BAB II
TEORI PENUNJANG

2.1 Jaringan Komputer

Berikut ini adalah definisi tentang jaringan komputer, struktur fisik jaringan, topologi fisik jaringan, dan kategori jaringan berdasarkan jangkauan geografisnya.

2.1.1 Definisi Jaringan

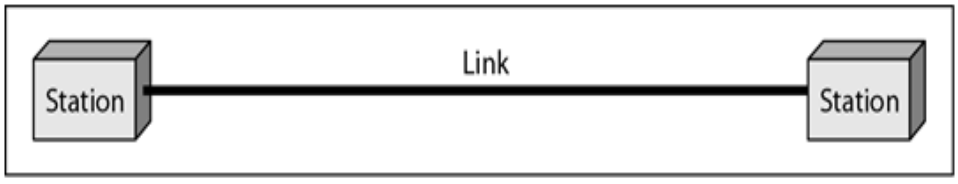
Jaringan adalah seperangkat alat (lebih sering disebut sebagai *node*) yang dihubungkan oleh jalur komunikasi. Sebuah *node* dapat berupa komputer, printer, atau peralatan lain yang mampu mengirim atau menerima data dari *node* lain dalam suatu jaringan [1].

2.1.2 Struktur Fisik Jaringan

Sebuah jaringan adalah beberapa perangkat yang saling berhubungan menggunakan *link*. *Link* adalah sebuah jalur komunikasi untuk mengirim data dari satu perangkat ke perangkat yang lain. Terdapat dua jenis koneksi, yaitu *point-to-point* dan *multipoint* [1].

1. *Point-to-point*

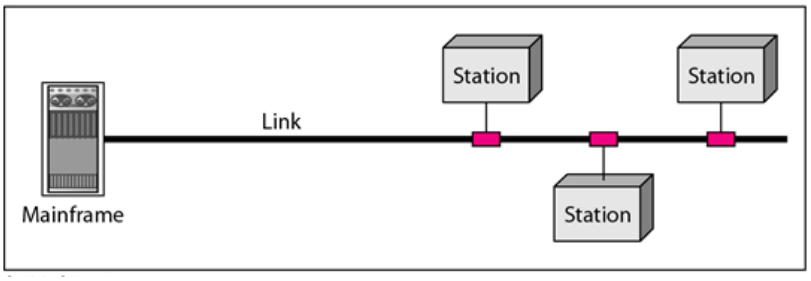
   Sebuah koneksi *point-to-point* adalah dua buah perangkat yang saling berhubungan menggunakan sebuah saluran. Kebanyakan koneksi *point-to-point* menggunakan kabel, bisa juga menggunakan *microwave* atau *link* satelit. Berikut ini merupakan gambar *point-to-point*:



Gambar 2.1 *Point-to-point*

2. *Multipoint*

Sebuah koneksi *multipoint* adalah koneksi antara satu atau beberapa perangkat dalam satu saluran yang digunakan bersama. Berikut ini merupakan gambar *multipoint*:



Gambar 2.2 *Multipoint*

2.1.3 Topologi Fisik Jaringan

Topologi fisik jaringan adalah representasi geometris dari hubungan semua saluran dan perangkat yang saling berhubungan satu sama lain. Topologi fisik jaringan merupakan desain aktual yang dibangun untuk menghubungkan beberapa perangkat dalam sebuah jaringan [1]. Berikut adalah beberapa jenis dari topologi fisik:

1. Topologi Bus

Topologi bus merupakan topologi yang menggunakan koneksi *multipoint*, karena memiliki satu saluran yang digunakan bersama untuk berbagi data. Dalam topologi bus setiap *node* dihubungkan oleh kebel yang langsung dihubungkan dengan saluran utama. Setiap kabel yang berhubungan dengan saluran utama dihubungkan oleh *tap*. Sinyal akan semakin melemah jika jarak yang ditempuhnya semakin jauh. Oleh karena itu perlu adanya beberapa *tap* untuk menjaga agar sinyal tidak melemah. Topologi ini seringkali dijumpai pada sistem *client/server*, dimana salah satu mesin pada jaringan tersebut difungsikan sebagai *file server*, yang berarti bahwa mesin tersebut dikhususkan hanya untuk pendistribusian data dan biasanya tidak digunakan untuk pemrosesan informasi.

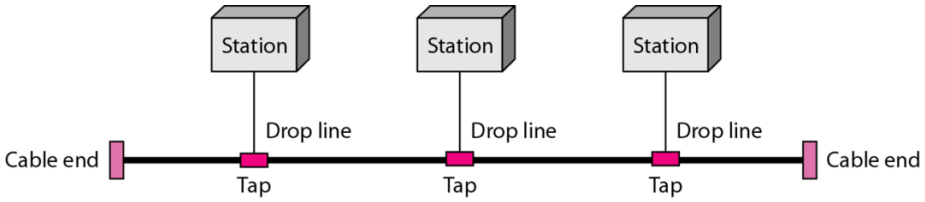
Kelebihan topologi bus:

1. Hemat kabel.
2. *Layout* kabel sederhana.
3. Pengembangan jaringan atau penambahan *workstation* baru dapat dilakukan dengan mudah tanpa mengganggu *workstation* lain.

Kekurangan topologi bus:

1. Deteksi dan isolasi kesalahan sangat kecil.
2. Kepadatan lalu lintas.
3. Bila salah satu *client* rusak, maka jaringan tidak bisa berfungsi.
4. Diperlukan *repeater* untuk jarak jauh.

Berikut ini merupakan gambar dari topologi bus:



Gambar 2.3 Topologi Bus

2. Topologi Ring

Pada topologi ring setiap perangkat berhubungan hanya dengan perangkat yang berada di sebelahnya menggunakan koneksi *point-to-point*, membentuk jalur melingkar membentuk cincin. Sinyal dilewatkan dalam satu arah dari satu perangkat ke perangkat lainnya hingga mencapai tujuan. Setiap perangkat dihubungkan dengan saluran utama menggunakan *repeater*. Sinyal akan semakin melemah jika jarak yang ditempuhnya semakin jauh, sehingga diperlukan *repeater*. Dengan adanya fungsi *repeater* ini maka sinyal data tersebut akan di-*generate* kembali sebelum diteruskan ke komputer yang dituju. Pada topologi ring, komunikasi data dapat terganggu jika satu titik mengalami gangguan. Jaringan *Fiber Distributed Data Interface* (FDDI) mengantisipasi kelemahan ini dengan mengirim data searah jarum jam dan berlawanan dengan arah jarum jam secara bersamaan.

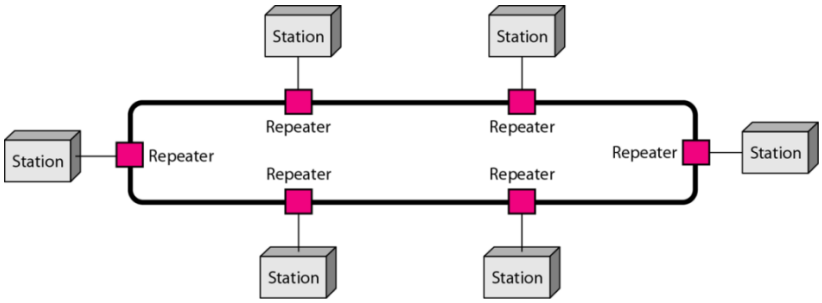
Kelebihan topologi ring:

1. Hemat kabel
2. Tidak terjadinya *collision* atau tabrakan pengiriman data seperti pada topologi bus, karena hanya satu *node* dapat mengirim data pada suatu saat.

Kekurangan topologi ring:

1. Setiap *node* dalam jaringan akan selalu ikut serta mengelola informasi yang dilewatkan dalam jaringan, sehingga jika terdapat ganguan di suatu *node* maka seluruh jaringan akan tergangu.
2. Pengembangan jaringan lebih kaku.

Berikut ini merupakan gambar dari topologi ring:



Gambar 2.4 Topologi Ring

3. Topologi Star

Pada topologi star, setiap perangkat memiliki saluran *point-to-point* hanya ke sebuah perangkat pusat, biasanya disebut hub. Perangkat tidak berhubungan langsung dengan perangkat lain. Pada topologi star jika suatu perangkat ingin mengirimkan data ke perangkat lain, perangkat pengirim harus mengirimkannya melalui hub, kemudian hub tersebut menyampaikan ke perangkat yang terhubung lainnya.

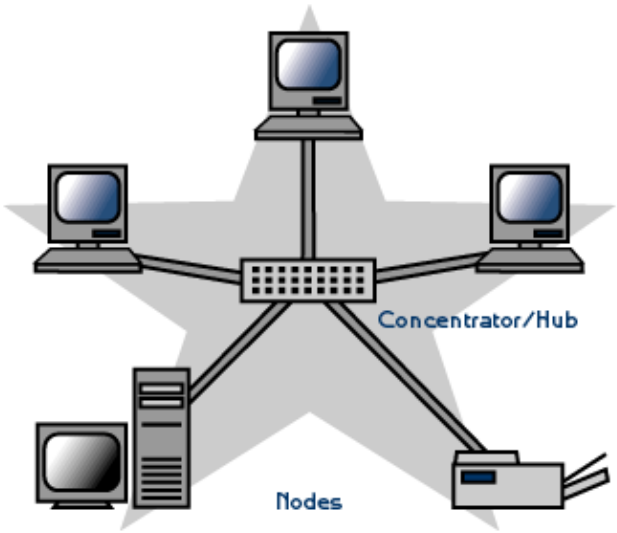
Kelebihan dari topologi star:

1. Kerusakan pada satu saluran hanya akan mempengaruhi jaringan pada saluran tersebut dan perangkat yang terpaut.

2. Tahan terhadap lalu lintas jaringan yang sibuk.
3. Penambahan dan pengurangan perangkat dapat dilakukan dengan mudah.

Kekurangan dari topologi star adalah jika *node* tengah mengalami kerusakan, maka seluruh jaringan akan terhenti.

Berikut ini merupakan gambar dari topologi star:



Gambar 2.5 Topologi Star

4. Topologi Tree

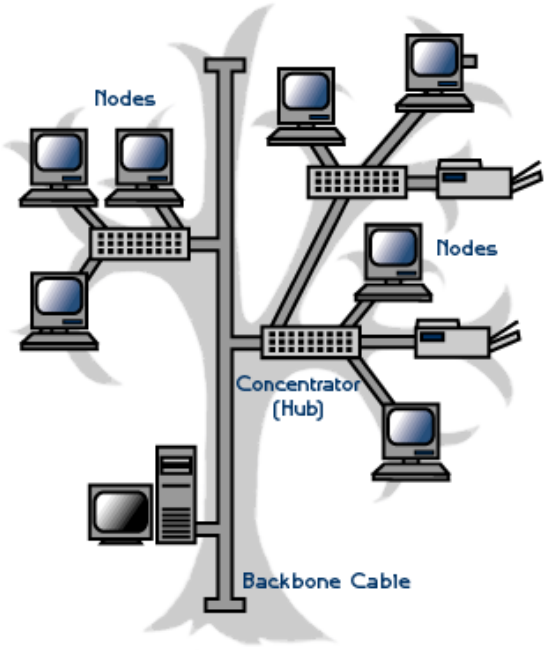
Topologi tree pada dasarnya merupakan bentuk yang lebih luas dari topologi star. Seperti halnya topologi star, perangkat (*node, device*) yang ada pada topologi tree juga terhubung kepada sebuah pusat pengendali (sentral hub atau switch) yang berfungsi mengatur lalu lintas di dalam jaringan. Meskipun demikian, tidak semua perangkat pada topologi tree terhubung secara langsung ke sentral hub. Pada topologi tree terdapat dua atau lebih hub yang digunakan untuk menghubungkan setiap perangkat ke dalam jaringan. Berdasarkan fungsinya, hub tersebut terbagi menjadi dua bagian, yaitu hub aktif dan hub pasif. Hub aktif berfungsi tidak hanya sekedar sebagai penerus sinyal data dari satu komputer ke komputer lainnya, tetapi juga memiliki fungsi sebagai *repeater*. Sedangkan hub pasif hanya berfungsi sebagai penerus sinyal data dari satu komputer ke komputer lainnya.

