BAB II
LANDASAN TEORI

2.1 Jaringan Komputer

Jaringan komputer adalah sekelompok komputer otonom yang dihubungkan satu dengan yang lainnya dengan menggunakan protokol komunikasi melalui media transmisi atau media komunikasi sehingga dapat saling berbagi data informasi, program–program, *share* perangkat keras seperti printer atau harddisk. Tujuan dibangunnya suatu jaringan komputer adalah untuk membawa data informasi dari sisi pengirim menuju penerima secara cepat dan tepat tanpa adanya kesalahan melalui media transmisi atau media komunikasi tertentu.

2.1.1 Jenis Jaringan Komputer

Jaringan komputer yang digunakan dewasa ini dapat dikelompokkan berdasarkan luasan area yang dapat dijangkau. Luasan area ini pada mulanya ditentukan berdasarkan jarak jangkau dalam satuan meter. Namun dalam perkembangan selanjutnya berdasarkan peralatan dan fasilitas. Penggunaan peralatan–peralatan yang dimaksud adalah peralatan–peralatan *inter* jaringan seperti *Repeater, Router* atau *Gateway* dari peralatan tersebut dapat dilihat jangkauan area jaringan dan luasan segmen jaringan yang dibangun.

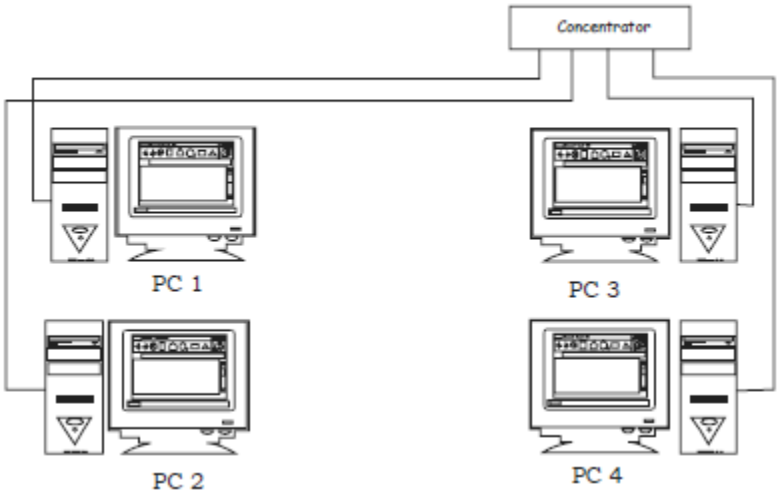
2.1.1.1 *Local Area Network* (LAN)

Local Area Network (LAN) merupakan jaringan milik pribadi di dalam sebuah gedung atau kampus yang berukuran sampai beberapa kilometer.

LAN seringkali digunakan untuk menghubungkan komputer-komputer pribadi dan workstation dalam kantor perusahaan atau pabrik-pabrik untuk memakai bersama resource (misalnya, printer, scanner) dan saling bertukar informasi. LAN dapat dibedakan dari jenis jaringan lainnya berdasarkan tiga karakteristik: ukuran, teknologi transmisi dan topologinya.

LAN mempunyai ukuran yang terbatas, yang berarti bahwa waktu transmisi pada keadaan terburuknya terbatas dan dapat diketahui sebelumnya. Dengan mengetahui keterbatasannya, menyebabkan adanya kemungkinan untuk menggunakan jenis desain tertentu. Hal ini juga memudahkan manajemen jaringan.

LAN seringkali menggunakan teknologi transmisi kabel tunggal. LAN tradisional beroperasi pada kecepatan mulai 10 sampai 100 Mbps (mega bit/detik) dengan delay rendah (puluhan mikro *second*) dan mempunyai faktor kesalahan yang kecil. LAN-LAN modern dapat beroperasi pada kecepatan yang lebih tinggi, sampai ribuan megabit/detik



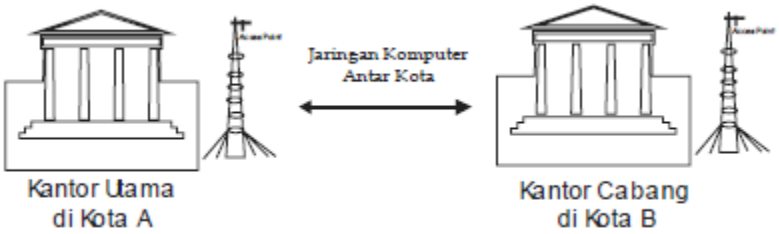
Gambar 2.1 *Local Area Network*

2.1.1.2 *Metropolitan Area Network* (MAN)

Metropolitan Area Network (MAN) pada dasarnya merupakan versi LAN yang berukuran lebih besar dan biasanya memakai teknologi yang sama dengan LAN. MAN dapat mencakup kantor-kantor perusahaan yang berdekatan dan dapat dimanfaatkan untuk keperluan pribadi (swasta) atau umum. MAN biasanya mampu menunjang data dan suara, dan bahkan dapat berhubungan dengan jaringan televisi kabel. MAN hanya memiliki sebuah atau dua buah kabel dan tidak mempunyai elemen switching, yang berfungsi untuk mengatur paket melalui beberapa output kabel. Adanya elemen switching membuat rancangan menjadi lebih sederhana.

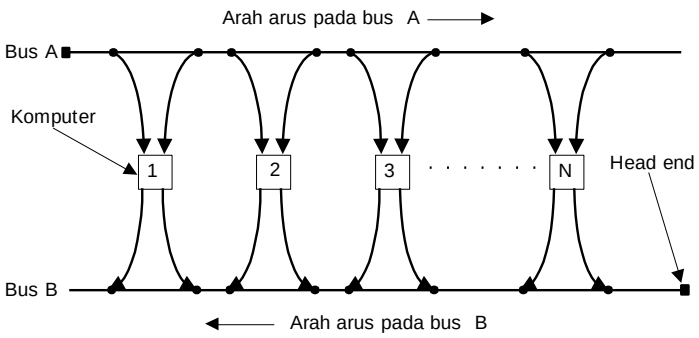
Sebuah MAN, biasanya meliputi area yang lebih besar dari LAN, misalnya antar wilayah dalam satu provinsi. Dalam hal ini jaringan menghubungkan beberapa buah jaringan-jaringan kecil ke dalam lingkungan area yang lebih besar,

sebagai contoh yaitu : jaringan Bank dimana beberapa kantor cabang sebuah Bank di dalam sebuah kota besar dihubungkan antara satu dengan lainnya.



Gambar 2.2 *Metropolitan Area Network*

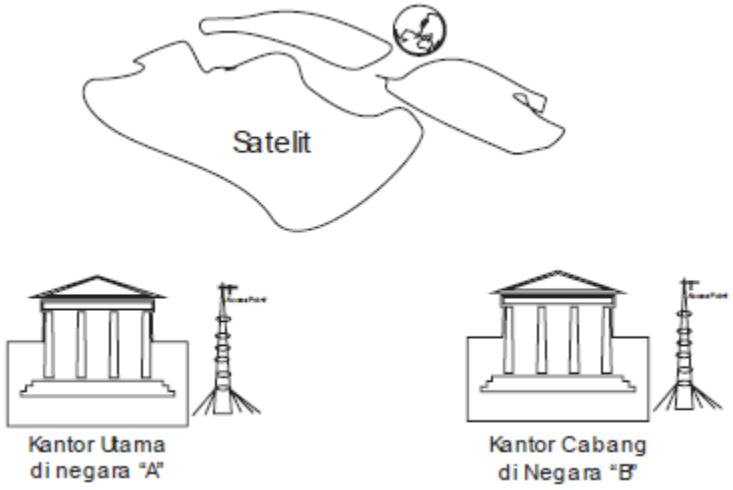
Alasan utama memisahkan MAN sebagai kategori khusus adalah telah ditentukannya standar untuk MAN, dan standar ini sekarang sedang diimplementasikan. Standart tersebut disebut DQDB (*Distributed Queue Dual Bus*) atau 802.6 menurut standart IEEE. DQDB terdiri dari dua buah kabel *unidirectional* dimana semua komputer dihubungkan, seperti ditunjukkan pada gambar 2.3. Setiap bus mempunyai sebuah *head–end*, perangkat untuk memulai aktivitas transmisi. Lalulintas yang menuju komputer yang berada di sebelah kanan pengirim menggunakan bus bagian atas. Lalulintas ke arah kiri menggunakan bus yang berada di bawah.



Gambar 2.3 *Arsitektur MAN DQDB*

2.1.1.3 *Wide Area Network* (WAN)

Wide Area Networks (WAN) adalah jaringan yang lingkupnya biasanya sudah menggunakan sarana satelit ataupun kabel bawah laut sebagai contoh keseluruhan jaringan suatu bank yang ada di Indonesia ataupun yang ada di negara-negara lain dengan menggunakan sarana WAN, Sebuah bank yang ada di Bandung bisa menghubungi kantor cabangnya yang ada di Jepang, hanya dalam beberapa menit. Biasanya WAN lebih rumit dan sangat kompleks apabila dibandingkan dengan LAN maupun MAN, serta menggunakan banyak sarana untuk menghubungkan antara LAN dan WAN ke dalam komunikasi global seperti internet.



Gambar 2.4 *Wide Area Network*

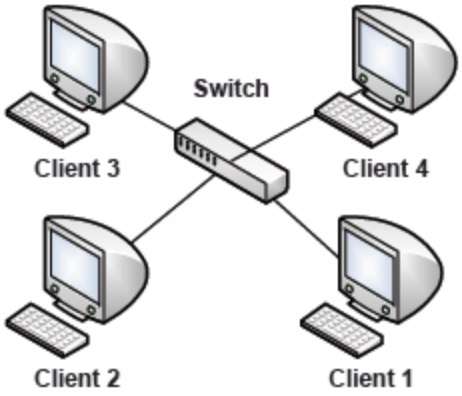
2.1.1.4 Internet

Internet merupakan jaringan Global Internasional di seluruh dunia. Biasanya teknologi ini menggunakan suatu fasilitas layanan yang biasa kita sebut *World Wide Web* (WWW).

2.1.2 Tipe Jaringan

2.1.2.1 Model *Peer to Peer*

Pada model jaringan terdapat dua atau lebih komputer yang saling terhubung. Masing-masing komputer mempunyai kedudukan yang setara atau setingkat sehingga komputer manapun dapat memulai (*initiate*) dan mengendalikan hubungan dalam jaringan. Model Jaringan ini merupakan jenis yang paling sederhana.



Gambar 2.5 *Model Jaringan Peer to Peer*

2.1.2.2 Model *Client-Server*

Konsep Jaringan *client-server* ini membedakan dengan jelas kedudukan suatu komputer, mana yang dapat memberikan layanan jaringan (*server*) dan mana yang hanya meminta layanan (*client*). Jika suatu komputer di*install* sebagai *server* yang memberikan segala sumber daya (*resource*), contohnya : printer, modem kepada komputer lain (*client)* yang terhubung ke jaringan. Untuk dapat saling berkomunikasi antara *server* dan *client*, maka digunakan suatu aplikasi jaringan yang dinamakan *client-server program*, dimana *server* menggunakan

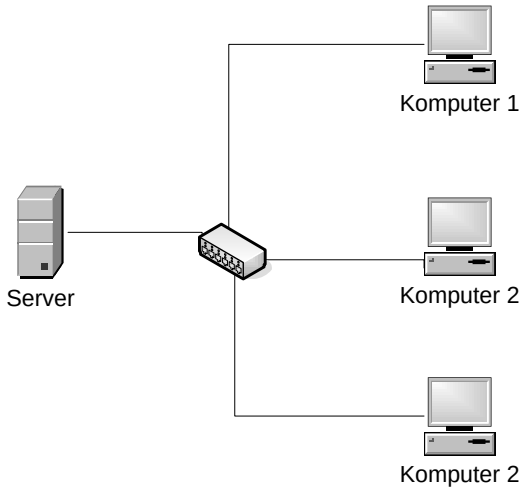
aplikasi jaringan yang disebut *server program*, sementara *client* menggunakan *client program* untuk dapat berkomunikasi dengan *server program* pada *server*.

Beberapa sifat dari *client-server* diantaranya sebagai berikut :

1. *Server* dan *Client* berada pada posisi serta proses yang berbeda.
2. *Server* dan *Client* dapat dijalankan pada mesin yang sama atau berbeda.
3. Setiap proses dapat menyembunyikan informasi tersendiri.
4. Komunikasi yang dilakukan bisa bersifat protokol *peer to peer*.

