3046975799	440.0000000000	466.1637615181	493.8833012561
5	739.9888454233	783.9908719635	830.6093951599	880.0000000000	932.3275230362	987.7666025122
6	1479.9776908465	1567.9817439270	1661.2187903198	1760.0000000000	1864.6550460724	1975.5332050245
7	2959.9553816931	3135.9634878540	3322.4375806396	3520.0000000000	3729.3100921447	3951.0664100490

Frekuensi-frekuensi nada di atas diperoleh dari <http://www.jordanautomations.com/notetofrequency.html>. *Note* pada musik dapat diubah menjadi suatu frekuensi sehingga disebut sebagai *musical note frequency*. *Musical note frequency* ini biasanya digunakan untuk keperluan lain

pada sebuah peralatan dimana peralatan tersebut tidak bisa menggunakan lagu atau musik yang telah jadi.

*Musical note frequency* dihasilkan dari tangga nada (*chord*) suatu alat musik seperti piano, gitar, dan lain-lain menjadi frekuensi tertentu sesuai tangga nadanya. Setiap *musical note* memiliki frekuensi yang diukur dalam satuan *Hertz* (Hz) atau jumlah getaran tiap detik.

Untuk memainkan suatu lagu yang diformat dalam bentuk frekuensi, maka harus mengetahui *chord* dari lagu yang dimainkan dan ketukan *chord* lagu tersebut agar bunyi sesuai dengan lagu yang dimainkan.

## **2.12 MikroTik**

Menurut Herlambang dan Catur L (2008, p19), MikroTik RouterOS™ adalah sistem operasi yang dirancang khusus untuk *network router*. Dengan menggunakan sistem operasi ini, dapat dibuat *router* dari sebuah komputer. Untuk negara berkembang, solusi MikroTik sangat membantu ISP atau perusahaan-perusahaan kecil yang ingin bergabung dengan internet. Walaupun sudah banyak tersedia perangkat *router* mini sejenis NAT, MikroTik merupakan solusi terbaik dalam beberapa kondisi penggunaan komputer dan perangkat lunak.

### **2.12.1 Jenis-Jenis MikroTik**

Jenis-jenis MikroTik yang dikenal sekarang ini adalah:

### 1. MikroTik RouterOS™

Merupakan versi MikroTik dalam bentuk perangkat lunak yang dapat di-*install* pada komputer melalui CD. Versi *trial* MikroTik ini dapat di-*download* pada *website* resmi MikroTik yaitu [www.mikrotik.com](http://www.mikrotik.com). Untuk dapat menggunakannya secara *full time*, harus membeli *license key* dengan catatan satu *license key* hanya untuk satu *harddisk*. MikroTik versi ini biasanya disebut juga dengan MikroTik non DoM (*Disk on Modul*).

### 2. *Built in hardware* MikroTik

Merupakan MikroTik dalam bentuk perangkat keras yang khusus dikemas dalam *board router* yang di dalamnya sudah ter-*install* MikroTik RouterOS™. Untuk versi ini, lisensi sudah termasuk dalam harga *router board* MikroTik. MikroTik ini dilengkapi dengan *Disk on Module* (DoM). Setiap pembelian OS MikroTik dalam bentuk DoM, maka akan mendapatkan sebuah DoM yang sudah dipasangkan OS MikroTik di dalamnya. DoM ini nantinya dipasangkan ke dalam sebuah PC dengan konektor *harddisk* PC. *Harddisk* yang akan dipasangkan MikroTik dapat berupa IDE *harddisk* atau SATA *harddisk*. Pada MikroTik versi dua tidak bisa menggunakan SATA *harddisk* karena tidak terdapat *integrated SATA driver*. Untuk MikroTik versi 3 ke atas dapat menggunakan *harddisk* IDE ataupun SATA karena MikroTik tersebut sudah mendukung SATA dan di dalam kernel mikrotiknya terdapat *integrated SATA driver*. Untuk kebutuhan

perangkat keras yang lebih kuat dari *router board*, misalnya untuk kebutuhan *routing*, maka pilihan membeli OS dalam bentuk DoM menjadi pilihan.



Gambar 2.19 MikroTik DoM

(Sumber: <http://mikrotik.co.id/produk.php?kategori=10>)

### 2.12.2 Fungsi-Fungsi MikroTik

MikroTik mempunyai banyak kegunaan, di antaranya adalah sebagai berikut:

#### 1. MikroTik sebagai *gateway*

MikroTik sebagai *gateway* merupakan salah satu bentuk implementasi yang paling banyak dipakai pada warnet. Tujuannya agar *client* dengan IP yang beragam dapat berkoneksi dengan internet. MikroTik minimal memakai dua LAN *card* jika difungsikan sebagai *router*. Satu LAN *card* berfungsi sebagai antarmuka publik yang terhubung dengan internet, sedangkan

LAN *card* yang satunya berfungsi sebagai antarmuka lokal yang berperan sebagai *gateway* jaringan lokal.