Kelebihan topologi tree:

1. Pemasangannya relatif mudah, karena setiap alat atau komputer hanya membutuhkan satu port I/O.
2. Deteksi kesalahan cukup mudah.
3. Kerusakan pada salah satu saluran tidak mempengaruhi sambungan yang lain, asalkan hub masih berfungsi.
4. Adanya hub sekunder memungkinkan lebih banyak perangkat yang dapat tersambung ke sebuah hub sentral, sehingga menambah jarak jangkauan jaringan.

Kekurangan topologi tree hampir sama dengan kekurangan pada topologi star, yaitu jika *node* sentral mengalami kerusakan, maka seluruh jaringan akan terhenti.

Berikut ini merupakan gambar dari topologi tree:

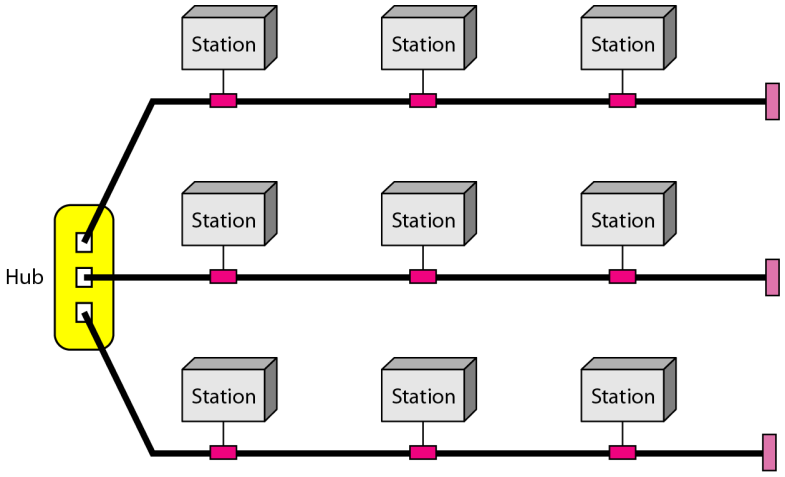


Gambar 2.6 Topologi Tree

5. Topologi Hybrid

Topologi hybrid merupakan gabungan dari beberapa topologi. Dalam hal ini topologi yang digunakan adalah topologi star yang setiap cabangnya

terhubung ke beberapa *station* pada topologi bus. Berikut ini merupakan gambar dari topologi hybrid:



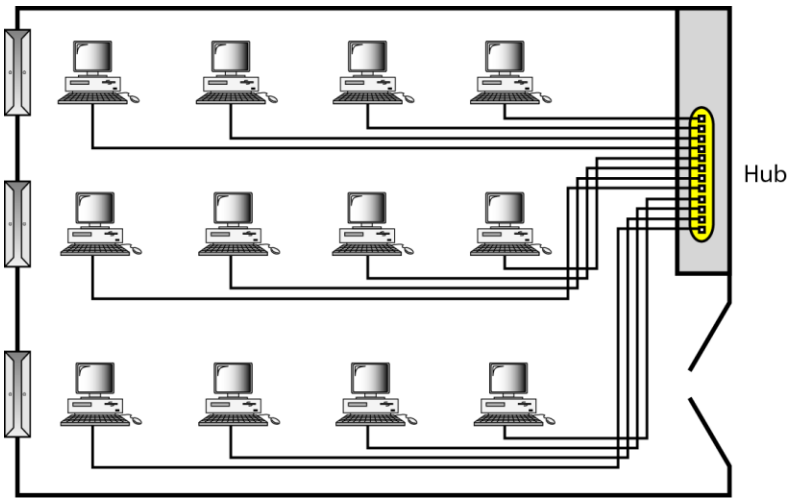
Gambar 2.7 Topologi Hybrid

2.1.4 Kategori Jaringan

Kategori jaringan komputer berdasarkan jangkauan geografis [1]:

1. *Local Area Network* (LAN)

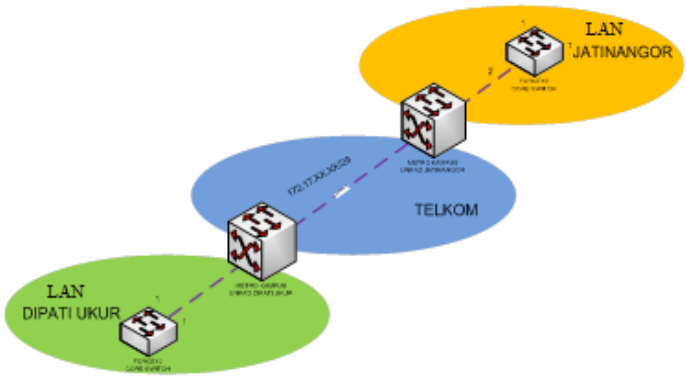
Merupakan suatu jaringan yang mencakup wilayah kecil atau dalam satu ruangan yang menghubungkan dua komputer atau lebih dalam cakupan seperti kantor, gedung, rumah, dan warnet.



Gambar 2.8 *Local Area Network*

2. *Metropolitan Area Network* (MAN)

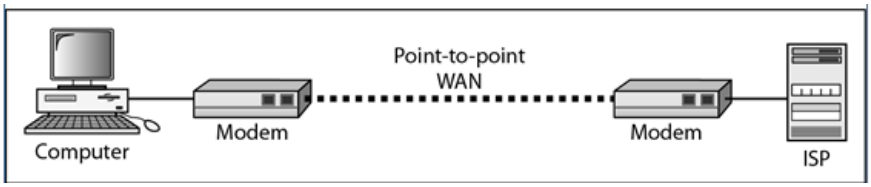
Merupakan suatu jaringan yang mencakup satu kota besar beserta daerah setempat. Jaringan MAN adalah gabungan dari beberapa jaringan LAN, yang menghubungkan berbagai lokasi seperti kampus, perkantoran, dan pemerintahan. Jangkauan dari jaringan MAN adalah 10km sampai 50km.



Gambar 2.9 *Metropolitan Area Network*

3. *Wide Area Network* (WAN)

Merupakan suatu jaringan yang mencakup seluruh dunia dalam artian sangat luas. WAN digunakan untuk menghubungkan jaringan area lokal yang satu dengan jaringan lokal lain, sehingga komputer di lokasi yang satu dapat berkomunikasi dengan komputer di lokasi yang lain.



Gambar 2.10 *Wide Area Network*

2.2 Protokol Jaringan

Dalam jaringan komputer, komunikasi terjadi antar *entity* dalam sistem yang berbeda. Sebuah *entity* adalah segala sesuatu yang mampu mengirim atau menerima informasi. Bagaimanapun juga, dua *entity* tidak bisa dengan mudah

mengirimkan bit satu sama lain dan berharap untuk dimengerti. Supaya komunikasi terjadi, suatu *entity* harus menggunakan protokol. Protokol adalah seperangkat aturan yang mengatur pertukaran informasi antar komputer. Elemen utama dari sebuah protokol adalah *syntax*, *semantic*, dan *timing*. [1]

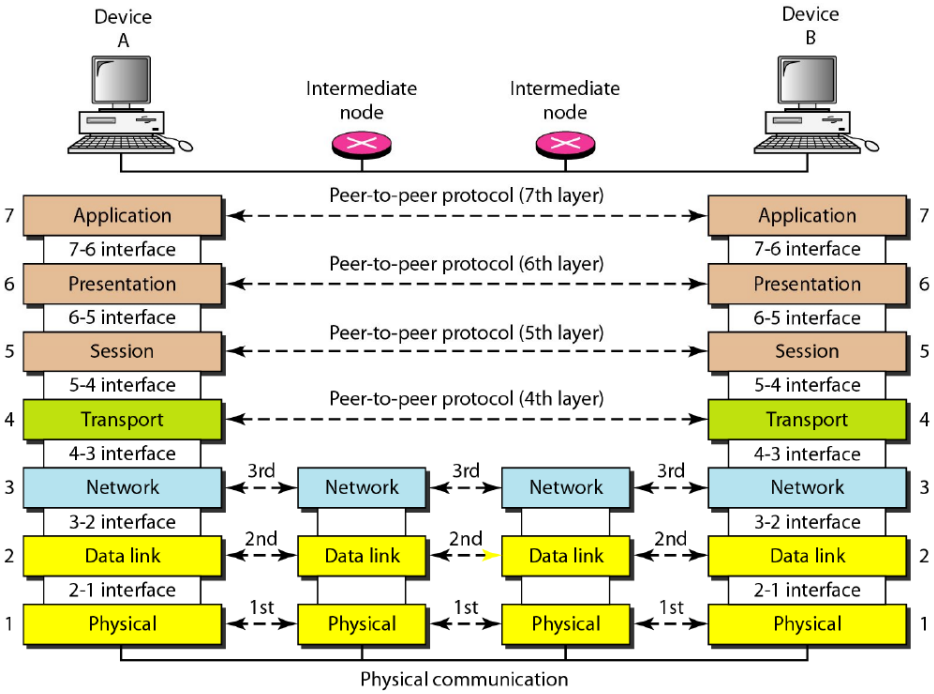
1. *Syntax* merupakan struktur atau format dari suatu data. Contohnya, sebuah protokol sederhana dari suatu data yang terdiri dari 8 bit pertama merupakan alamat dari pengirim dan 8 bit kedua merupakan alamat dari penerima.
2. *Semantic* merupakan pengertian yang dikandung oleh setiap pola bit dan byte. Contohnya, *syntax*-nya itu 10101010... *semantic*-nya *please synchronize*..
3. *Timing* terdiri dari dua karakteristik, yaitu kapan data harus dikirimkan dan seberapa cepat data dapat dikirimkan. Contohnya, jika pengirim mengirim data dengan kecepatan 100Mbps, tetapi penerima hanya dapat memproses data sebesar 1Mbps, transmisi tersebut akan melebihi kapasitas penerima sehingga sebagian data akan hilang.

Protokol internet pertama kali dirancang pada awal tahun 1980-an. Pada saat itu protokol tersebut hanya digunakan untuk beberapa *node* saja. Kemudian pada awal tahun 1990-an internet mulai berkembang dan banyak protokol yang bermunculan. *International Standard Organization* (ISO) membuat standardisasi protokol yang saat ini dikenal dengan protokol model *Open System Interconnection* (OSI) yang kemudian berkembang menjadi TCP/IP [2].