Koneksitas *client ke server :*

1. *Server* harus terlebih dahulu dijalankan (*running*).
2. *Server* harus menentukan *port* untuk koneksi yang masuk.
3. *Client* harus mengetahui nama *host* dari *server* beserta *port* yang digunakan untuk koneksi.
4. *Client* mengirim pesan ke *server* untuk melakukan suatu pekerjaan (*task*).



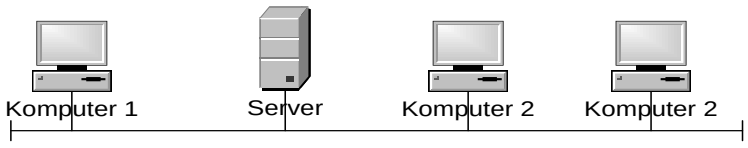
Gambar 2.6 *Model Jaringan Client Server*

2.1.3 Topologi Jaringan

Topologi jaringan merupakan suatu struktur atau bagaimana menghubungkan antara simpul dan pusat dalam suatu jaringan. Tiap struktur mempunyai keuntungan dan kerugiannya masing-masing. Topologi yang umumnya sebagai berikut :

2.1.3.1 Topologi Linear Bus

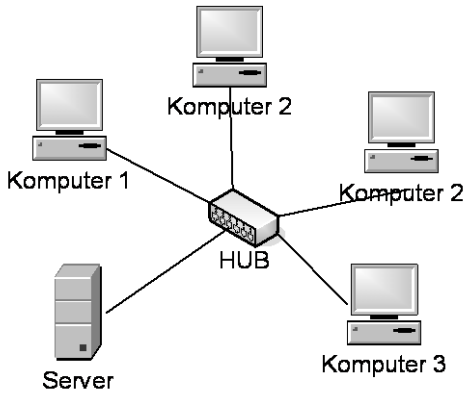
Topologi sebuah Linear BUS / garis lurus terdiri dari satu jalur kabel utama (*backbone*) dimana pada masing-masing ujungnya diberikan sebuah *terminator*. Semua *nodes* pada jaringan (*file server, workstation,* dan perangkat lainnya) terkoneksi ke sebuah kabel utama. Jaringan-jaringan *Ethernet* dan *Localtalk* menggunakan topologi *linear* ini.



Gambar 2.7 *Topologi Linear BUS*

2.1.3.2 Topologi Star

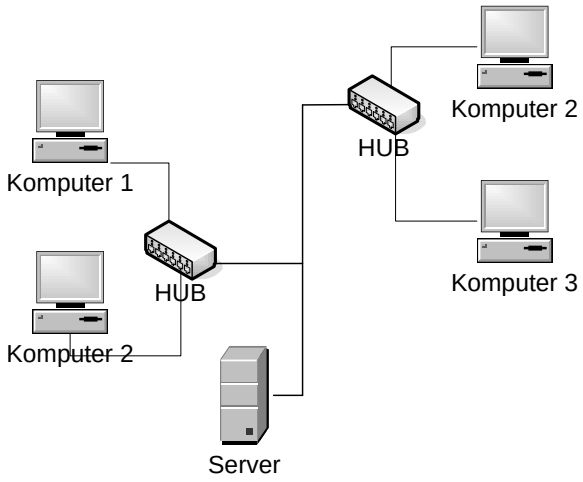
Topologi model ini didesain dimana setap *node* (*file server, workstation*, dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *concentrator* baik berupa HUB ataupun SWITCH. Data yang terkirim ke jaringan akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengontrol keseluruhan fungsi jaringan. Dia juga bertindak sebagai *repeater* dalam skala kecil dari aliran data. Konfigurasi pada jaringan model ini menggunakan kabel *Twisted Pair*. Gambar 2.8 menunjukkan topologi jaringan Bintang / *Star*.



Gambar 2.8 *Topologi STAR*

2.1.3.3 Topologi Tree

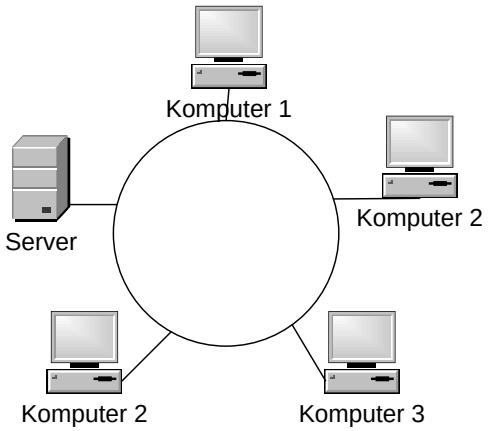
Sebuah topologi model ini merupakan perpaduan antara topologi Linear BUS / Garis Lurus dan Bintang, yang terdiri dari kelompok-kelompok dari *workstation* konfigurasi bintang yang terkoneksi ke kabel utama yang menggunakan topologi Linear BUS / Garis Lurus. Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada, dan memungkinkan sebuah perusahaan mengkonfigurasi jaringan sesuai dengan kebutuhannya.



Gambar 2.9 *Topologi Tree*

2.1.3.4 Topologi Ring

Dalam topologi Ring semua *workstation* dan *server* dihubungkan sehingga terbentuk suatu pola lingkaran atau cincin. Tiap *workstation* ataupun *server* akan menerima dan melewatkan informasi dari satu komputer ke komputer lain, bila alamat- alamat yang dimaksud sesuai maka informasi diterima dan bila tidak informasi akan dilewatkan.



Gambar 2.10 *Topologi RING*

2.2 Model Referensi Jaringan

Pada dasarnya komunikasi data adalah proses pengiriman informasi diantara dua titik menggunakan kode biner melewati saluran transmisi dan peralatan *switching*, dapat terjadi antara komputer dengan komputer, komputer dengan terminal atau komputer dengan peralatan. Komunikasi data merupakan gabungan dari teknik telekomunikasi dengan teknik pengolahan data. Proses ini melibatkan tiga komponen dasar, yaitu sumber, sistem tujuan dan sistem transmisi sebagai saluran tempat mengalirnya informasi

Sumber adalah pihak yang mengirim informasi yang bertugas mengirimkan berita atau informasi dan menempatkannya pada medium transmisi. Media transmisi tugasnya adalah menyalurkan berita yang diterima ke tempat tujuan. Penerima tugasnya adalah menerima berita yang dikirimkan oleh suatu sumber informasi. Sistem transmisi digunakan untuk mengatasi kendala jarak antara dua pihak yang saling berkomunikasi secara berjauhan. Komunikasi jarak jauh ini disebut telekomunikasi. Komunikasi dapat dilakukan dengan berbagai cara, yaitu komunikasi dengan suara, gambar dan cahaya.



Gambar 2.11 *Komunikasi Data*

Adapun tujuan komunikasi data adalah :

1. Memungkinkan pengiriman data dalam jumlah besar efisien, tanpa kesalahan dan ekonomis dari suatu tempat ke tempat yang lain
2. Memungkinkan penggunaan sistem komputer dan peralatan pendukung dari jarak jauh (*remote computer use*)
3. Memungkinkan penggunaan komputer secara terpusat maupun secara tersebar sehingga mendukung manajemen dalam hal kontrol, baik desentralisasi maupun sentralisasi
4. Mempermudah kemungkinan pengelolaan dan pengaturan data yang ada dalam berbagai macam sistem komputer
5. Mengurangi waktu untuk pengolahan data
6. Mendapatkan data langsung dari sumbernya
7. Mempercepat penyebarluasan informasi

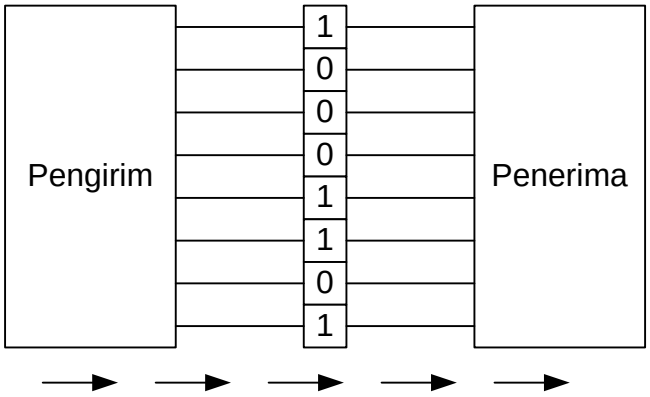
2.2.1 Jenis Transmisi

Jenis transmisi sinyal data atau informasi dalam suatu media komunikasi dapat dikelompokkan menjadi dua model, yaitu tansmisi paralel dan serial.

2.2.1.1 Transmisi Paralel

Pada transmisi paralel, satu konektor yang terdiri dari tujuh atau delapan bit (ASCII) ditransmisikan secara serentak setiap saat, seperti yang ditunjukkan pada gambar 2.12. Misalnya, bila digunakan kode ASCII, maka dibutuhkan sebanyak delapan jalur untuk mentransmisikan sekaligus 8 bit untuk satu karakter kode ASCII.

Pengiriman dengan model paralel ini memiliki kecepatan yang tinggi, karena setiap saat yang ditransmisikan secara paralel adalah bit–bit yang mewakili satu karakter. Komunikasi paralel biasanya digunakan untuk komunikasi jarak dekat, biasanya transmisi jenis ini digunakan untuk mentransmisikan sinyal di dalam komputer atau antara komputer ke printer.

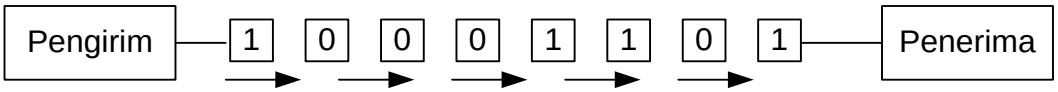


Gambar 2.12 *Transmisi Parallel*

2.2.1.2 Transmisi Serial

Transmisi model ini merupakan bentuk transmisi yang secara umum dipergunakan. Pada gambar 2.13 menunjukkan transmisi serial yang masing – masing bit dari suatu karakter dikirimkan secara berurutan, yaitu bit per bit, dimana satu bit diikuti oleh bit berikutnya. Dalam sistem ini, penerima akan mengumpulkan sejumlah bit (untuk sistem ASCII = 8 bit) yang dikirimkan oleh pengirim untuk kemudian dirakit menjadi satu karakter.

Meskipun kecepatannya lebih lambat dibandingkan dengan transmisi paralel, namun transmisi model ini paling banyak digunakan dalam lingkungan jaringan komputer.



Gambar 2.13 *Transmisi Serial*

2.2.2 Media Transmisi Kabel

2.2.2.1 Koaksial

Kabel koaksial berisi kawat tembaga keras sebagai intinya, yang sekelilingnya dilapisi dengan bahan penyekat. Terdapat dua jenis kabel koaksial, yaitu koaksial *Baseband* (kabel 50 Ohm) yang digunakan untuk transmisi digital dan kabel koaksial *Broadband* (kabel 75 Ohm) yang digunakan untuk transmisi analog.

1. ***Coaxial Thin Ethernet (Thinnet)***

   Thin Ethernet atau *Thinnet* memiliki keunggulan dalam hal biaya yang relatif lebih murah dibandingkan dengan tipe pengkabelan lain, serta pemasangan komponennya lebih mudah. Panjang kabel *thin coaxial* / RG-58 antara 0.5 – 185 meter dan maksimum 30 komputer terhubung.