## 2. MikroTik sebagai *proxy server*

Peran *proxy server* dibutuhkan untuk mempercepat akses *website*. Suatu halaman *website* yang pernah dikunjungi oleh *user* akan disimpan dalam *proxy server*. Jadi ketika ada *client* yang *request* suatu halaman *website*, *browser* tidak perlu untuk *request* langsung ke *web server*.

## 3. MikroTik sebagai DHCP *server*

*Dynamic Host Configuration Protocol* (DHCP) adalah suatu *protocol* yang dapat mendistribusikan *IP address*, *default gateway*, *DNS server*, dan *WINS server* (untuk *client Windows*) kepada *client* dalam sebuah jaringan secara otomatis. Dengan menggunakan MikroTik, kita bisa menambahkan sendiri batasan *IP* yang akan didistribusikan, tergantung jumlah komputer yang tersedia.

## 4. MikroTik sebagai *limit bandwidth*

Dalam fungsi ini, *bandwidth* tiap-tiap *client* dibatasi penggunaannya. *Bandwidth* yang tidak dibagi secara merata menyebabkan koneksi di beberapa *client* lain terasa lambat. Hal ini disebabkan adanya beberapa karakter pengguna yang tidak sama antara satu dengan yang lainnya.

#### 5. MikroTik sebagai *firewall*

Selain sebagai *gateway*, MikroTik juga dipadukan dengan kemampuan *firewall* untuk mencegah hal-hal yang mengganggu dari pihak lain, mengingat banyaknya aplikasi yang dijalankan oleh pengguna jaringan. Paket dengan ukuran kecil memang tidak mengganggu koneksi jaringan. Namun, jika paket yang kecil tersebut dalam jumlah banyak, hal ini bisa menurunkan kinerja jaringan. Oleh karena itu, *firewall* sangat penting digunakan untuk menghindari insiden jaringan yang bersifat negatif.

#### 6. MikroTik sebagai *bridging*

MikroTik sebagai *bridging* mempunyai banyak tujuan, antara lain, dijadikan *limit bandwidth*, *De-Militarize Zone (DMZ)* atau dijadikan *monitoring traffic*. Implementasi *bridging* lebih mudah karena tidak memerlukan konfigurasi *routing*, memungkinkan tidak memakai *IP address* atau IP lokal karena paket hanya dilewatkan saja melewati dua *LAN card router*.

### 2.12.3 Perbandingan MikroTik Versi Dua dan Tiga

Perbedaan antara MikroTik versi dua dengan versi tiga tidak terlalu jauh perbandingannya. Dengan peningkatan versi yang lebih tinggi, maka ada fitur-fitur yang baru dan kegunaan yang menjadi lebih bagus. Fitur-fitur tersebut seperti:

1. Dalam *console* diperbolehkan *command* seperti '*monitor*' dan '*tool fetch*' pada *scheduler scripts*.

2. Pengetikan *command line* dalam *console* akan berwarna-warni, sehingga dapat mengetahui apakah *command* yang diketik salah atau benar. Jika *command* yang diketik salah, maka tulisan akan berwarna merah.

Kelemahan MikroTik versi dua dengan versi tiga adalah MikroTik versi dua tidak dapat melakukan proses *update address-list* secara otomatis yang berguna untuk melakukan pemisah antara *bandwidth* internasional dengan IIX, sehingga harus dilakukan secara manual. Selain itu, penulisan *command line* dalam *console* tidak berwarna-warni sehingga tidak dapat mengetahui kesalahan dalam pengetikan. Oleh karena itu, dengan MikroTik versi tiga ada tambahan fitur-fitur yang berguna untuk pemakaian MikroTik dan memperbaiki *bug-bug* yang ada di versi dua.

### 2.13 *Bandwidth Monitor*

*Bandwidth monitor* merupakan perangkat lunak yang diinstal pada komputer. Perangkat lunak ini kompatibel dengan semua jenis koneksi jaringan. *Bandwidth monitor* digunakan untuk melihat pemakaian *bandwidth* dan semua koneksi jaringan pada komputer, seperti koneksi jaringan LAN, koneksi jaringan internet, dan koneksi VPN. Selain itu, *bandwidth monitor* dapat digunakan untuk menampilkan *real-time* kecepatan *download* dan *upload* dalam bentuk angka dan grafik, *log* penggunaan *bandwidth*, dan menyediakan ringkasan data laporan penggunaan *bandwidth*.

Cara *bandwidth monitor* bekerja:

- a. *Bandwidth monitor* mendeteksi koneksi jaringan pada komputer.
- b. *Bandwidth monitor* akan menampilkan *real-time* kecepatan *download* dan *upload* serta *log* penggunaan *bandwidth*.
- c. *Bandwidth monitor* akan memberitahu pengguna untuk menghindari terjadinya *over bandwidth*.

Manfaat *bandwidth monitor*:

- a. Me-monitor *bandwidth* jaringan

Pengguna dapat melihat jalannya *traffic* pada koneksi jaringan (LAN dan WAN) secara *real-time* dengan menggunakan *bandwidth monitor*.