2.3 Model *Open System Interconnection* (OSI)

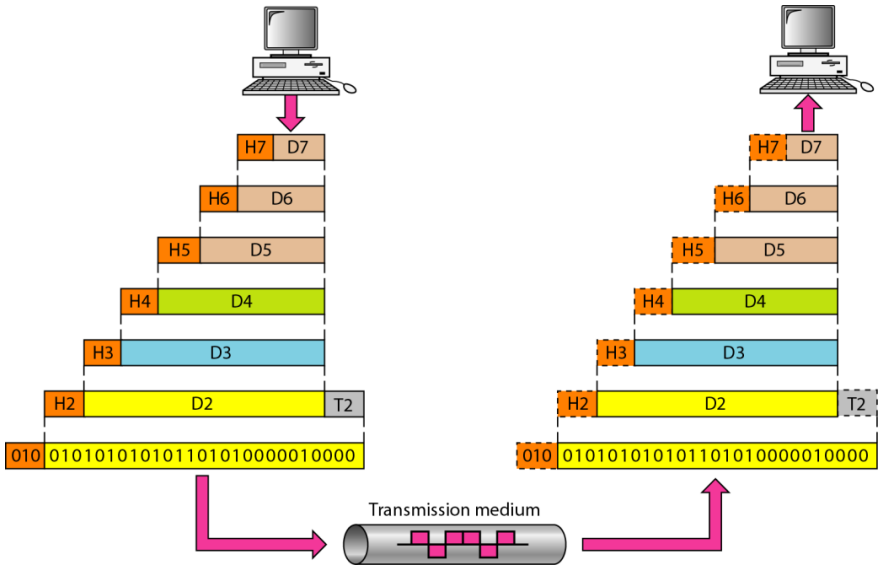
Model OSI adalah pembagian lapisan dalam mendesain suatu sistem jaringan yang mengizinkan semua jenis sistem komputer untuk saling berkomunikasi. Terbagi dalam tujuh lapisan yang saling berhubungan, setiap lapisan mendefiniskan proses pemindahan informasi lintas jaringan [1]

Proses yang dilakukan oleh perangkat untuk melakukan komunikasi pada lapisan adalah proses *peer-to-peer*. Komunikasi antar perangkat yang dilakukan secara *peer-to-peer* menggunkan protokol yang telah disediakan pada setiap lapisan. Berikut ini merupakan gambar proses *peer-to-peer*:



Gambar 2.11 Interaksi antar Lapisan pada Model OSI

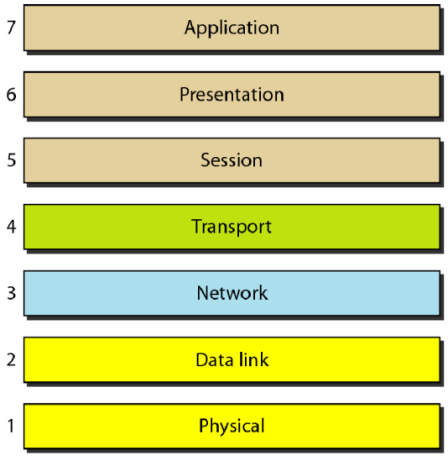
Transmisi data sebernarnya berupa transmisi vertikal. Data yang dikirimkan dibagian ujungnya akan ditambahkan *header* dan itu dilakukan pada setiap lapisan. Setelah data sampai di perangkat tujuan satu persatu *header* tersebut akan dilepaskan.



Gambar 2.12 Proses Pengiriman Data dalam Model OSI

2.4 Lapisan Model OSI

Berikut ini merupakan gambar dari tujuh lapisan model OSI [1]:



Gambar 2.13 Lapisan Model OSI

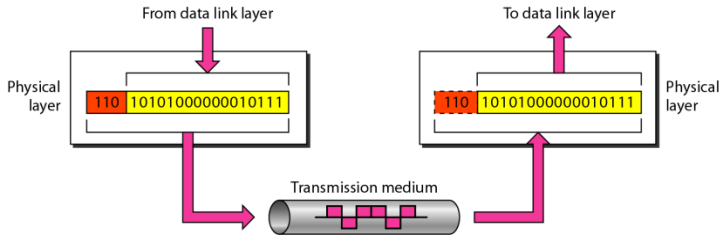
1. Lapisan Fisik (*Physical Layer*)

Lapisan fisik berfungsi sebagai penerus aliran bit/data pada media fisik jaringan. Berurusan dengan spesifik mekanik dan elektris sebuah antarmuka dan media transmisi. Lapisan ini menetapkan prosedur dan fungsi dari perangkat fisik untuk melakukan komunikasi. Lapisan fisik bertanggung jawab dalam perpindahan bit dari satu *node* ke *node* berikutnya. Berikut ini merupakan tujuan dari adanya lapisan fisik:

1. Karakter fisik dari antarmuka dan media. Lapisan fisik mendefinisikan karakteristik dari antarmuka antara perangkat dan media transmisi. Selain itu mendefiniskan jenis media transmisi.
2. Representasi dari bit. Data pada lapisan fisik terdiri dari bit-bit biner tanpa interpretasi. Supaya bisa dikirimkan, bit-bit tersebut harus dirubah dalam bentuk sinyal elektris atau optik. Lapisan fisik mendefiniskan jenis *encoding*, mengubah bit ke dalam bentuk sinyal.
3. *Data Rate*. Kecepatan pengiriman adalah jumlah bit yang dikirimkan per detik, didefinisikan oleh lapisan fisik. Dengan kata lain lapisan fisik mendefinisikan lamanya bit dikirimkan.

4. Konfigurasi saluran. Pada konfigurasi *point-to-point* dua perangkat disambungkan oleh sebuah saluran yang sifatnya *dedicated*. Pada konfigurasi *multipoint* sebuah saluran digunakan oleh beberapa perangkat.
5. Topologi Fisik. Topologi fisik mendefinisikan bagaimana perangkat dapat berhubungan dalam membentuk sebuah jaringan.
6. Cara Transmisi. Lapisan fisik juga mendefinisikan arah dari transmisi data antar dua perangkat. *Simplex*, *half-duplex*, atau *full-duplex*. Pada jenis *simplex*, hanya satu perangkat yang mengirim, yang lainnya hanya menerima. Pada *half-duplex*, dua perangkat dapat mengirim dan menerima, tetapi tidak dalam satu waktu. Pada *full-duplex*, dua perangkat dapat mengirim dan menerima dalam waktu yang sama.

Gambar 2.14 menunjukkan posisi lapisan fisik yang berhubungan dengan media transmisi dan lapisan jalur data.



Gambar 2.14 Lapisan Fisik

2. Lapisan Jalur Data (*Data Link Layer*)

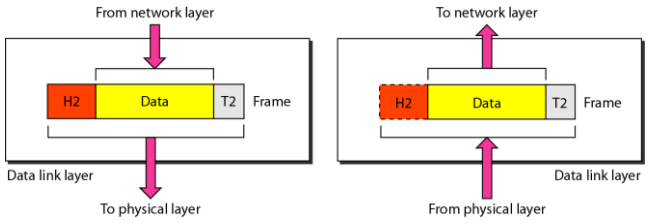
Lapisan jalur data berfungsi sebagai fasilitas transmisi *raw* data dan mentransformasikan data ke saluran yang bebas masalah dari kesalahan transmisi. Lapisan jalur data bertanggung jawab dalam memindahkan *frame* dari satu *node* ke *node* selanjutnya. Berikut ini merupakan yang menjadi tanggung jawab lapisan jalur data:

1. Penyusunan. Lapisan jalur data membagi aliran bit yang diterima dari lapisan jaringan menjadi unit data yang dapat diatur yang disebut *frame*.
2. Pengalamatan fisik. Jika *frame* didistribusikan ke sistem yang berbeda pada suatu jaringan, lapisan jalur data menambahkan sebuah *header* ke

dalam *frame* untuk mendefinisikannya sebagai pengirim atau penerima dari *frame* tersebut.

3. Kontrol aliran. Jika *data rate* yang diterima oleh penerima lebih kecil dari *data rate* yang dikirim oleh pengirim, lapisan jalur data menentukan mekanisme aliran data untuk menghindari terjadinya kelebihan data pada sisi penerima.
4. Kontrol kesalahan. Lapisan jalur data menjadikan lapisan fisik dapat diandalkan dengan cara menambahkan mekanisme untuk mendeteksi dan mengirimkan ulang *frame* yang rusak atau hilang.
5. Kontrol akses. Ketika dua perangkat berhubungan dalam saluran yang sama, protokol lapisan jalur data memberikan hak kepada satu perangkat untuk menggunakan saluran dalam waktu yang sudah ditentukan.

Gambar 2.15 menunjukkan hubungan lapisan jalur data dengan lapisan jaringan dan lapisan fisik.



Gambar 2.15 Lapisan Jalur Data

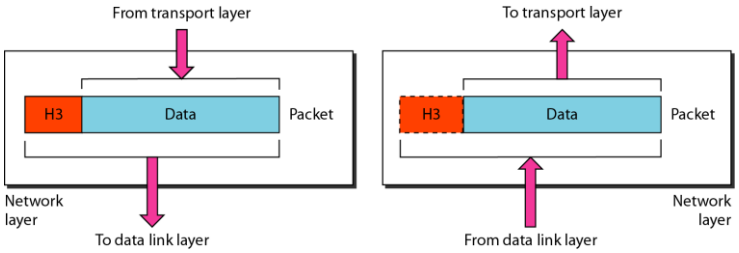
3. Lapisan Jaringan (*Network Layer*)

Lapisan jaringan bertanggung jawab atas sumber dan tujuan terkirimnya suatu paket, memungkinkan lintas jaringan. Mengingat lapisan jalur data mengatur pengiriman paket antar dua sistem dalam jaringan yang sama, lapisan jaringan menjamin keutuhan paketnya sampai ke tujuan.

Jika dua sistem berhubungan dalam saluran yang sama, itu biasanya tidak membutuhkan lapisan jaringan. Tetapi jika dua sistem saling berhubungan dalam jaringan yang berbeda barulah menggunakan lapisan jaringan dalam menentukan sumber dan tujuan pengiriman. Lapisan jaringan bertanggung jawab dalam pengiriman paket dari *host* sumber ke *host* tujuan. Yang termasuk tanggung jawab dari lapisan jaringan diantaranya:

1. Pengalamatan logik. Jika sebuah paket melewati batas jaringan, dibutuhkan pengalamatan lain untuk membedakan sistem sumber dan sistem tujuan. Lapisan jaringan menambahkan sebuah *header* ke dalam paket yang datang dari lapisan di atasnya termasuk alamat logik dari pengirim dan penerima.
2. *Routing*. Ketika saluran terhubung membentuk *internetwork* atau jaringan yang besar, perangkat penghubung yang sering dikenal dengan router atau switch menentukan rute suatu paket supaya sampai ditujuan akhir. Salah satu dari fungsi lapisan jaringan adalah menyediakan mekanisme tersebut. Misal, ketika sebuah paket tiba di suatu router, router tersebut akan menyampaikan paket ke router tujuan dengan menggunakan tabel *routing* untuk menentukan *hop* selanjutnya sebelum paket tersebut dikirimkan ke router tujuan terakhir.

Gambar 2.16 menunjukkan hubungan lapisan jaringan dengan lapisan jalur data dan lapisan transport.