2. ***Coaxial Thick Ethernet (Thicknet)***

   Dengan menggunakan *thick Ethernet* atau *thicknet*, jumlah komputer yang dapat dihubungkan dalam jaringan akan lebih banyak dan jarak antara komputer dapat diperbesar, tetapi biaya pengadaan pengkabelan ini lebih mahal serta pemasangannya relatif lebih sulit dibandingkan dengan *Thinnet*. Pada *Thicknet* digunakan *transceiver* untuk menghubungkan setiap komputer dengan sistem jaringan dan konektor yang digunakan adalah konektor tipe

DIX. Panjang kabel *transceiver* maksimum 50 meter, panjang kabel *Thick Ethernet* maksimum 500 meter dengan maksimum 100 *transceiver* terhubung. Pada gambar 2.14 menunjukkan kabel koaksial *Thicknet* dan koaksial *Thinnet.*



Gambar 2.14 *Kabel Thicknet dan Thinnet*

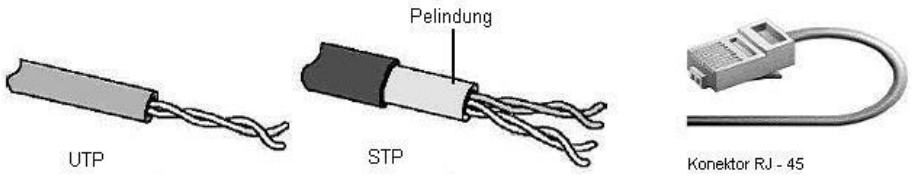
2.2.2.2 Twisted Pair

Kabel *Twisted Pair* ini terbagi menjadi dua jenis yaitu STP (*Shielded Twisted Pair*) dan UTP (*Unshielded.Twisted Pair*). *Shielded* adalah jenis kabel yang memiliki selubung pembungkus sedangkan *unshielded* tidak mempunyai selubung pembungkus, seperti ditunjukan pada gambar 2.15. Untuk koneksinya kabel jenis ini menggunakan konektor RJ-11 atau RJ-45. *Twisted pair* umumnya lebih handal (*reliable*) dibandingkan dengan *thin coax* karena HUB maupun SWITCH mempunyai kemampuan *data error correction* dan meningkatkan kecepatan transmisi. Saat ini ada beberapa kategori dari kabel *twisted pair*. Kategori 5 adalah yang paling *reliable,* memiliki kompabilitas yang tinggi, dan paling disarankan. Kategori 5 berjalan baik pada 10 Mbps dan *Fast Ethernet* (100Mbps). Kabel kategori 5 dapat dibuat *straight-through* atau *crossed*. Kabel *straight-through* digunakan untuk menghubungkan komputer ke HUB maupun

SWITCH. Kabel *crossed* digunakan untuk menghubungkan HUB ke HUB atau komputer dengan komputer. Panjang kabel maksimum kabel *Twisted-Pair* adalah 100 meter.

Tabel 2.1 *Kategori Kabel Twisted Pair*

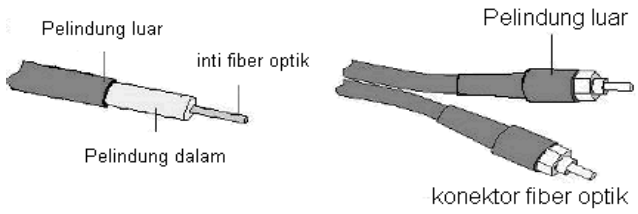
Kategori	Kegunaan
1	Digunakan untuk komunikasi suara (*voice*) biasanya untuk kabel telepon di rumah
2	Digunakan untuk komunikasi data sampai kecepatan 4 MBps (*local talk*)
3	Digunakan untuk transmisi data dengan kecepatan sampai 10 MBps dan digunakan untuk Ethernet dan Token Ring
4	Sama dengan kategori 3 tetapi dengan kecepatan trnsmisi dari 16 sampai 20 MBps
5	Digunakan untuk kecepatan transmisi sampai 100 MBps, biasanya digunakan untuk Fast ethernet atau network ATM



Gambar 2.15 *Kabel UTP, STP dan Konektor RJ-45*

2.2.2.3 Serat Optik

Jaringan yang menggunakan *Fibre Optic* (FO) biasanya adalah perusahaan besar, dikarenakan harga dan proses pemasangannya lebih sulit. Namun demikian, jaringan yang menggunakan serat optik dilihat dari segi kehandalan dan kecepatan tidak diragukan. Kecepatan pengiriman data dengan media serat optik lebih dari 100 Mbps dan bebas pengaruh lingkungan. Pada gambar 2.16 menunjukkan kabel serat optik beserta konektornya



Gambar 2.16 *Kabel Serat Optik dan Konektornya*

2.3 Model OSI

Model OSI merupakan sistem terbuka yang dapat diartikan sebagai suatu sistem yang terbuka untuk berkomunikasi dengan sistem-sistem yang lainnya. Model referensi OSI ini memiliki tujuh *layer*. Adapun prinsip-prinsip yang digunakan bagai ketujuh *layer* tersebut adalah:

a. Sebuah *layer* harus dibuat bila diperlukan tingkat abstraksi yang berbeda.
b. Setiap *layer* harus memiliki fungsi-fungsi tertentu.
c. Fungsi-fungsi tersebut sesuai dengan standarisasi internasional
d. Batas-batas layer harus dipilih untuk meminimalkan aliran informasi yang melewati *interface*.



Gambar 2.17 *Model Referensi OSI*

Sebuah *layer* tidak mendefinisikan protokol tunggal, tapi mendefinisikan suatu fungsi komunikasi data yang dapat dilakukan oleh sejumlah protokol. Jadi setiap *layer* dapat berisi banyak protokol, masing-masing menyediakan *service* yang cocok dengan fungsi *layer* tersebut. Sebagai contoh, *file transfer protocol* (FTP) dan *electronic mail protocol (E-Mail Protocol)* yang keduanya menyediakan *service* pada *user* dan keduanya merupakan bagian dari *layer* aplikasi.

Setiap protokol berkomunikasi dengan *peer* (pasangan) protokol yang sama dalam *layer* yang sama dalam *remote-system* misalnya FTP lokal merupakan *peer* dari FTP *remote*. Jadi secara abstrak setiap protokol hanya peduli terhadap komunikasi dengan *peer*-nya, tak perduli dengan *layer* di bawah dan di atasnya.

Adapun urutan *layer* dari *layer 1* sampai *layer* 7 sebagai berikut :

1. **Lapisan *Physical***
    a. Lapisan ini bertugas mendefinisikan besaran-besaran elektrik yang dipakai untuk berhubungan, misalnya atas tegangan yang benar (*Valid*) logika 0 dan 1, bentuk konektor yang digunakan, cara penyandian isyarat pada kabel dan lain-lain.
    b. Melaksanakan hubungan fisik melalui jalur transmisi.
2. **Lapisan *Data Link***
    a. Lapisan ini bertanggung jawab terhadap pengiriman data yang benar (*error free*) .
    b. Mengirim data secara paket dan memeriksa kesalahan pada pengiriman data.
    c. Menetapkan *prosedur* pengaksesan.
3. **Lapisan *Network***
    a. Lapisan ini menentukan jalur atau *route* yang harus ditempuh oleh paket untuk mencapai tujuannya, yang biasanya disebut *routing layer*.
    b. Pada lapisan ini pula menentukan *prosedur* untuk menghindari terjadinya kemacetan serta mengalihkan perjalanan paket ke jalur yang lain serta memungkinkan bukan jalur yang pendek.
4. **Lapisan *Transport***
    a. Lapisan ini mengatur bagaimana data atau informasi itu akan dibawa ke tempat tujuan termasuk juga jaminan kualitas dari pelayanan pengiriman data.

b. Lapisan ini pula memilih fasilitas pengiriman yang sebagaimana yang dikehendaki.

5. Lapisan *Session*

a. Lapisan ini berfungsi menentukan kapan dimulai dan kapan diakhirinya suatu babak atau *session* komunikasi atau koneksi.

b. Suatu *session* akan dimulai jika *session* sebelumnya telah selesai.

c. Melihat data yang sudah dikirim dan diterima oleh proses.

6. Lapisan *Presentation*

a. Lapisan ini mendefinisikan bagaimana aplikasi dapat memasuki jaringan dan mengatur supaya data yang dipertukarkan atau dikirim ke lapisan aplikasi atau pemakai akhir dapat dimengerti bentuknya.

b. Dalam lapisan ini, peralatan atau sistem yang menggunakan format data yang berlainan dapat saling berkomunikasi .

c. Lapisan ini juga menyediakan enkripsi ataupun kompresi data jika diperlukan.

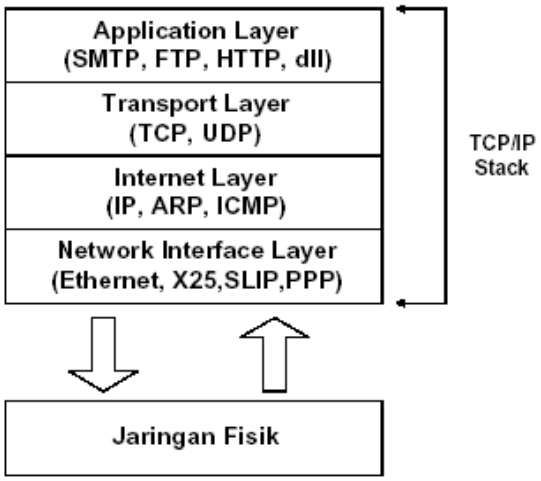
7. Lapisan *Application*

a. Lapisan ini berfungsi mengelola interaksi antara program dan pemakai.

b. Lapisan ini juga menerima perintah atau input dari pemakai dan mengembalikan kode *error* atau pesan kesalahan kepada pemakai jika terjadi *error*.

2.4 Model TCP/IP

Model ini terdiri dari empat lapisan. Pada model ini tidak memiliki *presentation* dan *session layers*. Fungsi kedua lapisan ini dapat dilakukan sesuai dengan kebutuhan.



Gambar 2.18 *Model TCP/IP*

Berikut merupakan penjelasan lapisan-lapisan atau *layer-layer* pada model TCP/IP mulai dari lapisan teratas :

1. ***Aplication Layer***
   a. *Layer* ini melingkupi semua proses yang menggunakan protokol *transport layer* untuk mengirimkan data.
   b. Karena *layer* ini berhubungan langsung dengan *service* yang ditawarkan pada pemakai jaringan maka protokol-protokol baru masih ditambahkan yang memperkaya pelayanan pada *user*.

2. Transport Layer

Dua protokol terpenting dalam layer ini adalah TCP *(Transmission Control Protocol)* dan UDP *(User datagram Protocol)*. TCP menyediakan pelayanan pengiriman data yang handal dengan deteksi dan koreksi kesalahan dari ujung ke ujung (*end to end*). Sedangkan UDP menyediakan pelayanan pengiriman *datagram* yang *connection-less* dan tanpa dilengkapi deteksi dan koreksi kesalahan. Kedua *protokol* mengirimkan data antara *layer* aplikasi dan *layer internet*. TCP merupakan protokol yang *connection oriented* dan *handshake*.

3. *Internet Layer*

a. Internet Protokol (IP)

IP merupakan inti dari TCP/IP dan merupakan protokol terpenting dalam *Internet Layer*. IP menyediakan pelayanan pengiriman paket elementer dimana jaringan TCP/IP dibangun.

Fungsi *Internet Protocol (IP)*

1. Mendefinisikan *datagram*, yang merupakan unit transmisi elementer di internet.
2. Mendefinisikan skema pengalamatan internet.
3. Melewatkan data antara *Network Access Layer* dan *Host to Host Transport layer* serta *routing datagram* ke *remote host.*
4. Menjalankan fragmentasi dan penyusunan kembali *datagram*.

5. IP merupakan protokol yang *Connection-less* (tidak memerlukan *handshake*), tidak dilengkapi dengan *error detection* dan *error recovery*.

b. Datagram

Datagram adalah format paket yang didefinisikan oleh IP. Internet sebagaimana asalnya *ARPANET* merupakan jaringan yang berbasis pada *packet-switching*. Jadi *datagram* merupakan unit transmisi elementer dalam jaringan TCP/IP. IP mengirimkan *datagram* dengan mengecek *destination address* dalam *header* kontrol di awal *datagram*. Jika *address* tujuan tidak berada di jaringan lokal maka paket dilewatkan ke *gateway* (*device* yang men-*switch* paket antara jaringan fisik yang berbeda). Memutuskan *gateway* yang mana yang digunakan untuk mencapai *address* tujuan disebut sebagai *routing*.

4. *Network Access Layer*

a. Merupakan *layer* terbawah dari hirarki protokol TCP/IP.

b. Menyediakan sarana untuk sistem untuk mengirim data ke *device* lain yang terhubung ke *network*.

c. Mendefinisikan bagaimana *datagram* ditransmisikan pada *network*.

d. Fungsi lain yang ditangani pada *level* ini termasuk enkapsulasi *datagram* ke dalam *frame* yang ditransmisikan oleh jaringan dan konversi *IP address* ke dalam alamat yang cocok untuk jaringan fisik dimana *datagram* ditransmisikan

2.5 Keamanan Jaringan

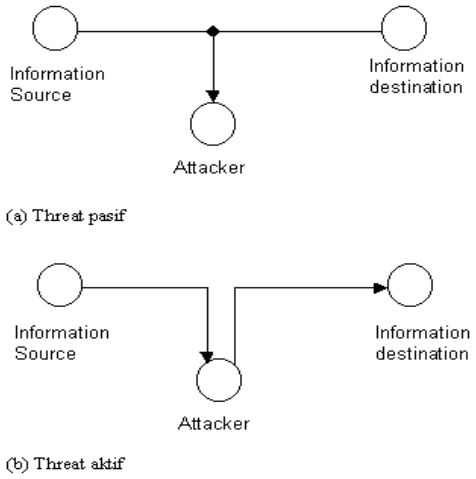
Keamanan jaringan didefinisikan sebagai sebuah perlindungan dari sumber daya daya terhadap upaya penyingkapan, modifikasi, utilisasi, pelarangan dan perusakan oleh *person* yang tidak diijinkan. Beberapa ahli jaringan mengatakan bahwa hanya ada satu cara mudah dan ampuh untuk mewujudkan sistem jaringan komputer yang aman yaitu dengan menggunakan pemisah antara komputer dengan jaringan selebar satu inci, dengan kata lain, hanya komputer yang tidak terhubung ke jaringanlah yang mempunyai keamanan yang sempurna. Meskipun ini adalah solusi yang buruk, tetapi ini menjadi *trade-off* antara pertimbangan fungsionalitas dan memasukan kekebalan terhadap gangguan.