- b. Melihat kecepatan *download* dan *upload* secara *real-time*

Pengguna dapat melihat kecepatan *download* dan *upload* pada jaringan atau koneksi internet secara *real-time*. *Bandwidth monitor* memiliki dua satuan, yaitu kB/sec (*kilobytes per seconds*) dan kbps (*kilobits per second*).

- c. Me-monitor penggunaan *bandwidth* dari *server*

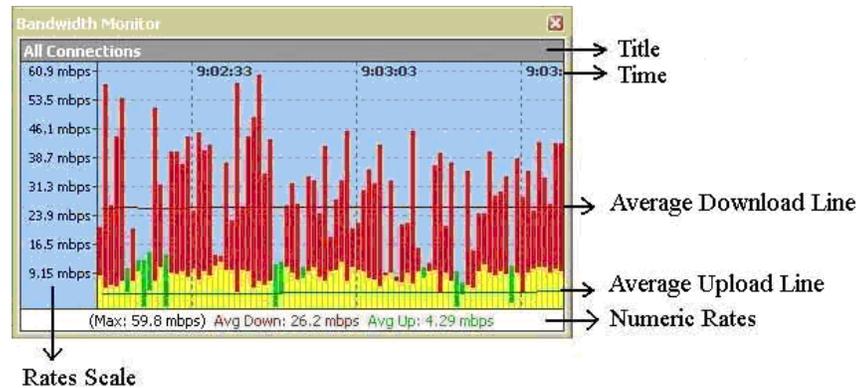
*Bandwidth monitor* me-monitor penggunaan *bandwidth* dari *server* tanpa perlu *login*. Selain itu, *bandwidth monitor* me-monitor penggunaan *bandwidth*, *record rate*.

- d. Menguji kecepatan *download* dan *upload* koneksi jaringan dalam jangka waktu yang ditetapkan pengguna

*Stopwatch* dapat membantu pengguna untuk menguji kecepatan *download* dan *upload* dari koneksi jaringan atau internet.

- e. Mencatat *traffic* koneksi jaringan secara spesifik

Pengguna dapat merekam kecepatan *download* dan *upload* dari satu atau lebih koneksi jaringan dan mengirim dalam bentuk teks atau file microsoft excel.



Gambar 2.20 *Bandwidth Monitor*

Keterangan:

Diagram batang yang berwarna merah menunjukkan *download rate*.

Diagram batang yang berwarna hijau menunjukkan *upload rate*.

Diagram batang yang berwarna kuning menunjukkan *download* dan *upload rate*.

## 2.14 *Multi Router Traffic Grapher (MRTG)*

Secara singkat (<http://oss.oetiker.ch/mrtg/doc/mrtg.en.html>), *Multi Router Traffic Grapher (MRTG)* adalah *tool* yang berguna untuk *me-monitor traffic* pada jaringan. MRTG pertama kali dibuat oleh Tobias Oetiker pada tahun 1994 dengan menggunakan protokol *Simple Network Management Protocol (SNMP)*

yang biasanya dimiliki oleh setiap *interface* jaringan (antara lain *hub*, *switch*, *router*, *network card* atau NIC, *access point*, dan lain-lain).

Pada saat itu, jaringan internet masih mempunyai sambungan internet dengan kecepatan 64 Kbps yang menghubungkan lebih dari 1000 komputer. Hal ini membuat *user* sangat tertarik untuk mengetahui keadaan status jaringan terutama pada *router* utamanya. Program MRTG aslinya ditulis menggunakan bahasa *script* Perl yang menggunakan utilitas eksternal untuk mengambil data SNMP dan membuat gambar tipe GIF untuk tampilan halaman HTML. Ketika MRTG diumumkan pada tahun 1995, *software* ini menyebar dengan cepat dan banyak dipergunakan.

MRTG menggunakan SNMP untuk mengirimkan *request* dengan dua *Object Identifier* (OID) ke peralatan yang digunakan. Peralatan yang menyediakan SNMP akan mempunyai *Management Information Base* (MIB) untuk melihat spesifikasi OID. Setelah mengumpulkan informasi, MIB akan mengirimkan kembali data yang dienkapsulasi pada protokol SNMP. MRTG mencatat data ini pada statistik (*log*) di sisi *client* bersamaan dengan data yang telah dicatat sebelumnya untuk peralatan yang digunakan pada jaringan. Kemudian MRTG membuat dokumen halaman HTML berbentuk gambar yang menyediakan visualisasi secara langsung berisi sekumpulan grafik mengenai keadaan *traffic* jaringan.

Keuntungan memakai *tool* MRTG adalah dari faktor kesederhanaan dan fungsionalitasnya. MRTG bisa dikonfigurasi dengan mudah untuk memantau penggunaan *bandwidth* akan suatu *interface* yang mendukung SNMP dan juga bisa memantau peningkatan *traffic* dalam berbagai skala (harian, rata-rata setiap

lima menit) sampai skala tahunan. Dalam hal ini, *user* dapat dengan mudah melihat jika ada lonjakan *traffic* yang menandakan ada sesuatu yang tidak beres dalam jaringan.