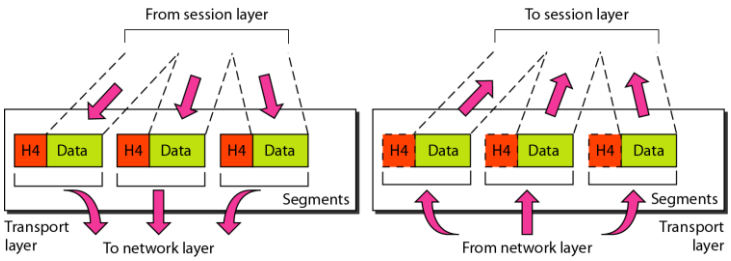
Gambar 2.16 Lapisan Jaringan

4. Lapisan Transport (*Transport Layer*)

Lapisan transport bertanggung jawab atas proses pengiriman semua pesan. Mengingat lapisan jaringan mengatur pengiriman paket dari sumber ke tujuan, lapisan jaringan tidak mengenali hubungan antara paket-paket tersebut. Lapisan transport memastikan seluruh paket tiba dengan utuh dan sesuai pesanan. Lapisan transport bertanggung jawab atas pengiriman sebuah pesan dari satu proses ke proses lainnya. Beberapa hal yang menjadi tanggung jawab dari lapisan transport:

1. Pengalamatan *service-point*. Untuk menjamin pengiriman pesan dari proses suatu komputer ke proses komputer selanjutnya *header* lapisan transport harus memiliki alamat *service-point* (atau alamat *port*).
2. Segmentasi dan menyusun ulang paket. Sebuah pesan yang dibagi menjadi beberapa bagian yang memiliki nomor urut pada saat pengiriman pesan. Lapisan transport menyusun pecahan paket tersebut berdasarkan nomor urutnya sehingga pada saat paket diterima di tujuan paket tersebut utuh.
3. Kontrol koneksi. Lapisan transport dapat bersifat *connectionless* atau *connection-oriented*. Lapisan transport yang bersifat *connectionless* memperlakukan setiap paket sebagai paket independen kemudian mengirimnya ke lapisan transport pada mesin tujuan. Sedangkan lapisan transport yang bersifat *connection-oriented* membuat koneksi dulu dengan lapisan transport yang ada di mesin tujuan sebelum paket dikirimkan. Setelah semua data dikirimkan, koneksi terputus.
4. Kontrol Aliran. Seperti pada lapisan jalur data, lapisan transport bertanggung jawab sebagai pengontrol aliran. Pengaturan aliran pada layer ini dilakukan *end-to-end*.
5. Kontrol kesalahan. Kontrol kesalahan pada layer ini dilakukan pada *process-to-process*. Lapisan transport yang mengirimkan pesan memastikan pesan yang dikirim akan diterima oleh lapisan transport yang menerimanya dengan utuh tanpa kesalahan. Koreksi kesalahan dapat dilakukan dengan cara mengirimkan ulang pesan.

Gambar 2.17 menunjukkan hubungan lapisan transport dengan lapisan jaringan dan lapisan sesi.



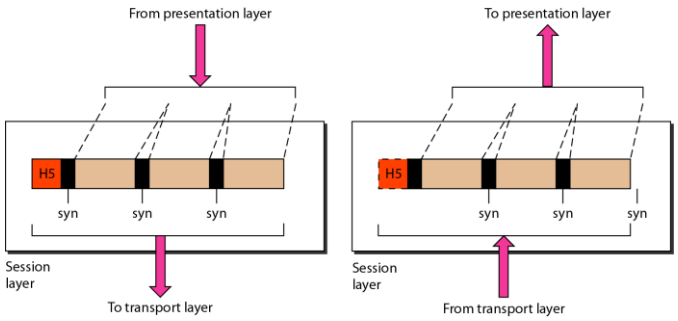
Gambar 2.17 Lapisan Transport

5. Lapisan Sesi (*Session Layer*)

Lapisan sesi adalah pengontrol dialog suatu jaringan. Lapisan sesi menetapkan, memelihara, dan melakukan sinkronisasi suatu interaksi saat sistem melakukan komunikasi. Lapisan sesi bertanggung jawab dalam kontrol dialog dan sinkronisasi. Tanggung jawab spesifik dari lapisan sesi adalah:

1. Kontrol dialog. Lapisan sesi mengizinkan dua sistem untuk berdialog. Lapisan sesi mengizinkan komunikasi antara dua proses untuk melakukan *half-duplex* atau *full duplex*.
2. Sinkronisasi. Lapisan sesi mengizinkan sebuah proses untuk menambahkan *checkpoint*, atau *synchronization point* untuk sebuah aliran data.

Gambar 2.18 menunjukkan hubungan lapisan sesi dengan lapisan transport dan lapisan presentasi.



Gambar 2.18 Lapisan Sesi

6. Lapisan Presentasi (*Presentation Layer*)

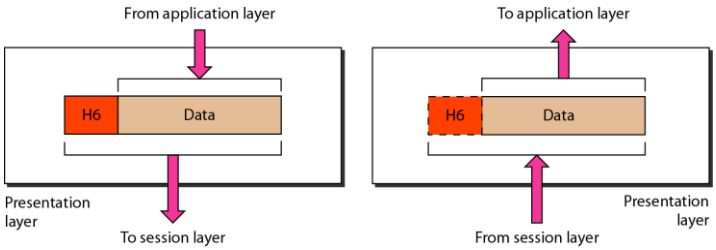
Lapisan presentasi berkonsentrasi pada *syntax* dan *semantic* pada pertukaran informasi antara dua sistem. Lapisan presentasi bertanggung jawab dalam translasi, kompresi, dan enkripsi. Tanggung jawab spesifik dari lapisan presentasi adalah:

1. Translasi. Proses (program yang sedang berjalan) pada dua sistem biasanya melakukan pertukaran informasi dalam bentuk karakter, nomor, dan sebagainya. Informasi harus diubah menjadi aliran bit sebelum

dikirimkankan. Karena komputer yang berbeda memiliki sistem *encoding* yang berbeda, lapisan presentasi bertanggung jawab dalam menangani perbedaan dari metode *encoding* tersebut. Lapisan presentasi yang berfungsi sebagai pengirim mengubah suatu informasi dari format *sender-dependent* ke dalam format yang umum. Lapisan presentasi mesin penerima mengubah format yang umum ke dalam format *receiver-dependent*.

2. Enkripsi. Dalam membawa informasi yang sifatnya sentitif, sebuah sitem harus dapat menjamin hal yang sifatnya privasi. Enkripsi berarti bahwa pengirim mentransformasikan informasi yang asli ke dalam format yang berbeda sebelum informasi dikirimkan. Dekripsi merupakan proses untuk mentransformasikan kembali pesan ke dalam bentuk yang asli.
3. Kompresi. Kompresi data mengurangi jumlah bit yang terkandung dalam suatu informasi. Kompresi data menjadi sangat penting pada transmisi pesan *multimedia*, diantaranya teks, audio, dan video.

Gambar 2.19 menunjukkan hubungan lapisan presentasi dengan lapisan sesi dan lapisan aplikasi.



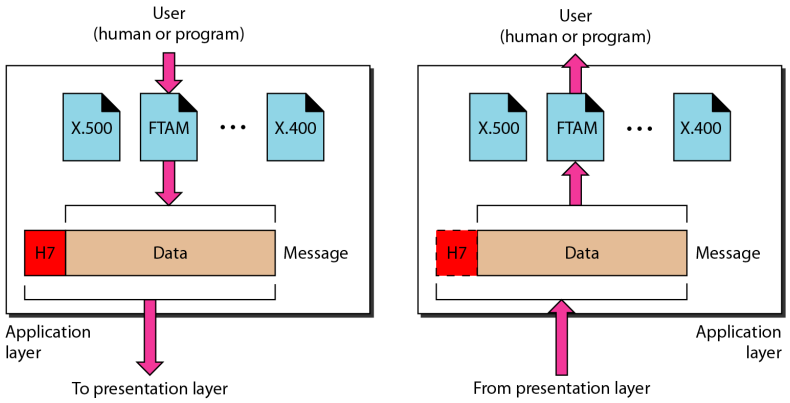
Gambar 2.19 Lapisan Presentasi

7. Lapisan Aplikasi (*Application Layer*)

Lapisan aplikasi mengizinkan pengguna, orang ataupun perangkat lunak untuk mengakses jaringan. Lapisan jaringan menyediakan antarmuka pengguna dan membantu dalam pelayanan seperti *e-mail*, manajemen *database*, dan layanan distribusi informasi lainnya. Lapisan aplikasi bertanggung jawab menyediakan layanan bagi pengguna internet. Lapisan aplikasi meyediakan layanan spesifik sebagai berikut:

1. *Network virtual terminal*. *Network virtual terminal* adalah versi perangkat lunak dari terminal fisik, dan itu mengizinkan seorang pengguna untuk *log on* ke sebuah *remote host*. Aplikasi membuat sebuah perangkat lunak dari terminal pada *remote host*.
2. Transfer, akses, dan manajemen file. Aplikasi ini mengizinkan pengguna untuk mengakses file dalam sebuah *remote host*.
3. Layanan *mail*. Aplikasi ini menyediakan dasar untuk meneruskan sebuah *e-mail* dan menyimpannya pada media penyimpanan.
4. Layanan direktori. Aplikasi ini menyediakan distribusi *database* sumber dan melakukan akses informasi global mengenai berbagai jenis objek dan layanan.

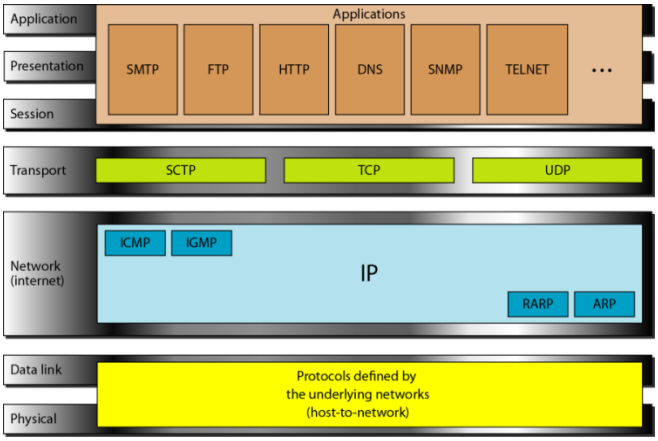
Gambar 2.20 menunjukkan hubungan lapisan aplikasi dengan pengguna dan lapisan presentasi. Contoh beberapa layanan yang diberikan oleh lapisan aplikasi diantaranya adalah X.400 (layanan *message-handling*), X.500 (layanan direktori), dan transfer, akses dan manajemen file (FTAM).



Gambar 2.20 Lapisan Aplikasi

2.5 Susunan Protokol TCP/IP

Lapisan pada susunan protokol TCP/IP tidak sama persis dengan model OSI. Susunan protokol TCP/IP memiliki empat lapisan, yaitu *host-to-network*, *internet*, *transport*, dan *application*. Gambar 2.21 merupakan gambar dari susunan protokol TCP/IP [1].



Gambar 2.21 Susunan Protokol TCP/IP

Tiap layer dijelaskan sebagai berikut:

1. Lapisan Host-to-Network (*Host-to-Network Layer*)

   Lapisan ini merupakan gabungan dari lapisan fisik dan lapisan jalur data pada lapisan model OSI. Terdiri dari protokol-protokol dasar jaringan seperti *ethernet*, *token ring*, *frame relay*, dan ATM.

2. Lapisan Internet (*Internet Layer*)

   Fungsinya sama dengan lapisan jaringan pada model OSI. Protokol yang bekerja pada lapisan ini diantaranya adalah *Internet Protocol* (IP), *Internet Control Message Protocol* (ICMP), *Internet Group Message Protocol* (IGMP), *Address Resolution Protocol* (ARP), *Reverse Address Resolution Protocol* (RARP).

3. Lapisan Transport (*Transport Layer*)

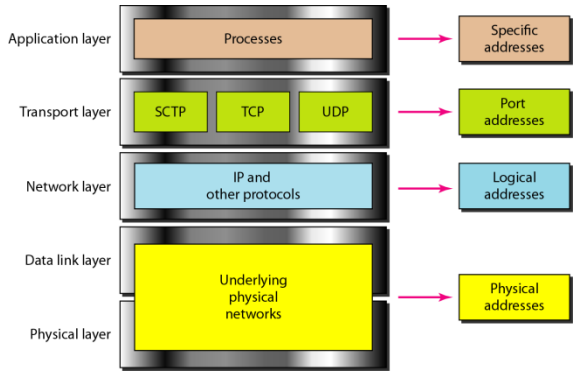
   Secara tradisional lapisan ini memiliki dua protokol, yaitu *Transmission Control Protocol* (TCP) dan *User Datagram Protocol* (UDP).