Protokol suatu jaringan sendiri dapat dibuat aman. *Server-server* baru yang menerapkan protokol-protokol yang sudah dimodifikasi harus diterapkan. Sebuah protokol atau layanan (*service*) dianggap cukup aman apabila mempunyai kekebalan ITL kelas 0. Sebagai contoh, protokol seperti FTP atau Telnet, yang sering mengirimkan password secara terbuka melintasi jaringan, dapat dimodifikasi dengan menggunakan teknik enkripsi. Jaringan daemon, seperti *sendmail* atau *fingered*, dapat dibuat lebih aman oleh pihak vendor dengan pemeriksaan kode dan *patching*. Permasalahan *mis*-konfigurasi, seperti misalnya spesifikasi yang tidak benar dari *netgroup,* dapat menimbulkan permasalahan kekebalan (menjadi rentan). Demikian juga kebijakan dari departemen teknologi informasi seringkali memunculkan kerumitan pemecahan masalah untuk membuat sistem menjadi kebal.

2.5.1 Tipe *Threat* (Ancaman)

Terdapat dua kategori *threat* yaitu ***threat* pasif** dan ***threat* aktif**.

***Threat* pasif** melakukan pemantauan dan atau perekaman data selama data ditranmisikan lewat fasilitas komunikasi. Tujuan penyerang adalah untuk mendapatkan informasi yang sedang dikirimkan. Kategori ini memiliki dua tipe *yaitu release of message contain* dan *traffic analysis*. Tipe *Release of message contain* memungkinan penyusup utnuk mendengar pesan, sedangkan tipe *traffic analysis* memungkinan penyusup untuk membaca *header* dari suatu paket sehingga bisa menentukan arah atau alamat tujuan paket dikirimkan. Penyusup dapat pula menentukan panjang dan frekuensi pesan.



Gambar 2.19 *Kategori Threat*

***Threat* aktif** merupakan pengguna gelap suatu peralatan terhubung fasilitas komunikasi untuk mengubah transmisi data atau mengubah isyarat kendali atau memunculkan data atau isyarat kendali palsu. Untuk kategori ini terdapat tida tipe yaitu : *message-stream modification, denial of message service*

dan *masquerade*. Tipe *message-stream modification* memungkinan pelaku untuk memilih untuk menghapus, memodifikasi, menunda, melakukan *reorder* dan menduplikasi pesan asli. Pelaku juga mungkin untuk menambahkan pesan-pesan palsu. Tipe *denial of message service* memungkinkan pelaku untuk merusak atau menunda sebagian besar atau seluruh pesan. Tipe *masquerade* memungkinkan pelaku untuk menyamar sebagi *host* atau switch asli dan berkomunikasi dengan yang *host* yang lain atau switch untuk mendapatkan data atau pelayanan.

2.5.2 *Internet Threat Level*

Celah-celah keamanan sistem internet, dapat disusun dalam skala klasifikasi. Skala klasifikasi ini disebut dengan istilah skala *Internet Threat Level* atau skala ITL. Ancaman terendah digolongkan dalam ITL kelas 0, sedangkan ancaman tertinggi digolongkan dalam ITL kelas 9. Tabel 2.1 menjelaskan masing-masing kelas ITL.

Kebanyakan permasalahan keamanan dapat diklasifikasikan ke dalam 3 kategori utama, tergantung pada kerumitan perilaku ancaman kepada sistem sasaran, yaitu :

a. Ancaman-ancaman lokal.
b. Ancaman-ancaman *remote*
c. Ancaman-ancaman dari lintas *firewall*

Klasifikasi ini dapat dipisah dalam derajat yang lebih rinci, yaitu :

a. *Read access*
b. *Non-root write and execution access*
c. *Root write and execution access*

Tabel 2.2 *Skala Internet Threat Level (ITL)*

Kelas	Penjelasan
0	Serangan pada *Denial of service* menyebabkan pengguna tidak dapat mengakses file file atau program
1	pengguna lokal bisa mendapatkan akses baca ke file di sistem lokal.
2	Pengguna lokal dapat menulis dan atau akses ke file eksekusi *non-root* pada sistem
3	Pengguna lokal bisa mendapatkan menulis dan atau akses ke file eksekusi milik *root* pada sistem.
4	pengguna lokal bisa mendapatkan menulis dan atau akses ke file eksekusi milik *root* pada sistem
5	*Remote* pengguna di jaringan yang sama dapat menulis dan atau akses ke file eksekusi *non-root* pada sistem atau yang dikirimkan melalui jaringan.
6	*Remote* pengguna di jaringan yang sama dan atau akses ke file eksekusi milik *root* pada sistem.
7	*Remote* pengguna di *firewall* bisa mendapatkan akses baca ke file pada sistem atau yang dikirimkan melalui jaringan
8	*Remote* pengguna di *firewall* dapat menulis dan atau akses ke file eksekusi *non-root* pada sistem atau yang dikirimkan melalui jaringan.

9	*Remote* pengguna di *firewall* dapat menulis dan atau akses ke file eksekusi milik *root* pada sistem.

Seberapa besar tingkat ancaman dapat diukur dengan melihat beberapa faktor, antara lain :

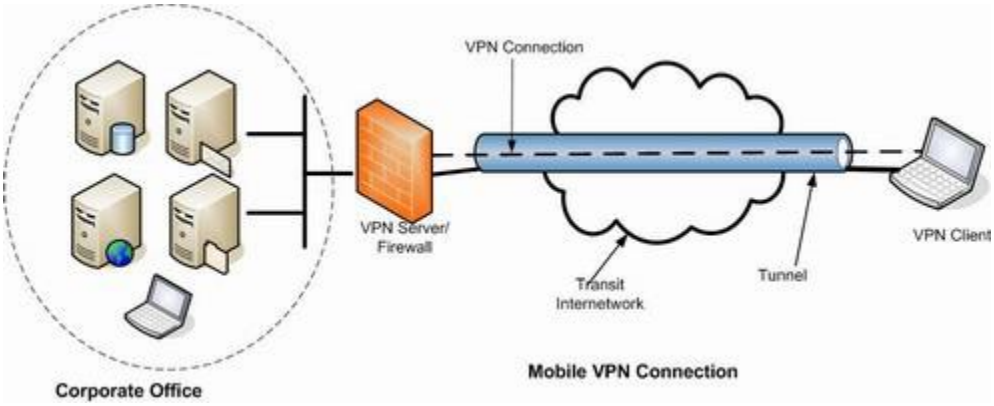
a. Kegunaan sistem
b. Kerahasiaan data dalam sistem.
c. Tingkat kepentingan dari integritas data
d. Kepentingan untuk menjaga akses yang tidak boleh terputus
e. Profil pengguna
f. Hubungan antara sistem dengan sistem yang lain.

2.6 Virtual Private Network

Virtual Private Network (VPN) adalah sebuah teknologi komunikasi yang memungkinkan untuk dapat terkoneksi ke jaringan publik dan menggunakannya untuk dapat bergabung dengan jaringan lokal. Dengan cara tersebut maka akan didapatkan hak dan pengaturan yang sama seperti halnya berada di dalam kantor atau LAN itu sendiri, walaupun sebenarnya menggunakan jaringan milik publik.

VPN dapat terjadi antara dua *end-system* atau dua komputer atau antara dua atau lebih jaringan yang berbeda. VPN dapat dibentuk dengan menggunakan teknologi *tunneling* dan enkripsi. Koneksi VPN juga dapat terjadi pada semua layer pada protokol OSI, sehingga komunikasi menggunakan VPN dapat digunakan untuk berbagai keperluan. Dengan demikian, VPN juga dapat

dikategorikan sebagai infrastruktur WAN alternatif untuk mendapatkan koneksi *point-to-point* pribadi antara pengirim dan penerima. Dan dapat dilakukan dengan menggunakan media apa saja, tanpa perlu media *leased line* atau *frame relay.*



Gambar 2.20 *Ilustrasi VPN*

2.6.1 Fungsi Utama Teknologi VPN

Teknologi VPN menyediakan beberapa fungsi utama untuk penggunaannya. Fungsi-fungsi utama tersebut antara lain sebagai berikut.

1. ***Confidentially (Kerahasiaan)***

   Dengan digunakannnya jaringan *publik* yang rawan pencurian data, maka teknologi VPN menggunakan sistem kerja dengan cara mengenkripsi semua data yang lewat melaluinya. Dengan adanya teknologi enkripsi tersebut, maka kerahasiaan data dapat lebih terjaga. Walaupun ada pihak yang dapat menyadap data yang melewati internet bahkan jalur VPN itu sendiri, namun belum tentu dapat membaca data tersebut, karena data tersebut telah teracak. Dengan menerapkan sistem

enkripsi ini, tidak ada satupun orang yang dapat mengakses dan membaca isi jaringan data dengan mudah.

2. ***Data Intergrity*** **(Keutuhan data)**

Ketika melewati jaringan internet, sebenarnya data telah berjalan sangat jauh melintasi berbagai negara. Pada saat perjalanan tersebut, berbagai gangguan dapat terjadi terhadap isinya, baik hilang, rusak, ataupun dimanipulasi oleh orang yang tidak seharusnya. Pada VPN terdapat teknologi yang dapat menjaga keutuhan data mulai dari data dikirim hingga data sampai ditempat tujuan.

3. ***Origin Authentication*** **(Autentikasi sumber)**

Teknologi VPN memiliki kemampuan untuk melakukan autentikasi terhadap sumber-sumber pengirim data yang akan diterimanya. VPN akan melakukan pemeriksaan terhadap semua data yang masuk dan mengambil informasi dari sumber datanya. Kemudian, alamat sumber data tersebut akan disetujui apabila proses autentikasinya berhasil. Dengan demikian, VPN menjamin semua data yang dikirim dan diterima berasal dari sumber yang seharusnya. Tidak ada data yang dipalsukan atau dikirim oleh pihak-pihak lain.

4. ***Non-repudiation***

Yaitu mencegah dua pihak dari menyangkal bahwa mereka telah mengirim atau menerima sebuah file mengakomodasi perubahan.

5. **Kendali akses**

Menentukan siapa yang diberikan akses ke sebuah sistem atau jaringan, sebagaimana informasi apa dan seberapa banyak seseorang dapat menerima.

2.6.2 Perangkat VPN

Pada dasarnya, semua perangkat komputer yang dilengkapi dengan fasilitas pengalamatan IP dan diinstal dengan aplikasi pembuat *tunnel* dan algoritma enkripsi dan dekripsi, dapat dibangun komunikasi VPN di dalamnya. Komunikasi VPN dengan *tunneling* dan enkripsi ini dapat dibangun antara sebuah *router* dengan *router* yang lain, antara sebuah *router* dengan beberapa *router*, antara PC dengan *server* VPN *concentrator*, antara *router* atau PC dengan *firewall* berkemampuan VPN, dan masih banyak lagi.

2.6.3 Teknologi *Tunneling*

Teknologi *tunneling* merupakan teknologi yang bertugas untuk manangani dan menyediakan koneksi *point-to-point* dari sumber ke tujuannya. Disebut *tunnel* karena koneksi *point-to-point* tersebut sebenarnya terbentuk dengan melintasi jaringan umum, namun koneksi tersbut tidak mempedulikan paket-paket data milik orang lain yang sama-sama melintasi jaringan umum tersebut, tetapi koneksi tersebut hanya melayani transportasi data dari pembuatnya. Hal ini sama dengan seperti penggunaan jalur *busway* yang pada dasarnya menggunakan jalan raya, tetapi dia membuat jalur sendiri untuk dapat dilalui bus khusus.

Koneksi *point-to-point* ini sesungguhnya tidak benar-benar ada, namun data yang dihantarkannya terlihat seperti benar-benar melewati koneksi pribadi yang bersifat *point-to-point*.