4. Lapisan Aplikasi (*Application Layer*)

   Lapisan aplikasi pada model TCP/IP merupakan gabungan dari lapisan sesi, presentasi dan aplikasi pada lapisan model OSI. Pada lapisan ini banyak protokol yang digunakan, diantaranya adalah *Simple Mail Transfer Protocol* (SMTP), *HyperText Transfer Protocol* (HTTP), dan lain sebagainya.

2.6 *Addressing*

Terdapat empat level alamat yang digunakan oleh sebuah internet, yaitu alamat fisik (*link*), alamat logik (IP), alamat *port*, dan alamat spesifik, namun dalam subbab yang dijelaskan hanya alamat fisik dan alamat logik. Setiap alamat berhubungan kepada setiap spesifik lapisan arsitektur TCP/IP, ditunjukkan pada gambar 2.22 [1].



Gambar 2.22 Hubungan Layer dan Alamat pada TCP/IP

2.6.1 Alamat Fisik (*Phsycal Addresses*)

Alamat fisik atau dikenal juga dengan alamat *link* adalah alamat dari sebuah *node* yang didefinisikan oleh LAN atau WAN. Alamat ini merupakan alamat level terendah. Ukuran dan format dari alamat ini bermacam-macam. Sebagai contoh, Ethernet menggunakan alamat yang berukuran 6-byte (48-bit) yang terdapat pada *Network Ethernet Card* (NIC). LocalTalk (Apple) memiliki 1-byte alamat dinamis yang berubah setiap waktu saat *station up*. Alamat fisik juga dikenal sebagai alamat *Media Access Control* (MAC). Contoh alamat fisik yang terdapat pada NIC, yaitu **07:01:02:01:2C:4B**.

2.6.2 Alamat Logik (*Logical Addresses*)

Alamat logik digunakan sebagai alamat dalam hubungan antar *host* di Internet sehingga merupakan sebuah sistem komunikasi yang universal. Alamat logik memiliki ukuran alamat sebesar 32-bit. Jika suatu komputer memiliki lebih dari satu antarmuka (misalkan menggunakan dua *ethernet*) maka harus diberikan dua alamat logik untuk setiap antarmukanya. Alamat logik disebut juga alamat IP.

2.7 IP *Address* Versi 4

IP *address* terdiri dari dua versi, yaitu IPv4 dan IPv6. Namun dalam subbab ini hanya akan menjelaskan IPv4. IPv4 memiliki ukuran sebesar 32-bit. IPv4 dibagi ke dalam lima kelas, yaitu A, B, C, D, dan E. Metode ini disebut dengan *classful addressing*. Pada umumnya yang digunakan dalam pengalamatan IPv4 untuk setiap *host* hanya tiga kelas, yaitu A, B, dan C. Berikut ini pembagian setiap kelas-kelasnya [3]:

1. Kelas A
   Format : 0nnnnnnn hhhhhhhh hhhhhhhh hhhhhhhh
   Byte Pertama : 0 – 127
   Jumlah : 126 (*Network* ID)
   Range IP : 1.xxx.xxx.xxx sampai 126.xxx.xxx.xxx
   Jumlah IP : 16.777.214 IP *address* untuk *host* yang berada pada segmen jaringan yang sama
2. Kelas B
   Format : 10nnnnnn nnnnnnnn hhhhhhhh hhhhhhhh
   Byte Pertama : 128 – 191
   Jumlah : 16.384 (*Network* ID)
   Range IP : 128.0.xxx.xxx sampai 191.255.xxx.xxx
   Jumlah IP : 65.532 IP *address* untuk *host* yang berada pada segmen jaringan yang sama
3. Kelas C
   Format : 110nnnn nnnnnnnn nnnnnnnn hhhhhhhh
   Byte Pertama : 192 – 223
   Jumlah : 2.097.152 (*Network* ID)
   Range IP : 192.0.0.xxx sampai 223.255.255.xxx
   Jumlah IP : 254 IP *address* untuk *host* yang berada pada segmen jaringan yang sama

Keterangan: n = *network* bit, h = *host* bit

2.8 *Subnetting*

Untuk beberapa alasan yang menyangkut efisiensi IP *address*, mengatasi masalah topologi jaringan dan organisasi, administrator jaringan biasanya melakukan *subnetting*. Esensi dari *subnetting* adalah ‘‘memindahkan” garis pemisah antara bagian *network* dan bagian *host* dari suatu IP *address*. Beberapa bit dari bagian *host* dialokasikan menjadi bit tambahan pada bagian *network*. Menurut struktur baku alamat suatu *network* dipecah menjadi beberapa *subnetwork*. Cara ini menciptakan sejumlah *network* tambahan, tetapi mengurangi jumlah maksimum *host* yang ada dalam tiap *network* tersebut [3].

Suatu *subnet* didefinisikan dengan mengimplementasikan *masking* bit (*subnet mask*) kepada IP *address*. Struktur *subnet mask* sama dengan struktur IP *address*, yaitu terdiri dari 32-bit yang dibagi atas empat segmen. Bit-bit dari IP *address* yang “ditutupi” (*masking*) oleh bit-bit *subnet mask* yang aktif dan bersesuaian akan diinterpretasikan sebagai *network* bit. Bit 1 pada *subnet mask* berarti mengaktifkan *masking* (on), sedangkan bit 0 tidak aktif (off). Sebagai contoh kasus, IP *address* kelas A 44.132.1.20/16. Ilustrasinya dapat dilihat pada tabel berikut [3]:

Tabel 2.1. Contoh IP *address* Kelas A Sebelum *Subnetting*

IP *Address*	44	132	1	20
	00101100	10000100	00000001	00010100
Subnet Mask	255	0	0	0
	11111111	00000000	00000000	00000000
Network Address	44	0	0	0
	00101100	00000000	00000000	00000000
Broadcast Address	44	255	255	255
	00101100	11111111	11111111	11111111

Dengan aturan standard, *network* ID dari IP *address* ini adalah 44 dan *host* ID adalah 132.1.20. Jaringan tersebut dapat menampung maksimum lebih dari 16.000.000 *host* yang terhubung langsung. Pada IP *address* diimplementasikan

subnet mask sebanyak 16-bit 255.255.0.0 (Hexa = FF.FF.00.00 atau Biner = 11111111.11111111.00000000.00000000). Pada 16-bit pertama dari *subnet mask* tersebut bernilai 1, sedangkan 16-bit berikutnya bernilai 0. Dengan demikian, 16-bit pertama dari suatu IP *address* yang dikenakan *subnet mask* tersebut akan dianggap sebagai *network* bit. *Network* ID akan berubah menjadi 44.132 dan *host* ID menjadi 1.20. Kapasitas maksimum *host* yang langsung terhubung pada jaringan menjadi sekitar 65.532 *host*. Setelah dilakukan *subnetting* IP *address* kelas A tersebut menjadi seperti pada tabel berikut [3]:

Tabel 2.2. Contoh IP *address* Kelas A Setelah *Subnetting*

IP *Address*	44	132	1	20
	00101100	10000100	00000001	00010100
Subnet Mask	255	255	0	0
	11111111	11111111	00000000	00000000
Network Address	44	132	0	0
	00101100	10000100	00000000	00000000
Broadcast Address	44	132	255	255
	00101100	10000100	11111111	11111111

Subnet mask di atas identik dengan standar IP *address* kelas B. Dengan menerapkan *subnet mask* tersebut pada satu *network* kelas A, dapat dibuat 256 *network* baru dengan kapasitas masing-masing *subnet* setara *network* kelas B. Penerapan *subnet* yang lebih jauh seperti 255.255.255.0 (24-bit) pada kelas A akan menghasilkan jumlah *network* yang lebih besar (lebih dari 65.000 *network*) dengan kapasitas masing-masing *subnet* sebesar 256 *host*. Network kelas C juga dapat dibagi-bagi lagi menjadi beberapa *subnet* dengan menerapkan *subnet mask* yang lebih tinggi seperti untuk 25-bit (255.255.255.128), 26-bit (255.255.255.192), 27-bit (255.255.255.224) dan seterusnya [3].

2.9 Media Transmisi

Media transmisi adalah segala sesuatu yang dapat membawa informasi dari sumber ke tujuan. Media transmisi dibagi menjadi dua kategori, yaitu media terpandu (*guided media*) dan media tidak terpandu (*unguided media*) [1].

2.9.1 Media Terpandu (*Guided Media*)

Media terpandu adalah media yang mengalirkan data melalui suatu jalur fisik tertentu.

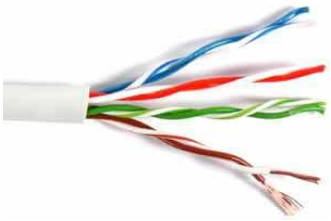
1. Kabel *Unshielded Twisted-Pair* (UTP)

   Umumnya yang dipakai dalam jaringan komputer terdiri dari empat pasang kabel/delapan kabel dan dihubungkan dengan konektor RJ-45. Kabel UTP dapat mempunyai *transfer rate* sebesar 10Mbps sampai dengan 200Mbps, tetapi mempunyai jarak jangkau yang pendek, yaitu maksimum 100m. Terdapat 6 kategori kabel UTP, yaitu sebagai berikut:

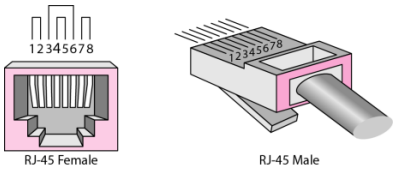
Tabel 2.3. Kategori Kabel UTP

Kategori	**Fungsi**	***Data Rate* (Mbps)**
1	Digunakan untuk telekomunikasi telepon.	< 0.1
2	Digunakan dalam T-line.	2
3	Peningkatan dari CAT 2 digunakan dalam LAN.	10
4	Peningkatan dari CAT 3 digunakan dalam jaringan Token Ring.	20
5	Merupakan jenis yang paling populer dipakai dalam jaringan komputer di dunia pada saat ini.	100
6	Sebuah kategori baru untuk meningkatkan kecepatan dalam transmisi data.	200

Berikut ini merupakan gambar kabel UTP dan konektor RJ-45:



Gambar 2.23 Kabel UTP



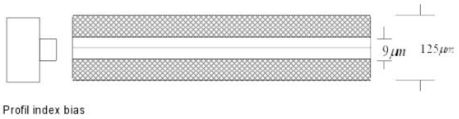
Gambar 2.24 Konektor RJ-45

2. Kabel Serat Optik

Kabel serat optik adalah suatu media transmisi yang menggunakan media cahaya sebagai penyalur informasi. Pada teknologi ini terjadi perubahan informasi yang biasanya berbentuk sinyal elektris menjadi sinyal optik (cahaya), yang kemudian disalurkan melalui kabel serat optik dan diterima pada sisi penerima untuk diubah kembali menjadi sinyal elektris. Jenis serat dapat dibagi menjadi dua macam, yaitu:

1. *Singlemode*

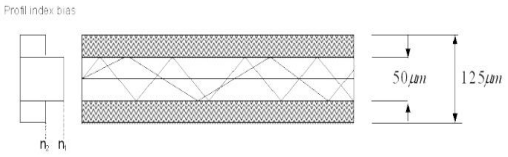
Serat *singlemode* mempunyai ukuran diameter *core* yang sangat kecil yaitu sekitar 4-10µm dan diameter *cladding* sebesar 125µm seperti pada gambar 2.25. Serat ini hanya dapat mentransmisikan sinyal dalam satu *mode*. Serat ini memiliki pita frekuensi transmisi yang lebar dan kapasitas transmisi yang besar.



Gambar 2.25 Kabel Serat Optik *Singlemode*

2. *Multimode*

Pada panjang gelombang operasi tertentu, jika serat optik mentransmisikan sinyal dalam berbagai mode, maka disebut serat *multimode*. Serat *multimode* biasanya memiliki diameter *core* antara 50-70µm dan diameter *cladding* antara 100-200µm seperti pada gambar 2.26. Jenis serat ini biasanya memiliki performansi transmisi yang buruk, *bandwidth* yang sempit dan kapasitas transmisi yang kecil.



Gambar 2.26 Kabel Serat Optik *Multimode*

2.9.2 Media Tidak Terpandu (*Unguided Media*)

Jenis komunikasi pada media ini bersifat *wireless*. *Wireless* atau disebut nirkabel adalah teknologi yang menghubungkan dua perangkat untuk bertukar data tanpa media kabel. Data dipertukarkan melalui media gelombang cahaya tertentu atau gelombang radio dengan frekuensi tertentu.

1. Gelombang Radio

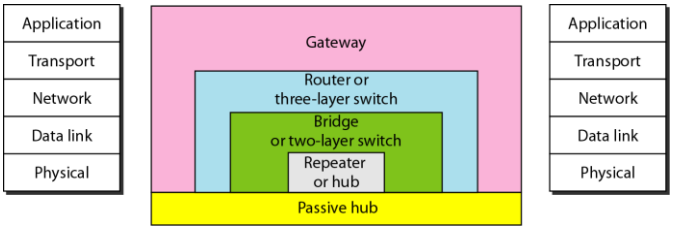
   Gelombang radio adalah satu bentuk dari radiasi elektromagnetik. Frekuensinya antara 3kHz sampai 1Ghz.

2. Gelombang Mikro

   Gelombang mikro (*microwave*) adalah gelombang elektromagnetik dengan panjang gelombang antar 1mm hingga 1m. Atau dengan kata lain memiliki frekuensi di antara 1GHz hingga 300GHz.

2.10 Perangkat Penghubung

Perangkat penghubung dibagi ke dalam lima kategori berbeda berdasarkan lapisan yang digunakan dalam sebuah jaringan. Berikut ini merupakan gambar dari lima kategori perangkat penghubung [1]:



Gambar 2.27 Lima Kategori Perangkat Penghubung

Penjelasan dari gambar 2.27:

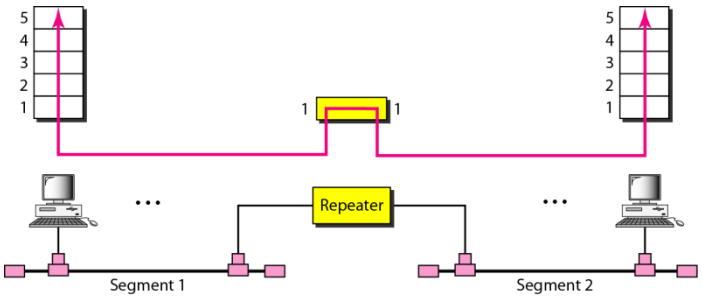
1. Perangkat yang beroperasi di bawah lapisan fisik (hub pasif).
2. Perangkat yang beroperasi pada lapisan fisik (*repeater* atau hub aktif).
3. Perangkat yang beroperasi pada lapisan fisik dan lapisan jalur data (bridge atau switch layer 2).
4. Perangkat yang beroperasi pada lapisan fisik, lapisan jalur data, dan lapisan jaringan (router atau switch layer 3).
5. Perangkat yang dapat beroperasi pada lima lapisan (*gateway*).

2.10.1 Hub Pasif

Hub pasif hanyalah sebuah konektor yang menghubungkan kabel dari cabang yang berbeda. Dalam sebuah topologi star Ethernet LAN, sebuah hub pasif adalah tempat di mana sinyal datang dari beberapa *station* yang bertemu, hub pasif adalah tempat pertemuannya. Tipe dari hub ini adalah bagian dari media yang yang berada di bawah lapisan fisik pada model Internet (TCP/IP).

2.10.2 *Repeater*

Repeater adalah perangkat yang beroperasi hanya pada lapisan fisik. Sinyal yang membawa informasi dalam sebuah jaringan dapat melemah yang mengakibatkan keutuhan data berubah. Sebuah *repeater* menerima sinyal sebelum sinyal tersebut melemah atau rusak, meregenerasikan bentuk asli bit dari suatu data. *Repeater* kemudian mengirimkan sinyal yang telah diperbaharui. Sebuah *repeater* dapat memperpanjang jalur fisik dari sebuah LAN. Berikut ini merupakan gambar dari salah satu fungsi *repeater*:

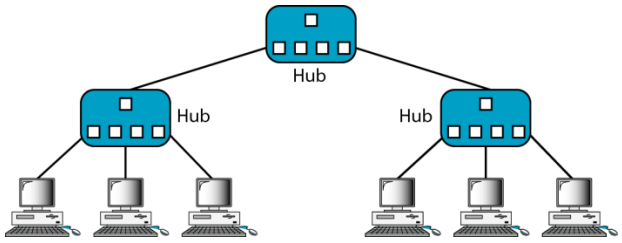


Gambar 2.28 Sebuah *Repeater* yang Menghubungkan Dua Segmen LAN

Sebuah *repeater* sebenarnya tidak menghubungkan dua LAN, *repeater* hanya menghubungkan dua segmen dari LAN yang sama dan masih merupakan satu bagian dari suatu LAN. Sebuah *repeater* bukanlah sebuah perangkat yang menghubungkan dua LAN dari protokol yang berbeda.

2.10.3 Hub Aktif

Hub aktif sebenarnya adalah *repeater* yang memiliki banyak *port*. Normalnya digunakan untuk menghubungkan antar *station* pada sebuah topologi star. Bagaimanapun juga, hub dapat digunakan untuk membuat hirarki yang memiliki banyak level. Ditunjukkan pada gambar 2.29.

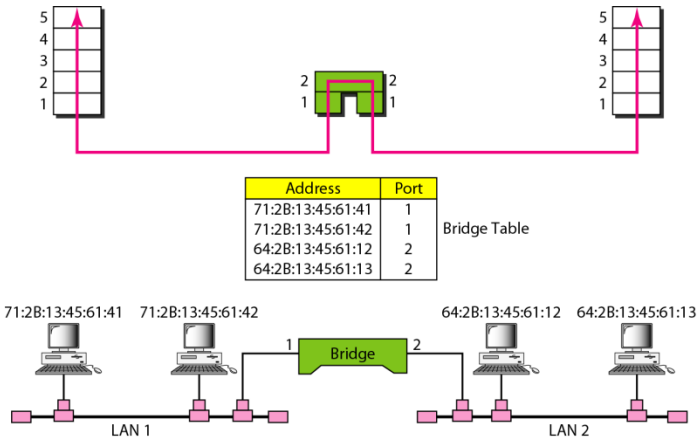


Gambar 2.29 Hirarki Hub

2.10.4 Bridge

Bridge beroperasi pada lapisan fisik dan lapisan jalur data. Pada lapisan fisik bridge meregenerasikan sinyal yang diterimanya. Pada lapisan jalur data, bridge dapat memeriksa alamat fisik (MAC *address*) sumber dan tujuan yang terkandung dalam sebuah *frame*. Sebuah bridge memiliki sebuah tabel yang digunakan untuk menyaring keputusan, sebagai contoh kasus, dua LAN terhubung dengan sebuah bridge. Jika sebuah *frame* ditujukan untuk station 712B13456142

tiba pada *port* 1, bridge memeriksa tabelnya untuk menemukan *port* pemberangkatannya. Berdasarkan informasi yang berada pada tabelnya, *frame* untuk 712B13456142 pergi melalui *port* 1, oleh karena itu *frame* tidak perlu diteruskan. Pada kasus lain, jika sebuah *frame* untuk 712B13456141 tiba pada *port* 2, *port* pemberangkatan adalah *port* 1 sehingga *frame* akan diteruskan. Pada kasus pertama, LAN 2 masih bebas dari lalu lintas, pada kasus kedua, pada kedua LAN terdapat lalu lintas data. Ilustrasi dari contoh kasus di atas ditunjukkan pada gambar 2.30.

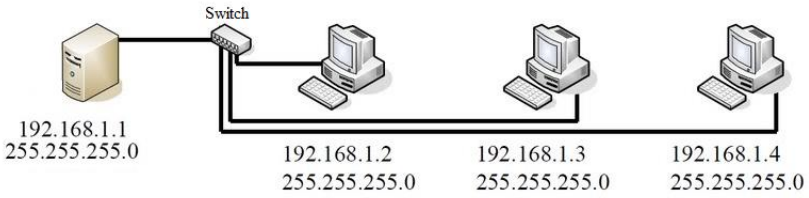


Gambar 2.30 Sebuah Bridge yang Menghubungkan Dua LAN

2.10.5 Switch Layer 2

Switch layer 2 beroperasi pada lapisan fisik dan lapisan jalur data. Switch layer 2 adalah sebuah bridge yang memiliki banyak *port* dan didesain untuk lebih baik (lebih cepat) dalam hal performansinya. Sebuah bridge dengan beberapa *port* dapat berhubungan dengan beberapa LAN bersama-sama. Sebuah bridge dengan banyak *port* dapat mengalokasikan sebuah *port* unik pada setiap *station* yang memiliki *entity* yang independen. Dalam hal ini berarti tidak ada tabrakan data.

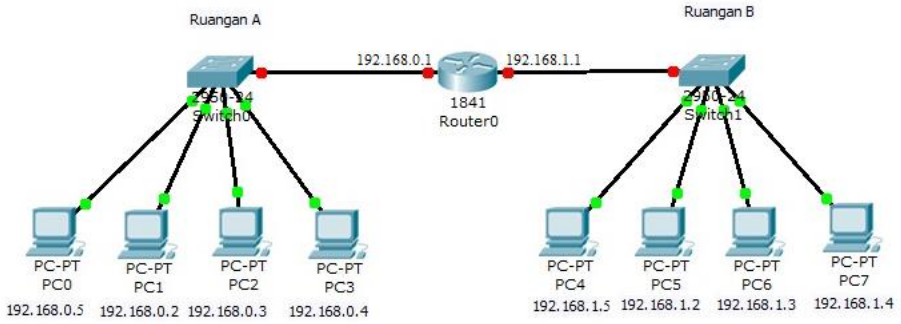
Switch layer 2 yang bekerja sebagai bridge menyaring tujuan berdasarkan MAC *address* dari *frame* yang diterimanya. Switch layer 2 dapat lebih canggih lagi, karena memiliki sebuah faktor *switching* untuk meneruskan *frame* dengan lebih cepat. Beberapa switch layer 2 yang baru disebut *cut-through* switch, yang didesain untuk meneruskan *frame* sesegera mungkin seperti ketika memeriksa MAC *address header* pada *frame*. Di bawah ini adalah gambar switch pada LAN:



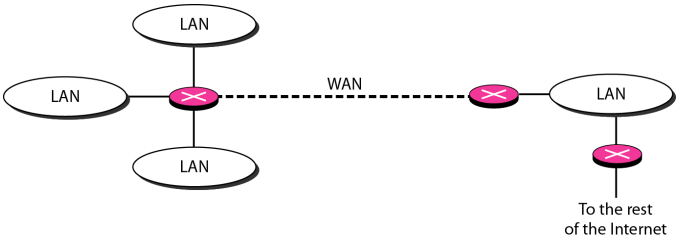
Gambar 2.31 Sebuah Switch pada LAN

2.10.6 Router

Router merupakan sebuah perangkat layer 3 yang memetakan paket berdasarkan pada alamat-alamat logik (pengalamatan antar *host*). Sebuah router dapat menghubungkan dua segmen jaringan yang berbeda dalam LAN, ditunjukkan pada gambar 2.32. Selain itu router juga dapat menghubungkan LAN dengan WAN, ditunjukkan pada gambar 2.33. Router memiliki tabel *routing* yang digunakan untuk menentukan tujuan data yang akan dikirimkan. Tabel *routing* yang digunakan normalnya bersifat dinamis dan diperbaharui menggunakan protokol *routing*.