Teknologi ini dapat dibuat di atas jaringan dengan pengaturan *IP Addressing* dan *IP Routing* yang sudah matang. Maksudnya, antara sumber *tunnel* dengan tujuan *tunnel* telah dapat saling berkomunikasi melalui jaringan dengan pengalamatan IP. Apabila komunikasi antara sumber dan tujuan dari *tunnel* tidak dapat berjalan dengan baik, maka *tunnel* tersebut tidak akan terbentuk dan VPN pun tidak dapat dibangun.

Apabila *tunnel* tersebut telah terbentuk, maka koneksi *point-to-point "*palsu" tersebut dapat langsung digunakan untuk mengirim dan menerima data. Namun, di dalam teknologi VPN, *tunnel* tidak dibiarkan begitu saja tanpa diberikan sistem keamanan tambahan. *Tunnel* dilengkapi dengan sebuah sistem enkripsi untuk menjaga data-data yang melewati *tunnel* tersebut. Proses enkripsi inilah yang menjadikan teknologi VPN menjadi mana dan bersifat pribadi.

2.6.4 Teknologi Enkripsi

Teknologi enkripsi menjamin data yang berlalu-lalang di dalam *tunnel* tidak dapat dibaca dengan mudah oleh orang lain yang bukan merupakan komputer tujuannya. Semakin banyak data yang lewat di dalam *tunnel* yang terbuka di jaringan publik, maka teknologi enkripsi ini semakin dibutuhkan. Enkripsi akan mengubah informasi yang ada dalam *tunnel* tersebut menjadi sebuah *ciphertext* atau teks yang dikacaukan dan tidak ada artinya sama sekali apabila dibaca secara langsung. Untuk dapat membuatnya kembali memiliki arti

atau dapat dibaca, maka dibutuhkan proses dekripsi. Proses dekripsi terjadi pada ujung-ujung dari hubungan VPN. Pada kedua ujung ini telah menyepakati sebuah algoritma yang akan digunakan untuk melakukan proses enkripsi dan dekripsinya. Dengan demikian, data yang dikirim aman sampai tempat tujuan, karena orang lain di luar *tunnel* tidak memiliki algoritma untuk membuka data tersebut.

2.6.5 Metode Sistem Keamanan VPN

Sistem keamanan di VPN menggunakan beberapa metode lapisan sistem keamanan, diantaranya :

1. **Metode *Tunneling* (terowongan),**

   Membuat terowongan virtual diatas jaringan publik menggunakan protokol seperti *Point to Point Protocol* (PPTP), *Layer 2 Tunnelin Protocol* (L2TP), *Secure Socket Layer* (SSL) *Generic Routing Encapsulation* (GRE) atau IPSec. PPTP dan L2TP adalah *layer* 2 *tunneling protocol*. keduanya melakukan pembungkusan *payload* pada *frame Point to Point Protocol* (PPP) untuk di lewatkan pada jaringan. IPSec berada di layer 3 yang menggunakan packet, yang akan melakukan pembungkusan IP *header* sebelum dikirim ke jaringan.

2. **Metode Enkripsi**

   Untuk *Encapsulations* (membungkus) paket data yang lewat di dalam *tunneling*, data yang dilewatkan pada pembungkusan tersebut, data disini akan dirubah dengan metode algoritma kriptograpi tertentu seperti DES, 3DES, atau AES.

3. **Metode Autentikasi User**

   Karena banyak user yang akan mengakses biasanya digunakan beberapa metode autentikasi user seperti *Remote Access Dial In User Services* (RADIUS) dan *Digital Certificates*.

4. **Integritas Data**

   Paket data yang dilewatkan di jaringan publik perlu penjaminan integritas data / kepercayaan data apakah terjadi perubahan atau tidak. Metode VPN menggunakan HMA C-MD5 atau HMA CSHA1 untuk menjadi paket tidak dirubah pada saat pengiriman.

2.6.6 Jenis implementasi VPN

2.6.6.1 *Remote Access* VPN

Pada umumnya implementasi VPN terdiri dari 2 macam. Pertama adalah *remote access* VPN, dan yang kedua adalah s*ite-to-site* VPN. *Remote access* yang biasa juga disebut *virtual private dial-up network* (VPDN), menghubungkan antara pengguna yang *mobile* dengan *local area network* (LAN).

Jenis VPN ini digunakan oleh pegawai perusahaan yang ingin terhubung ke jaringan khusus perusahaannya dari berbagai lokasi yang jauh (*remote*) dari perusahaannya. Biasanya perusahaan yang ingin membuat jaringan VPN tipe ini akan bekerjasama dengan *enterprise service provider* (ESP). ESP akan memberikan suatu *network access server* (NAS) bagi perusahaan tersebut. ESP juga akan menyediakan *software* klien untuk komputer-komputer yang digunakan pegawai perusahaan tersebut.

Untuk mengakses jaringan lokal perusahaan, pegawai tersebut harus terhubung ke NAS dengan men-*dial* nomor telepon yang sudah ditentukan. Kemudian dengan menggunakan *sotware* klien, pegawai tersebut dapat terhubung ke jaringan lokal perusahaan. Perusahaan yang memiliki pegawai yang ada di lapangan dalam jumlah besar dapat menggunakan *remote access* VPN untuk membangun WAN. VPN tipe ini akan memberikan keamanan, dengan mengenkripsi koneksi antara jaringan lokal perusahaan dengan pegawainya yang ada di lapangan. Pihak ketiga yang melakukan enkripsi ini adalah ISP.

2.6.6.2 *Site-to-site* VPN

Jenis implementasi VPN yang kedua adalah *site-to-site* VPN. Implementasi jenis ini menghubungkan antara 2 kantor atau lebih yang letaknya berjauhan, baik kantor yang dimiliki perusahaan itu sendiri maupun kantor perusahaan mitra kerjanya. VPN yang digunakan untuk menghubungkan suatu perusahaan dengan perusahaan lain (misalnya mitra kerja, *supplier* atau pelanggan) disebut ekstranet. Sedangkan bila VPN digunakan untuk menghubungkan kantor pusat dengan kantor cabang, implementasi ini termasuk jenis intranet *site-to-site* VPN.

Ada empat buah protokol yang biasa dan sering digunakan dalam pengimplementasian VPN(Virtual Private Network).

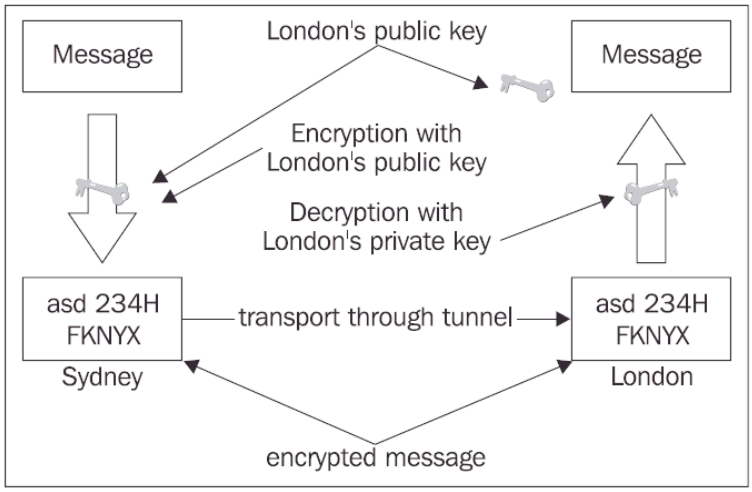
1. Ipsec (IP *Security Protocol*)
2. *Layer*-2 *Forwarding*
3. *Layer*-2 *Tunneling Protocol* (L2TP)
4. *Point to Point Tunneling Protocol*

2.7 *Secure Socket Layer* (SSL)

Secure Socket Layer (SSL) adalah protokol yang digunakan untuk berkomunikasi lewat internet secara aman. Teknologi SSL menggunakan konsep teknologi kriptografi kunci publik untuk bisa mencapai komunikasi yang aman antar *server* dan pengunjung (*client*). Kedua pihak yang berkomunikasi ini (*server* dan *client*) saling mengirimkan data yang disamarkan dan untuk membacanya digunakan sandi dan kunci yang dimiliki kedua pihak yang berkomunikasi tersebut, sehingga pihak lain yang mencoba menyadap data yang dikirim tersebut tidak akan dapat membacanya karena sandi dan kunci yang dibutuhkan tersebut hanya dimiliki oleh kedua pihak yang berkomunikasi.

2.7.1 *Asymmetric Encryption*

SSL/TLS menggunakan salah satu teknologi enkripsi terbaik yang disebut enkripsi asimetric untuk memastikan identitas dari mitra VPN. Kedua mitra enkripsi memiliki dua kunci masing-masing: satu publik dan satu privat. Kunci publik itu diserahkan kepada mitra komunikasi, untuk meningkatkan datanya. Oleh karena algoritma mathematical yang terpilih digunakan untuk menciptakan pasangan kunci *public/private*, hanya kunci pribadi penerima itu yang dapat mengurangi data yang disandikan oleh kunci publiknya.



Gambar 2.21 *Asymmetric Encryption dengan SSL/TLS*

Pada Gambar 2.21 di atas, suatu pesan teks dienkripsi pada Sydney dengan kunci publik dari London. Kode acak dikirim kepada London, di mana hal itu dapat diuraikan menggunakan kunci pribadi London. Hal ini bisa dilakukan sebaliknya untuk data dari London ke Sydney, yang dienkripsi oleh kunci publik Sydney di London dan hanya dapat dideskripsikan oleh kunci pribadi Sydney di Sydney.

Suatu prosedur yang serupa dapat juga digunakan untuk tujuan pengesahan: London mengirim suatu nomor acak yang besar kepada Sydney, di mana nomor ini disandikan dengan kunci pribadi dan dikembalikan. Di London, kunci publik Sydney dapat memecahkan kode nomor. Jika nomor yang dikirim dan dideskripsi sesuai, kemudian pengirim harus memiliki kunci pribadi Sydney. Ini disebut tanda tangan digital.

2.7.2 SSL/TLS Security

SSL juga yang dikenal sebagai TLS, adalah suatu protokol yang awalnya dirancang oleh Netscape Communications Corporation untuk memastikan integritas data dan keaslian untuk perkembangan Internet pada tahun 1990. Semua orang yang menggunakan browser modern dapat mengambil bagian di dalam komunikasi yang dienkripsi. SSL/TLS adalah satu teknologi yang terkemuka yang sedang digunakan di mana-mana web untuk perbankan, e-commerce, atau aplikasi di mana keleluasaan pribadi lain manapun dan keamanan diperlukan. Itu sedang pasti terkendali, *debugged*, yang diuji, dan yang diperbaiki oleh kedua-duanya pengembang sumber dan kepemilikan yang terbuka dan banyak korporasi.

2.8 SSL VPN

2.8.1 Konsep SSL VPN

Konsep dan Teknologi SSL (*Secure Socket Layer*) VPN dapat menjawab kebutuhan untuk mengakses sumberdaya perusahaan melalui penggunaan jaringan internet yang sudah tersedia dan bercakupan luas. Dibandingkan jaringan *leased lines* atau *frame relay*, SSL VPN menggunakan infrastruktur publik yang sudah ada di internet untuk melakukan pertukaran data antara kantor pusat sebuah perusahaan dan kantor cabangnya.

Karena dilewatkan pada jaringan internet publik, permasalahan konsep SSL VPN muncul sama seperti jaringan IP pada umumnya. Salah satu masalah jaringan internet (*IP public*) adalah tidak mempunyai dukungan yang baik terhadap keamanan. SSL VPN muncul untuk mengatasi persoalan keamanan

tersebut. Dasar dari konsep SSL VPN ini adalah penggunaan infrastruktur IP untuk hubungan suatu perusahaan dengan kantor cabangnya dengan cara pengalamatan secara *private* dengan melakukan pengamanan terhadap transmisi paket data.