Gambar 2.32 Sebuah Router yang Menghubungkan Dua Segmen Jaringan yang Berbeda dalam LAN



Gambar 2.33 Router yang Menghubungkan LAN dengan WAN

2.10.7 Switch Layer 3

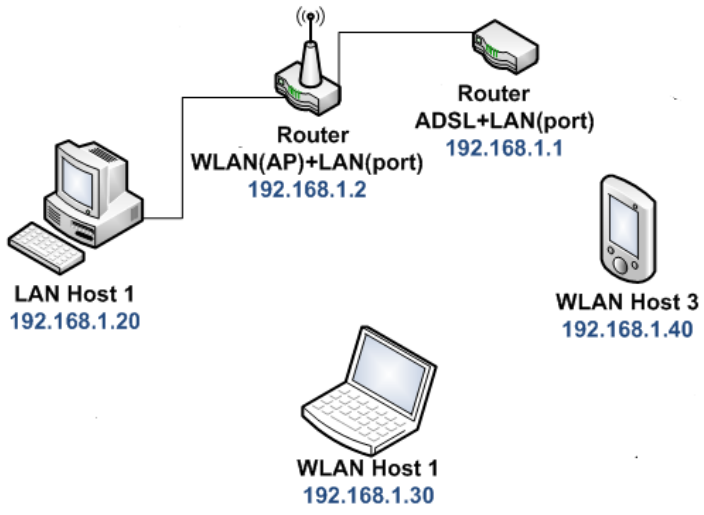
Switch layer 3 merupakan sebuah router, tetapi lebih cepat dan lebih canggih. Struktur *switching* pada switch layer 3 mengizinkan *lookup* tabel yang lebih cepat.

2.10.8 *Gateway*

Sebuah *gateway* normalnya adalah sebuah komputer yang beroperasi pada lima lapisan pada Internet atau tujuh lapisan pada model OSI. Sebuah *gateway* dapat digunakan sebagai perangkat penghubung antar dua *internetworks* yang menggunakan model berbeda. Sebagai contoh, sebuah jaringan yang didesain menggunakan model OSI dapat terhubung dengan jaringan lain yang menggunakan model Internet.

2.11 *Access Point* (AP)

Access point adalah perangkat seperti router *wireless* yang memungkinkan perangkat *wireless* untuk terhubung ke jaringan. *Access Point* berfungsi sebagai hub/switch yang menghubungkan jaringan lokal dengan jaringan internet. Pada *access point* koneksi data atau dikirim melalui gelombang radio [4]. Di bawah ini merupakan gambar dari *access point* yang digunakan pada LAN:



Gambar 2.34 Sebuah *Access Point* pada LAN

2.12 *Network Management System* (NMS)

Network Management System adalah sistem yang digunakan untuk mengatur dan memonitor jaringan. Yang dimonitor bisa berupa perangkat jaringan ataupun *server*. NMS telah sangat luas digunakan untuk mempermudah pekerjaan administrator dalam mengatur dan mengawasi keadaan sistem dan jaringan. Banyak keuntungan dari NMS, salah satu keuntungan penerapan NMS adalah mengawasi dan mencegah sesuatu yang tidak diharapkan. NMS dapat memberitahukan sedini mungkin (*early detection*) masalah yang terjadi dalam suatu jaringan [5].

2.13 *Network Mapper* (Nmap)

Nmap merupakan perangkat lunak *open source* untuk eksplorasi dan audit keamanan jaringan. Dirancang untuk memeriksa jaringan besar secara cepat. Nmap menggunakan paket IP *raw* dalam menentukan *host* mana saja yang tersedia pada jaringan. Namun banyak administrator sistem dan jaringan menganggapnya berguna untuk tugas rutin seperti inventori jaringan, mengelola jadwal *upgrade* layanan dan melakukan monitoring *uptime host* atau layanan [6].

2.14 *Apache Web Server*

Web Server adalah sebuah perangkat lunak *server* yang berfungsi menerima permintaan *HyperText Transfer Protocol* (HTTP) atau *HyperText Transfer Protocol Secure* (HTTPS) dari *client* yang dikenal dengan *web browser* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen HTML. Salah satu *web server* yang terkenal di Linux adalah Apache. Apache merupakan *web server* antar *platform* yang dapat berjalan di beberapa *platform* seperti Linux dan Windows. Didesain dengan PHP, yaitu bahasa pemrograman yang digunakan untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML [7].

2.15 Calstats

Calstats adalah seperangkat skrip sederhana yang digunakan untuk melakukan monitoring suatu kondisi perangkat jaringan komputer. Calstats menciptakan struktur jaringan dan menandai perangkat atau komputer yang sedang aktif atau tidak aktif [8]. Calstats tidak dapat berdiri sendiri, karena membutuhkan perangkat lunak lain seperti Nmap, Crontab, dan Apache dalam mengoperasikannya. Tampilan Calstats merupakan antarmuka sistem monitoring jaringan yang berbentuk gambar dalam format *Portable Network Graphics* (PNG) yang akan ditampilkan dalam *web browser*. Dalam Calstats terdapat file yang bernama *gen_stats*. File tersebut merupakan algoritma yang digunakan untuk dapat mengenali kondisi perangkat jaringan.

2.16 *Internet Control Message Protocol* (ICMP)

ICMP adalah sebuah protokol yang didesain untuk menangani pesan kesalahan yang tidak mampu dilaporkan oleh *Internet Protocol* (IP) jika terjadi masalah pada sebuah jaringan [1]. Pesan atau paket ICMP dikirim jika terjadi masalah pada lapisan IP dan lapisan diatasnya (TCP dan UDP). Beberapa karakteristik dari ICMP adalah:

1. ICMP merupakan bagian internal dari IP dan diimplementasikan di setiap modul IP.
2. ICMP digunakan untuk menyediakan *feedback* tentang beberapa kesalahan pada sebuah proses datagram.
3. Tidak mendukung kehandalan pengiriman paket IP. Datagram/paket bisa tidak terkirim dan tidak ada laporan pemberitahuan tentang kehilangan datagram. Jika diperlukan adanya keandalan maka harus diimplementasikan pada lapisan transport (pada model TCP/IP).

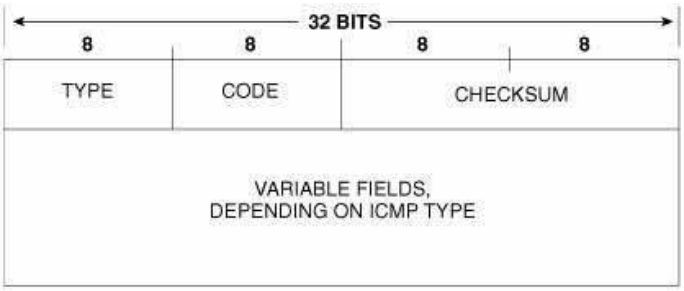
2.16.1 Format Pesan ICMP

Sebuah pesan ICMP memiliki 3 *field*, yaitu:

1. 8 bit : *field TYPE* yang mengidentifikasikan pesan.
2. 8 bit : *field CODE* yang menyediakan informasi tentang jenis pesan.

3. 16 bit : *field CHECKSUM* untuk pemerikasaan pesan ICMP.

Format umum dari pesan ICMP adalah sebagai berikut:



Gambar 2.35 Format Umum Pesan ICMP

Sedangkan untuk jenis – jenis pesan ICMP terdapat pada gambar 2.36.

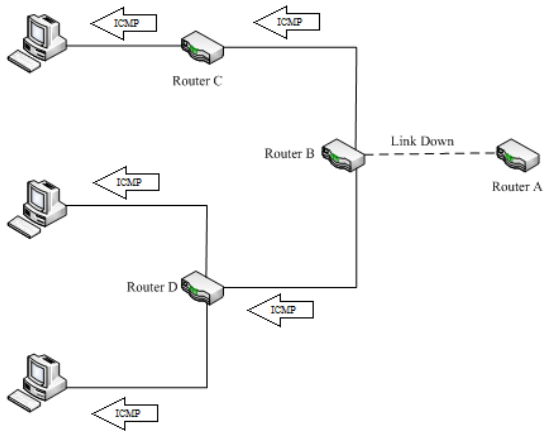
ICMP Message Types	
0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirect/ Change Request
8	Echo Request
9	Router Advertisement
10	Router Selection
11	Time Exceeded
12	Parameter Problem
13	Timestamp Request
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address Mask Request
18	Address Mask Reply

Gambar 2.36 Jenis - Jenis Pesan ICMP

2.16.2 ICMP *Error Messages*

Setiap ICMP *error messages* terdiri dari IP *header* dan sekurangnya 8-byte pada datagram. Pada *field type* menspesifikasikan tipe dari pesan, sementara kode kesalahan pada datagram dilaporkan oleh pesan ICMP pada *field code*. Pada kondisi normal IP berjalan baik dan menghasilkan proses penggunaan memori serta sumber daya transmisi yang efisien. Namun ada beberapa kondisi di mana koneksi IP terganggu, misalnya karena router yang *crash*, putusnya kabel atau matinya *host* tujuan. Pada saat ini ICMP berperan membantu menstabilkan kondisi jaringan. Hal ini dilakukan dengan cara memberikan pesan-pesan tertentu,

sebagai respon atas kondisi tertentu yang terjadi pada jaringan tersebut. Di bawah ini merupakan cara kerja ICMP saat terjadi masalah dalam suatu jaringan.



Gambar 2.37 Masalah yang Terjadi pada Suatu Jaringan

Pada gambar 2.37 di atas hubungan antar router A dan B mengalami masalah, sehingga router B akan secara otomatis mengirimkan paket ICMP *Destination Unreachable* ke *host* pengirim paket yang berusaha melewati *host* A menuju tujuannya. Dengan adanya pemberitahuan ini maka *host* tujuan tidak akan terus menerus berusaha mengirimkan paketnya melewati router A.

Terdapat dua jenis pesan yang dapat dihasilkan oleh ICMP, yaitu ICMP *Error Message* dan ICMP *Query Message*. ICMP *Error Message* dihasilkan jika terjadi kesalahan pada jaringan. Sedangkan ICMP *Query Message* ialah jenis pesan yang dihasilkan oleh ICMP jika pengirim paket menginginkan informasi tertentu yang berkaitan dengan kondisi jaringan. ICMP *Error Message* dibagi menjadi beberapa jenis diantaranya:

1. *Destination Unreachable*. Pesan ini dihasilkan oleh router jika pengiriman paket mengalami kegagalan akibat masalah putusnya jalur, baik secara fisik maupun secara logik. *Destination Unreachable* ini dibagi menjadi beberapa tipe. Beberapa tipe yang penting adalah:
    1. *Network Unreachable*, jika jaringan tujuan tak dapat dihubungi.
    2. *Host Unreachable*, jika host tujuan tak bisa dihubungi.
    3. *Protocol at Destination is Unreachable*, jika ditujuan tak tersedia protokol tersebut.

4. *Port is Unreachable*, jika tidak ada *port* yang dimaksud pada tujuan.
5. *Destination Network is Unknown*, jika jaringan tujuan tak diketahui.
6. *Destination Host is Unknown*, jika *host* tujuan tidak diketahui.