2.8.2 Spesifikasi Teknis SSL VPN

SSL juga digunakan sebagai protokol kriptografi yang digunakan sebagai protokol keamanan komunikasi data di internet. Di dalam proses kerjanya, SSL terdiri dari beberapa fase, yaitu:

1. Negosiasi antar peer terhadap algoritma yang didukung oleh masing-masing peer/endpoint tersebut.
2. Pertukaran *Public Key* (*PKI*) untuk mekanisme enkripsi dan autentifikasi berbasis sertifikat.
3. Enkripsi simetris enkripsi

SSL VPN gateway memegang peranan penting untuk pengawasan transaksi pertukaran data/akses sehingga harus mempunyai mekanisme untuk memblokir akses ke jaringan *private* perusahaan ketika jumlah akses bersamaan melebihi batas yang telah ditentukan (tersedia dalam profil pelanggan korporasi). Disamping itu, SSL VPN gateway harus mempunyai mekanisme sehingga setiap pelanggan korporasi memiliki administratornya sendiri yang mempunyai kewenangan seperti mengkostumisasi halaman situs web, mengatur akses user pada sumber daya mereka. Tujuan pengawasan tersebut agar mencegah serangan yang dilakukan terhadap pesan yang disandikan, serta dibobolnya *public key* oleh pihak yang tidak berkepentingan. Perlu memeriksa sertifikat yang diterimanya

agar lebih yakin bahwa dia sedang berkomunikasi dengan *provider* yang diinginkan. *Client* memeriksa sertifikat digital itu dengan membandingkan tanda tangan OS (otoritas sertifikat) pada sertifikat digital itu dengan daftar OS yang dimiliki. Penyertaan serfikat digital OS utama pada browser akan menghindarkan *client* dari pemalsuan sertifikat OS utama.

Adapun mekanisme *profiling* terhadap *remote access*, SSL VPN berdasarkan pada *previledge access* per pelanggan korporasi. SSL VPN bersama dengan sistem sekuriti harus mampu mengolah dan menerapkan kebijakan yang sesuai terhadap akses *incoming* berdasarkan pada pilihan IP *address* (*source, destination address*), protokol yang digunakan oleh trafik (TCP, UDP).

Adapun algoritma kriptografi yang saat ini umumnya digunakan adalah:

1. Untuk kriptografi *Public Key*: RSA, Diffie-Hellman, DSA atau Fortezza;
2. Untuk enkripsi simetris: RC2, RC4, IDEA, DES, Triple DES atau AES;
3. Untuk fungsi *hash* searah: MD5 atau SHA.

2.8.2.1 Protokol

Terdapat empat protokol yang biasa digunakan untuk mendukung implementasi SSL VPN di internet, yaitu:

1. *Point-to-Point Tunneling Protocol* (PPTP)
2. *Layer-2 Forwarding* (L2F)
3. *Layer-2 Tunneling Protocol* (L2TP)
4. IP *Security Protocol*

Berbeda dengan penerapan masing-masing protokol di atas secara *stand alone*, penerapan SSL VPN lebih memberikan kebijakan *security* di dalam setiap

koneksi yang dibangun untuk dapat melakukan akses ke sumber daya yang spesifik sesuai dengan kebutuhan, lokasi dan perangkat user, tanpa keharusan untuk menginstall software tertentu di sisi *client*.

2.8.2.2 Fungsi

Beberapa Spesifikasi Fungsi yang umumnya ada dalam perangkat SSL VPN antara lain adalah:

1. Mode Operasi

Ada dua mode operasi yang biasanya ada pada perangkat SSL VPN yaitu:

a. Static routes : untuk mendukung pembentukan VPN dengan menggunakan rute statis yang di *assign* oleh admin secara permanent dengan software operasi.
b. Unrestricted : untuk mendukung pembentukan VPN dengan menggunakan mode berdasarkan kebutuhan *support* dari user, biasanya dilakukan secara dinamis dan *on demand.*

2. Atribut SSL VPN

Beberapa atribut yang dimiliki secara umum oleh perangkat SSL VPN yang menunjukkan kapabilitas sistem tersebut adalah:

a. Kapabilitas kreasi jumlah tunnel SSL VPN *Tunnels.*
b. Kemampuan mendukung Versi SSL.
c. Kemampuan melakukan enkripsi dan dukungannya terhadap metode enkripsi yang saat ini ada.
d. SSL *Message Integrity.*
e. *Certificate Support.*

3. Mode Akses

Dalam mendukung kerja SSL VPN beberapa mode akses dimiliki oleh perangkat SSL VPN secara umum, yaitu:

a. *Browser Support*

b. *Full Network Access Support*

c. *Port Forwarding*

d. *Support for Terminal Services and* VNC

e. WebCIFS, Telnet, FTP, SSH, WebFTP

f. *User and group-level*

g. Sup*port*s automatic cache cleanup after session termination

4. *User Repository Support,* yang diantaranya adalah kemampuan untuk:

a. *Local User Database*

b. Microsoft Active Directory

c. LDAP *Directory*

d. NT *Domains*

e. RADIUS (PAP, CHAP, MSCHAP, MSCHAPv2)

5. System Management, yang mendukung pengoperasian dan pemeliharaan sistem yang diantaranya adalah fungsi-fungsi sebagai berikut: E-mail notification, NTP *support*, *Remote logging*, Telnet, Web GUI

2.9 Internet

Internet (*Interconnected Network*) adalah sebuah sistem komunikasi global yang menghubungkan komputer-komputer dan jaringan-jaringan komputer di seluruh dunia tanpa mengenal batas teritorial, hukum dan budaya, sebagai sarana berkomunikasi dan menyebarkan informasi. Setiap komputer dan jaringan terhubung - secara langsung maupun tidak langsung - ke beberapa jalur utama yang disebut internet *backbone* dan dibedakan satu dengan yang lainnya menggunakan *unique name* yang biasa disebut dengan alamat IP 32 bit. Contoh: 202.155.4.230 .

Menurut Lani Sidharta (1996) : walaupun secara fisik Internet adalah interkoneksi antar jaringan komputer namun secara umum Internet harus dipandang sebagai sumber daya informasi. Isi internet adalah informasi, dapat dibayangkan sebagai suatu database atau perpustakaan multimedia yang sangat besar dan lengkap. Bahkan Internet dipandang sebagai dunia dalam bentuk lain (maya) karena hampir seluruh aspek kehidupan di dunia nyata ada di Internet seperti bisnis, hiburan, olah raga, politik dan lain sebagainya.

2.9.1 Sejarah Internet

Awal mula jaringan Internet yang kita kenal saat ini pertama kali dikembangkan tahun 1969 oleh Departemen Pertahanan Amerika Serikat dengan nama ARPAnet (US Defense Advanced Research Projects Agency). ARPAnet dibangun dengan sasaran untuk membuat suatu jaringan komputer yang tersebar untuk menghindari pemusatan informasi di satu titik yang dipandang rawan untuk dihancurkan apabila terjadi peperangan. Dengan cara ini diharapkan apabila satu

bagian dari jaringan terputus, maka jalur yang melalui jaringan tersebut dapat secara otomatis dipindahkan ke saluran lainnya.

Di awal 1980-an, ARPANET terpecah menjadi dua jaringan, yaitu ARPANET dan Milnet (sebuah jaringan militer), akan tetapi keduanya mempunyai hubungan sehingga komunikasi antar jaringan tetap dapat dilakukan. Pada mulanya jaringan interkoneksi ini disebut DARPA Internet, tapi lama-kelamaan disebut sebagai internet saja. Sesudahnya, internet mulai digunakan untuk kepentingan akademis dengan menghubungkan beberapa perguruan tinggi, masing-masing UCLA, University of California at Santa Barbara, University of Utah, dan Stanford Research Institute. Ini disusul dengan dibukanya layanan Usenet dan Bitnet yang memungkinkan internet diakses melalui sarana komputer pribadi (PC). Berkutnya, protokol standar TCP/IP mulai diperkenalkan pada tahun 1982, disusul dengan penggunaan sistem DNS (*Domain Name Service*) pada 1984. Di tahun 1986 lahir National Science Foundation Network (NSFNET), yang menghubungkan para periset di seluruh negeri dengan 5 buah pusat super komputer. Jaringan ini kemudian berkembang untuk menghubungkan berbagai jaringan akademis lainnya yang terdiri atas universitas dan konsorsium-konsorsium riset. NSFNET kemudian mulai menggantikan ARPANET sebagai jaringan riset utama di Amerika hingga pada bulan Maret 1990 ARPANET secara resmi dibubarkan. Pada saat NSFNET dibangun, berbagai jaringan internasional didirikan dan dihubungkan ke NSFNET. Australia, negara-negara Skandinavia, Inggris, Perancis, Jerman, Kanada dan Jepang segera bergabung kedalam jaringan ini. Pada awalnya, internet hanya menawarkan layanan berbasis teks, meliputi

remote access, *email/messaging*, maupun diskusi melalui newsgroup (Usenet). Layanan berbasis grafis seperti *World Wide Web* (WWW) saat itu masih belum ada. Yang ada hanyalah layanan yang disebut Gopher yang dalam beberapa hal mirip seperti web yang kita kenal saat ini, kecuali sistem kerjanya yang masih berbasis teks. Kemajuan berarti dicapai pada tahun 1990 ketika World Wide Web mulai dikembangkan oleh CERN (Laboratorium Fisika Partikel di Swiss) berdasarkan proposal yang dibuat oleh Tim Berners-Lee. Namun demikian, WWW browser yang pertama baru lahir dua tahun kemudian, tepatnya pada tahun 1992 dengan nama Viola. Viola diluncurkan oleh Pei Wei dan didistribusikan bersama CERN WWW. Tentu saja web browser yang pertama ini masih sangat sederhana, tidak secanggih browser modern yang kita gunakan saat ini.

Terobosan berarti lainnya terjadi pada 1993 ketika InterNIC didirikan untuk menjalankan layanan pendaftaran domain. Bersamaan dengan itu, Gedung Putih (White House) mulai online di Internet dan pemerintah Amerika Serikat meloloskan National Information Infrastructure Act. Penggunaan internet secara komersial dimulai pada 1994 dipelopori oleh perusahaan Pizza Hut, dan Internet Banking pertama kali diaplikasikan oleh First Virtual. Setahun kemudian, Compuserve, America Online, dan Prodigy mulai memberikan layanan akses ke Internet bagi masyarakat umum.

Sejarah internet Indonesia bermula pada awal tahun 1990-an, saat itu jaringan internet di Indonesia lebih dikenal sebagai paguyuban network, dimana semangat kerjasama, kekeluargaan dan gotong royong sangat hangat dan terasa diantara para pelakunya. Agak berbeda dengan suasana Internet Indonesia pada

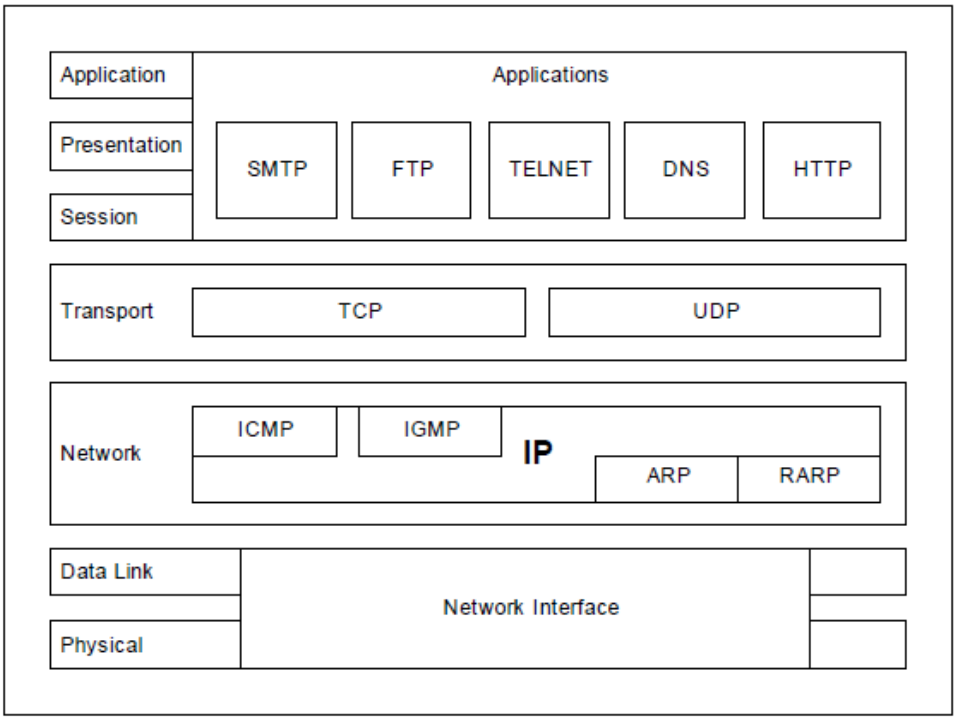
perkembangannya yang terasa lebih komersial dan individual di sebagian aktifitasnya terutama yang melibatkan perdagangan Internet.

Inspirasi tulisan-tulisan awal Internet Indonesia datangnya dari kegiatan di amatir radio, khususnya di Amatir Radio Club (ARC) ITB di tahun 1986. Bermodal pesawat Transceiver HF SSB Kenwood TS430 milik Harya Sudirapratama dengan komputer Apple II milik Onno W. Purbo sekitar belasan anak muda ITB seperti Harya Sudirapratama, J. Tjandra Pramudito, Suryono Adisoemarta bersama Onno W. Purbo berguru pada para senior amatir radio seperti Robby Soebiakto, Achmad Zaini, Yos, di band 40m (7MHz). Teknologi radio paket TCP/IP yang kemudian di adopsi oleh rekan-rekan BPPT, LAPAN, UI, dan ITB yang kemudian menjadi tumpuan PaguyubanNet di tahun 1992-1994.

Di tahun 1989 sampai 1990-an, mahasiswa Indonesia di luar negeri mulai membangun tempat diskusi di internet, salah satu tempat diskusi Indonesia di internet yang pertama berada di *indonesians@janus.berkeley.edu*. Berawal dari mailing list pertama di Janus diskusi-diskusi antar teman-teman mahasiswa Indonesia di luar negeri pemikiran alternatif berserta kesadaran masyarakat ditumbuhkan. Pola mailing list ini ternyata terus berkembang dari sebuah mailing list legendaris di janus, akhirnya menjadi sangat banyak sekali mailing list Indonesia terutama di host oleh *server* di ITB dan egroups.com. Mailing list ini akhirnya menjadi salah satu sarana yang sangat strategis dalam pembangunan komunitas di Internet Indonesia.

2.10 TCP/IP

TCP/IP dikembangkan mengacu pada model *Open System Interconnection* (OSI), dimana, *layer-layer* yang terdapat pada TCP tidak persis sama dengan *layer-layer* yang terdapat pada model OSI. Terdapat empat *layer* pada TCP/IP, yaitu: *network interface*, *network*, *transport* dan *application*. Tiga *layer* pertama pada TCP/IP menyediakan *physical standards*, *network interface*, *internetworking*, dan fungsi *transport*, yang mengacu pada empat *layer* pertama pada model OSI. Tiga *layer* teratas dari model OSI direpresentasikan di model TCP/IP sebagai satu *layer*, yaitu *application layer*.



Gambar 2.22 *TCP/IP dan OSI Model*

2.10.1 Internet Protocol Version 4 (IPv4)

IP merupakan suatu mekanisme transmisi yang digunakan oleh protokol-protokol TCP/IP, dimana IP bersifat *unreliable, connectionless* dan *datagram delivery service*.

Unreliable berarti bahwa protokol IP tidak menjamin *datagram* (Paket yang terdapat di dalam IP *layer*) yang dikirim pasti sampai ke tempat tujuan. Protokol IP hanya berusaha sebaik-baiknya agar paket yang dikirim tersebut sampai ke tujuan. Jika dalam perjalanan, paket tersebut mengalami gangguan seperti jalur putus, kongesti pada *router* atau target *host down*, protokol IP hanya bisa menginformasikan kepada pengirim paket melalui protokol ICMP bahwa terjadi masalah dalam pengiriman paket IP. Jika diinginkan keandalan yang lebih baik, keandalan itu harus disediakan oleh protokol yang berada di atas IP *layer* misalnya TCP dan aplikasi pengguna.

Connectionless berarti bahwa dalam mengirim paket dari tempat asal ke tujuan, baik pihak pengirim dan penerima paket IP sama sekali tidak mengadakan perjanjian terlebih dahulu (*handshake*).

Datagram delivery service berarti bahwa setiap paket yang dikirimkan tidak tergantung pada paket data yang lain. Akibatnya jalur yang ditempuh oleh masing-masing paket data bisa jadi berbeda satu dengan yang lainnya.

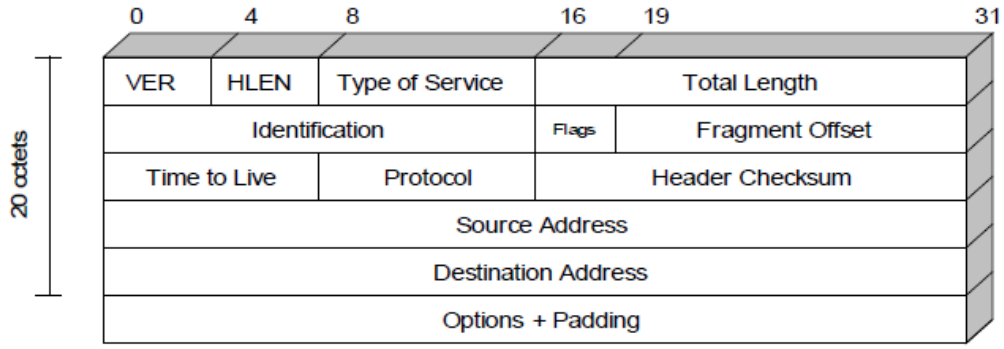
Pada saat ini secara umum internet masih menggunakan IP versi 4, dimana pemakaiannya sudah semakin terbatas mengingat jumlah pengguna internet yang berkembang dengan cepat. Hal ini disebabkan oleh panjang alamat yang dimiliki

IPv4 yaitu 32 bit. Pada gambar 2.23 dibawah ini ditunjukkan format *header* dari IPv4.

Informasi yang terdapat pada *header* IP :

a. *Version* (VER), berisi tentang versi protokol IP yang dipakai.

b. *Header Length* (HLEN), berisi panjang *header* IP bernilai 32 bit.



Gambar 2.23 *IPv4 Header*

c. *Type of Service* (TOS), berisi kualitas *service* cara penanganan paket IP.

d. *Total Length of Datagram*, total panjang datagram IP dalam ukuran *byte*.

e. *Identification*, *Flags*, dan *Fragment Offset*, berisi tentang data yang berhubungan dengan fragmentasi paket.

f. *Time to Live* (TTL), berisi jumlah *router/hop* maksimal yang boleh dilewati paket IP. Setiap kali paket IP melewati *router*, isi *field* akan dikurangi satu. Jika TTL telah habis dan paket belum sampai ke tujuan, paket akan dibuang dan *router* terakhir akan mengirimkan paket ICMP *time exceeded*.

g. *Protocol*, berisi angka yang mengidentifikasikan protokol *layer* atas, yang menggunakan isi data dari paket IP ini.

h. *Header Checksum*, berisi nilai *checksum* yang dihitung dari seluruh *field* dari *header* paket IP. Sebelum dikirimkan, protokol IP terlebih dahulu menghitung *checksum* dari *header* paket IP tersebut untuk nantinya dihitung kembali di sisi penerima. Jika terjadi perbedaan maka paket dianggap rusak dan dibuang.

i. *Source IP Address*, alamat asal/sumber.

j. *Destination IP Address*, alamat tujuan.

k. *Option*, mengkodekan pilihan-pilihan yang diminta oleh pengirim seperti *security label*, *source routing*, *record routing*, dan *time stamping*.

l. *Padding*, digunakan untuk meyakinkan bahwa *header* paket bernilai kelipatan dari 32 bit.

2.11 Pengalamatan IP

Alamat IP merupakan representasi dari 32 bit bilangan *unsigned* biner. Ditampilkan dalam bentuk desimal dengan titik. Contoh 10.252.102.23 merupakan contoh valid dari IP.

2.11.1 Alamat IP (IP *Address*)

Pengalamatan IP dapat di lihat di RFC 1166 – Internet Number. Untuk mengidentifikasi suatu *host* pada internet, maka tiap host diberi IP *address*, atau *internet address*. Apabila *host* tersebut tersambung dengan lebih dari 1 jaringan maka disebut *multi-homed* dimana memiliki 1 IP *address* untuk masing-masing *interface*. IP *Address* terdiri dari:

IP *Address* = <nomer network><nomer host>

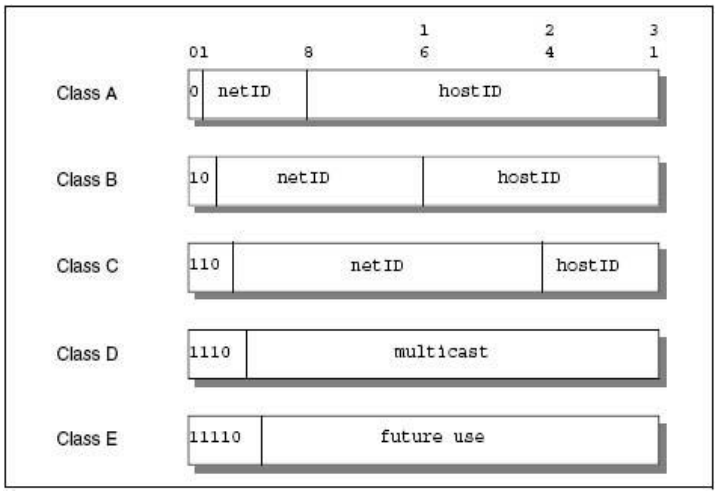
Nomor *network* diatur oleh suatu badan yaitu Regional Internet Registries (RIR), yaitu :

1. American Registry for Internet Number (ARIN), bertanggung jawab untuk daerah Amerika Utara, Amerika Selatan, Karibia, dan bagian sahara dari Afrika.
2. Reseaux IP Europeens (RIPE), bertanggung jawab untuk daerah Eropa, Timur Tengah dan bagian Afrika.
3. Asia Pasific Network Information Center (APNIC), bertanggung jawab untuk daerah Asia Pasific.

IP *address* merupakan 32 bit bilangan biner dimana bisa dituliskan dengan bilangan desimal dengan dibagi menjadi 4 kolom dan dipisahkan dengan titik. Bilangan biner dari IP *address* 128.2.7.9 adalah : 10000000 00000010 00000111 00001001. Penggunaan IP *address* adalah unik, artinya tidak diperbolehkan menggunakan IP *address* yang sama dalam satu jaringan.

2.11.2 Pembagian Kelas Alamat IP (*Class-based* IP *address*)

Bit pertama dari alamat IP memberikan spesifikasi terhadap sisa alamat dari IP. Selain itu juga dapat memisahkan suatu alamat IP dari jaringan. Network. Alamat Network (*network address*) biasa disebut juga sebagai netID, sedangkan untuk alamat host (*host address*) biasa disebut juga sebagai hostID. Ada 5 kelas pembagian IP *address* yaitu:



Gambar 2.24 *Pembagian Kelas pada IP*

Keterangan:

1. Kelas A : Menggunakan 7 bit alamat network dan 24 bit untuk alamat host. Dengan ini memungkinkan adanya 2^7-2 (126) jaringan dengan 2^{24}-2 (16777214) host, atau lebih dari 2 juta alamat.
2. Kelas B : Menggunakan 14 bit alamat network dan 16 bit untuk alamat host. Dengan ini memungkinkan adanya 2^{14}-2 (16382) jaringan dengan 2^{16}-2 (65534) host, atau sekitar 1 juga alamat.
3. Kelas C : Menggunakan 21 bit alamat network dan 8 bit untuk alamat host. Dengan ini memungkin adanya 2^{21}-2 (2097150) jaringan dengan 2^8-2 (254) host, atau sekitar setengah juta alamat.
4. Kelas D : Alamat ini digunakan untuk multicast.
5. Kelas E : Digunakan untuk selanjutnya. Kelas A digunakan untuk jaringan yang memiliki jumlah *host* yang sangat banyak. Sedangkan kelas C digunakan untuk jaringan kecil dengan jumlah *host* tidak sampai 254.

sedangkan untuk jaringan dengan jumlah *host* lebih dari 254 harus menggunakan kelas B.