2. *Time Exceeded*. Paket ICMP jenis ini dikirimkan jika isi *field* TTL dalam paket IP sudah habis dan paket belum juga sampai ke tujuannya. Setiap kali sebuah paket IP melewati satu router, nilai TTL dalam paket tersebut dikurangi satu. TTL ini diterapkan untuk mencegah timbulnya paket IP yang terus menerus berputar di jaringan karena suatu kesalahan tertentu, sehingga menghabiskan sumber daya jaringan yang ada.
3. *Parameter Problem*. Paket ini dikirimkan jika terdapat kesalahan parameter pada *header* paket IP.
4. *Source Quench*. paket ICMP ini dikirimkan jika router atau tujuan mengalami kongesti. Sebagai respon pada paket ini, pihak pengirim paket harus memperlambat pengiriman paketnya.
5. *Redirect*. Paket ini dikirimkan jika router merasa *host* mengirimkan paket IP melalui router yang salah. Paket ini seharusnya dikirimkan melalui router lain.

2.16.3 ICMP *Query Messages*

ICMP *Query Messages* terdiri dari:

1. *Echo Request dan Echo Reply*. Bertujuan untuk memeriksa apakah sistem tujuan dalam keadaan aktif. Program ping merupakan program pengiriman paket ini. Sistem tujuan harus mengembalikan data yang sama dengan data yang dikirimkan.
2. *Timestamp dan Timestamp Reply*. Menghasilkan informasi waktu yang diperlukan sistem tujuan untuk memproses suatu paket.
3. *Address Mask*. Untuk mengetahui berapa *netmask* yang harus digunakan oleh suatu *host* dalam suatu jaringan.

2.17 Ping

Ping (*Packet Internet Groper*) adalah sebuah program *utilitas* yang digunakan untuk memeriksa konektivitas jaringan berbasis teknologi TCP/IP [9].

Berfungsi untuk memeriksa apakah *host* tujuan dalam keadaan aktif atau tidak dengan mengirimkan paket ICMP (*echo request* dan *echo reply*).

Ping bekerja dengan cara mengirimkan ICMP *echo request* ke *host* tujuan dan menunggu ICMP *echo reply* dari *host* tujuan. ICMP bekerja pada layer *network* (internet) berdampingan dengan IP. ICMP bergantung pada IP sebagai transportasi untuk melintas dalam jaringan. Program Ping dijalankan pada layer aplikasi. Program Ping memanfaatkan *raw socket* untuk dapat melewati layer transport. *Socket* merupakan fungsi untuk mengalokasikan *resource* sehingga program bisa berinteraksi dengan protokol TCP/IP di bawahnya. Ada tiga jenis socket, yaitu *stream*, *datagram*, dan *raw*. TCP menggunakan *stream* dan UDP menggunakan *datagram*. *Raw socket* digunakan oleh semua aplikasi yang ingin berinteraksi langsung dengan IP, tapi melewati TCP dan UDP, termasuk program Ping.

Pada waktu program Ping memulai eksekusi, *raw socket* terbuka hanya untuk ICMP. Pada *output*, pengiriman ICMP *echo request*, program harus memformat pesan ICMP. Sistem akan memberikan IP *header* dan *Ethernet header*. IP *address* sumber dan IP *address* tujuan terdapat pada IP *header*. Pada *input*, program harus memeriksa semua pesan ICMP yang datang. Format pesan pada *echo request*, *field type* bernilai 8 dan *field code* bernilai 0 (gambar 2.38). Sedangkan format pesan pada *echo reply*, *field type* dan *field code* bernilai 0 (gambar 2.39). *Checksum* digunakan untuk mendeteksi *error*. *Field identification* yang berisi ID proses (PID) yang menunjukkan eksekusi program Ping. Selanjutnya adalah *field sequence*, yang dimulai dari nol dan naik satu setiap *echo request* dikirim. Kemudian terdapat *field* opsional bernama *test data*. Ini meliputi waktu yang digunakan dalam kalkulasi waktu pulang-pergi saat menerima *echo reply*.

Pada waktu pesan ICMP datang, tugas ping sedikit lebih kompleks. Karena *ping* menggunakan *raw socket*, program menerima semua pesan ICMP yang datang. Ini berarti *ping* tidak hanya melihat *echo reply* tapi juga yang lain, seperti pesan *Destination Unreachable Source Quench* dan *Time Exceeded*. Jika ada dua program Ping yang dijalankan pada waktu bersamaan, maka masing-masing

program harus mengidentifikasi pesan yang ditujukan untuk dirinya. Itu bisa diketahui dengan melihat ID proses pada *Field Identification*. Gambar 2.38 dan gambar 2.39 merupakan gambar format pesan *echo request* dan *echo reply*.

00 01 02 03 04 05 06 07	08 09 10 11 12 13 14 15	16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Type = 8	Code = 0	Header Checksum
Identifier		Sequence Number
Time Stamp		Data
Data		

Gambar 2.38 *Echo Request*

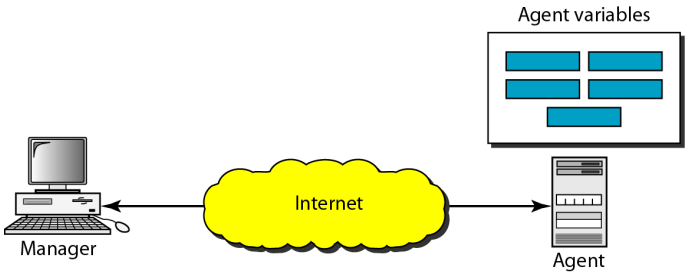
00 01 02 03 04 05 06 07	08 09 10 11 12 13 14 15	16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Type = 0	Code = 0	Header Checksum
Identifier		Sequence Number
Time Stamp		Data
Data		

Gambar 2.39 *Echo Reply*

2.18 *Simple Network Management Protocol* (SNMP)

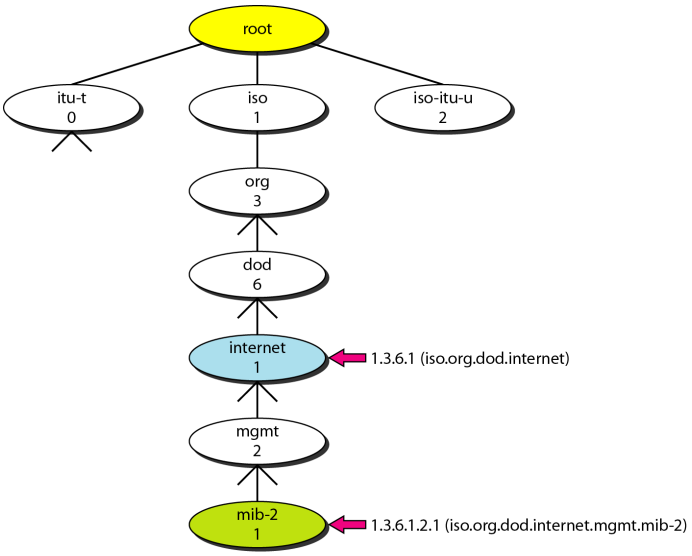
SNMP merupakan protokol jaringan yang berfungsi untuk memeriksa dan mengontrol perangkat jaringan/*server*. Memeriksa diartikan dapat memonitor perangkat/*server*. Mengontrol di sini dimaksudkan bisa melakukan pengaturan parameter dalam rangka mengontrol perangkat [5].

SNMP menggunakan konsep *manager* dan *agent*. *Manager* biasanya adalah sebuah *host*, mengontrol dan memonitor seperangkat *agent*. *Agent* biasanya adalah router/*host* [1].



Gambar 2.40 Konsep SNMP

Dalam SNMP terdapat *Object Identifier*, lebih dikenal dengan singkatan OID. OID adalah pengindetifikasian suatu objek dengan menggunakan angka (*string numerik*). Suatu objek haruslah diwakilkan dengan kombinasi numerik yang berbeda dengan objek lainnya. Demikian juga dengan perangkat jaringan/*server*. Visualisasi percabangan dari OID hampir sama dengan pengertian suatu direktori mempunyai sub direktori lain di bawahnya. Berikut adalah gambarnya:



Gambar 2.41 *Object Identifier*

Struktur pohon di atas dimulai dengan *root*. Setiap objek didefinisikan dengan menggunakan urutan integer yang dipisahkan oleh titik. Selain itu dapat didefinisikan juga dengan menggunakan urutan nama yang dipisahkan oleh titik. Representasi *integer-dot* digunakan pada SNMP. Sedangkan *name-dot* digunakan oleh orang. Sebagai contoh, berikut ini merupakan objek yang sama dalam dua notasi yang berbeda [1].

iso.org.dod.internet.mgmt.mib-2 →1.3.6.1.2.1

Objek yang digunakan dalam SNMP yang berada di bawah objek *mib-2* memiliki identitas yang diawali dengan 1.3.6.1.2.1.

Dalam SNMP terdapat MIB (*Management Information Base*), yaitu database yang berisi deskripsi dari kumpulan objek suatu perangkat jaringan. Kadangkala *developer*/vendor memberi MIB dalam format *text*/*name-dot* [5].

2.19 Parameter QoS

Quality of Service (QoS) merupakan istilah umum untuk menyatakan efek dari kinerja layanan secara keseluruhan pada jaringan dari sudut pandang *user* [10]. Terdapat dua parameter yang dapat menentukan QoS pada sistem monitoring jaringan dengan menggunakan ICMP, yaitu *packet loss* dan *delay/latency*.

2.19.1 *Packet Loss*

Definisi *packet loss* adalah banyaknya paket yang hilang selama proses transmisi dari sumber ke tujuan. Nilai *packet loss* selama proses transmisi dari sumber ke tujuan dalah sebagai berikut [10]:

$$Packet\ Loss = \frac{(Packets\ transmitted - Packets\ received)}{Packets\ transmitted} \times 100\%$$

Secara umum terdapat empat kategori penurunan performansi jaringan berdasarkan nilai *packet loss* sesuai dengan versi TIPHON-*Telecommunications and Internet Protocol Harmonization Over Networks* (Joesman 2008), yaitu seperti pada tabel 2.4 [11] :

Tabel 2.4 Performansi Jaringan IP Berdasarkan *Packet Loss*

Kategori Degredasi	***Packet Loss***
Sangat Bagus	0%
Bagus	3%
Sedang	15%
Jelek	25%

2.19.2 *Delay*

Latency atau *delay* mendefinisikan berapa lama seluruh pesan dapat terkirim dengan utuh sampai di tujuan sejak bit pertama dikirimkan dari sumber [1].

Menurut versi TIPHON (Joesman 2008), besarnya *delay* dapat diklasifikasikan seperti yang terdapat pada tabel berikut [10]:

Tabel 2.5 Performansi Jaringan IP Berdasarkan *Delay*/*Latency*

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

2.20 Cacti dan Perangkat Lunak Pendukung

Cacti adalah salah satu aplikasi sistem monitoring jaringan berbasis web yang menggunakan RRDTool untuk menampilkan grafik dan Net-SNMP dalam pengambilan data, Cacti menyimpan sebuah informasi penting untuk membuat grafik dan mengumpulkan data tersebut ke dalam sebuah *database* MySQL [5]. Terdapat beberapa perangkat lunak yang digunakan untuk dapat menjalankan Cacti. Berikut ini merupakan penjelasan dari masing-masing perangkat lunak yang digunakan dalam Cacti:

1. RRDTool

   RRDTool (*Round Robin Database Tool*) adalah sistem yang digunakan untuk menyimpan dan menampilkan data *time-series* (misalnya *bandwidth* jaringan, suhu suatu perangkat, beban *server* rata-rata, dan lain-lain). RRDTool mampu menyimpan data yang disajikan dalam bentuk grafik.
2. Net-SNMP

   Net-SNMP adalah aplikasi yang digunakan untuk mengimplementasikan SNMP v1, SNMP v2c dan SNMP v3 menggunakan IPv4 dan IPv6.
3. MySQL

   MySQL adalah suatu program yang digunakan untuk membuat sebuah *database* yang bersifat *open source*.