Pembagian kelas-kelas IP *address* didasarkan pada dua hal antara lain :

1. Network ID, merupakan bagian dari IP *address* yang digunakan untuk menunjukkan jaringan tempat komputer ini berada.
2. *Host* ID, adalah bagian dari IP *address* yang digunakan untuk menunjukkan *workstation, server router*, dan semua *host* TCP / IP lainnya dalam jaringan tersebut. Dalam satu jaringan, *host ID* ini tidak boleh ada yang sama.
3. Pengalamatan IP diorganisasikan ke dalam kelas-kelas dengan pemeriksaan *octet* pertama sebagai berikut:

   a. Kelas A

   Format: 0nnnnnn.hhhhhhhh.hhhhhhhh.hhhhhhhh

   Bit Pertama: 0

   Panjang Net ID: 8 bit

   Panjang Host ID: 24 bit

   Byte Pertama: 0-127

   Jumlah: 126 KelasA (0 dan127 dicadangan)

   Range IP: 1.xxx.xxx.xxx sampai 126.xxx.xxx.xxx

   Jumlah IP: 16.777.214 IP *Address* pada tiap kelas A

   Contoh IP : 113.46.5.6

   Network ID: 113

Host ID: 46.5.6

b. Kelas B

Format: 10nnnnn.nnnnnnnn.hhhhhhhh.hhhhhhhh

Bit Pertama: 10

Panjang Net ID: 16 bit

Panjang Host ID: 16 bit

Byte Pertama: 128-191

Jumlah: 16.184 KelasB

Range IP: 128.0.xxx.xxx sampai 191.255.xxx.xxx

Jumlah IP: 65.532 IP *Address* pada tiap kelas B

Contoh IP : 132.92.121.1

Network ID: 132.92

Host ID: 121.1

c. Kelas C

Format: 110nnnn.nnnnnnnn.nnnnnnnn.hhhhhhhh

Bit Pertama: 110

Panjang Net ID: 24 bit

Panjang Host ID: 8 bit

Byte Pertama: 192-223

Jumlah: 2.097.152 KelasC

Range IP: 192.0.0.xxx sampai223.255.255.xxx

Jumlah IP: 245 IP *Address* pada tiap kelas C

Contoh IP : 222.124.203.53

Network ID: 222.124.203

Host ID: 53

2.12 Linux

Linux adalah sebuah sistem operasi komputer bertipe Unix. Linux merupakan salah satu contoh perangkat lunak bebas dan *open source* karena sebagian besar kode sumbernya (*source code*) dapat secara bebas dimodifikasi digunakan dan didistribusikan kembali oleh semua orang. Penggunaan nama "Linux" berasal dari nama *kernel*nya (*kernel* Linux), yang dibuat tahun 1991 oleh Linus Torvalds.

Sistemnya, peralatan sistem dan pustakanya umumnya berasal dari sistem operasi GNU(*General Public License*), yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi GNU adalah dasar dari munculnya nama alternatif GNU/Linux. Istilah Linux atau GNU/Linux digunakan sebagai rujukan kepada keseluruhan *distro* Linux (Linux *distribution*), yang di dalamnya disertakan program-program lain pendukung sistem operasi. Contoh program tersebut adalah *server* web, bahasa pemrograman, basisdata, tampilan layar (*Desktop Environment*) , dan aplikasi perkantoran (*office suite*) seperti . *Distro* Linux telah mengalami pertumbuhan yang pesat dari segi popularitas, sehingga lebih populer dari 32 versi UNIX yang menggunakan sistem lisensi dan berbayar (*proprietary*) maupun versi UNIX bebas lain yang pada awalnya.

2.12.1 Kelebihan Linux

Beberapa kelebihan dari sistem operasi Linux/UNIX sebagai *server*:

1. Dapat berjalan pada spesifikasi *hardware* yang minimal.
2. *Multi User*, Linux dapat digunakan oleh satu atau lebih orang untuk menggunakan program yang sama atau berbeda dalam suatu mesin yang sama pada saat bersamaan, di terminal yang sama atau berbeda.
3. *Multiconsole*, Dalam satu komputer, pengguna dapat melakukan *login* dengan nama *user* yang sama atau berbeda lebih dari satu kali, tanpa perlu menutup sesi sebelumnya. *Multiconsole* dapat dilakukan pada linux karena linux merupakan *Non-Dedicated Server* yaitu *user* dapat bekerja seperti halnya melalui *client* menggunakan komputer *server* selagi *server* bekerja melayani *client-client* yang ada.
4. Linux yang di khususkan untuk *server* menggunakan perintah (CLI) *Command Line Interface*, sehingga tidak memakan *resource* yang besar untuk menjalankan perintah, dikarenakan pada *server resource* merupakan salah satu aspek yang penting.

2.12.2 Perintah-perintah Dasar Linux

a. alias

Digunakan untuk memberi nama lain dari sebuah perintah. Misalnya bila ingin perintah ls dapat juga dijalankan dengan mengetikkan perintah dir, maka buatlah aliasnya sbb:

```
$ alias dir=ls
```

Kalau akan menampilan tampilan berwarna-warni, dengan perintah berikut:

```
$ alias dir=ls -ar –color:always
```

Untuk melihat perintah-perintah apa saja yang mempunyai nama lain saat itu, cukup ketikkan alias saja (tanpa argumen).

b. cat

Menampilkan isi dari sebuah file di layar. Contoh:

```
$ cat /nama/suatu/file
```

c. chmod

Digunakan untuk menambah dan mengurangi ijin pemakai untuk mengakses file atau direktori. Anda dapat menggunakan sistem *numeric coding* atau sistem *letter coding*. Ada tiga jenis *permission*/perijinan yang dapat diubah yaitu:

1. r untuk *read*,
2. w untuk *write*, dan
3. x untuk *execute*.

Dengan menggunakan *letter coding*, dapat merubah *permission* diatas untuk masing-masing u (*user*), g (*group*), o (*other*) dan a (*all*) dengan hanya memberi tanda plus (+) untuk menambah ijin dan tanda minus (-) untuk mencabut ijin.

Misalnya untuk memberikan ijin baca dan eksekusi file coba1 kepada *owner* dan *group*, perintahnya adalah:

```
$ chmod ug+rx coba1
```

Untuk mencabut ijin-ijin tersebut:

```
$ chmod ug-rx coba1
```

Dengan menggunakan sitem *numeric coding*, *permission* untuk *user*, *group* dan *other* ditentukan dengan menggunakan kombinasi angka-angka, 4, 2 dan 1 dimana 4 (*read*), 2 (*write*) dan 1 (*execute*).

Misalnya untuk memberikan ijin baca(4), tulis(2) dan eksekusi(1) file coba2 kepada *owner*, perintahnya adalah:

```
$ chmod 700 coba2
```

Contoh lain, untuk memberi ijin baca(4) dan tulis(2) file coba3 kepada user, baca(4) saja kepada group dan other, perintahnya adalah:

```
$ chmod 644 coba3
```

d. chown

Merubah user ID (*owner*) sebuah file atau direktori

```
$ chown <user id> <file>
```

e. find

Untuk menemukan dimana letak sebuah file. Perintah ini akan mencari file sesuai dengan kriteria yang ditentukan. Sintaksnya adalah perintah itu sendiri diikuti dengan nama direktori awal pencarian, kemudian nama file (bisa menggunakan *wildcard*, *metacharacters*) dan terakhir menentukan bagaimana

hasil pencarian itu akan ditampilkan. Misalnya akan dicari semua file yang berakhiran .doc di *current* direktori serta tampilkan hasilnya di layar:

```
$ find . -name *.doc -print
```

Contoh hasil:

```
./public/docs/account.doc
./public/docs/balance.doc
./public/docs/statistik/prospek.doc
./public/docs/statistik/presconf.doc
```

f. grep

Global regular expresion parse atau grep adalah perintah untuk mencari file-file yang mengandung teks dengan kriteria yang telah ditentukan.

Format perintah:

```
$ grep <teks> <file>
```

Misalnya akan dicari file-file yang mengandung teks marginal di *current* direktori:

```
$ grep marginal <file>
```

diferent.doc: Catatan: perkataan marginal luas dipergunakan di dalam ilmu ekonomi prob.rtf: oleh fungsi hasil marginal dan fungsi biaya marginal jika fungsi prob.rtf: jika biaya marginal dan hasil marginal diketahui maka biaya total.

g. gzip

ini adalah software kompresi zip versi GNU, fungsinya untuk mengkompresi sebuah file.

```
$ gzip <namafile>
```

h. hostname

Untuk menampilkan host atau *domain name sistem* dan bisa pula digunakan untuk menset nama *host* sistem. Contoh pemakaian:

```
[user@localhost mydirectoryname] $ hostname
localhost.localdomain
```

i. man

Untuk menampilkan *manual page* atau teks yang menjelaskan secara detail bagaimana cara penggunaan sebuah perintah. Perintah ini berguna sekali bila sewaktu-waktu lupa atau tidak mengetahui fungsi dan cara menggunakan sebuah perintah.

```
$ man <perintah>
```

j. mount

Perintah ini akan me-*mount file system* ke suatu direktori atau *mount-point* yang telah ditentukan. Hanya superuser yang bisa menjalankan perintah ini. Untuk melihat *file system* apa saja beserta *mount-point*nya saat itu, ketikkan perintah mount.

```
$ mount
/dev/hda3 on / type ext2 (rw)
none on /proc type proc (rw)
```

```
/dev/hda1 on /dos type vfat (rw)
/dev/hda4 on /usr type ext2 (rw)
none on /dev/pts type devpts (rw,mode=0622)
```

k. tar

Menyimpan dan mengekstrak file dari media seperti *tape drive* atau *hard disk*. File arsip tersebut sering disebut sebagai file tar. Sintaknya sebagai berikut:

```
$ tar <aksi> <option> <file atau direktori>
```

Contoh:

```
$ tar -czvf namaFile.tar.gz /nama/direktori/*
```

Perintah di atas digunakan untuk memasukkan semua isi direktori, lalu dikompres dengan format tar lalu di zip dengan gzip, sehingga menghasilkan sebuah file bernama namafile.tar.gz

```
$ tar -xzvf namaFile.tar.gz
```

l. umount

Adalah kebalikan dari perintah mount, yaitu untuk meng-*unmount file system* dari *mount-point*nya. Setelah perintah ini dijalankan direktori yang menjadi *mount-point* tidak lagi bisa digunakan.

```
# umount <filesystem>
```

m. who

Untuk menampilkan siapa saja yang sedang *login*. Perintah ini akan menampilkan informasi mengenai *login name*, jenis terminal, waktu *login* dan *remote hostname* untuk setiap *user* yang saat itu sedang *login*. Misalnya:

```
$ who
root ttyp0 May 22 11:44
flory ttyp2 May 22 11:59
pooh ttyp3 May 22 12:08
```

2.13 OpenVPN

OpenVPN adalah salah satu aplikasi VPN berbasiskan *open source* untuk membuat koneksi *encrypted tunnels* secara *virtual* dengan menggunakan *authenticate* dengan yang lainnya menggunakan *pre-shared secret key*, *certificates*, atau *username OpenVPN* memiliki beberapa kelebihan untuk membuat VPN *server* seperti berbasiskan *open source*, keamanan, kestabilan, mekanisme *autentikasi* dan *enkripsi*.

OpenVPN memiliki beberapa fitur unggulan yang diantaranya yaitu:

1. **Dukungan *Multi-client***: OpenVPN menawarkan suatu modus koneksi yang khusus, di mana *client-client* TLS-*authenticated* (di-blacklist di CRL) disiapkan dalam bentuk DHCP dengan IP dan *networking* (*tunnel*) data. Pada bagian ini, beberapa *tunnel* (sampai dengan 128 *tunnel*) dapat dikomunikaskan di atas *port* TCP protokol atau UDP yang sama.
2. **Pilihan Push/Pull**: susunan Network dari *client-client* dapat dikendalikan oleh *server*. Setelah susunan suatu *tunnel* sukses, *server* dapat

memberitahukan kepada *client* (baik Windows maupun Linux) untuk menggunakan suatu susunan jaringan yang berbeda dengan segera.

3. **Terdapat antar muka manajemen (Telnet).**

2.14 TCPdump

TCPdump adalah tools yang berfungsi meng-*capture*, membaca atau men-*dump*ing paket yang sedang ditransmisikan melalui jalur TCP. TCPdump diciptakan untuk menolong programmer ataupun administrator dalam menganalisa dan *troubleshooting* aplikasi *networking*. Seperti pisau yang bermata dua (hal ini sering kali disebut-sebut).

TCPDump seringkali digunakan oleh para *cracker* untuk melaksanakan perkerjaannya, karena TCPdump bisa meng-*capture* atau men-*sniff* semua paket yang diterima oleh *network interface*. Sebagai contoh, digunakan *network* yang terdiri dari tiga komputer yang dihubungkan menggunakan hub. Komputer pertama, menggunakan sistem operasi Windows XP dengan IP *address* 192.0.0.1, sedang melakukan koneksi melalui telnet ke komputer kedua yang menggunakan Slackware 8.0 dengan IP *address* 192.168.0.2 dan *host* ketiga komputer Redhat 7.1 dengan IP *address* 192.168.0.3 yang menggunakan *utility* TCPdump. Alasan untuk membedakan *Operating System* yang digunakan adalah untuk menunjukkan bahwa TCP/IP dapat berkomunikasi dengan baik pada dua platform yang berbeda.

Untuk menjalankan TCPdump, ketik perintah tcpdump di *console* pada *host* 192.168.0.3 sebagai *root*. *Output* yang diperlihatkan di bawah ini adalah

output yang bergulir *non-stop*, terus berganti baris tanpa henti hingga menekan Ctrl+C (^C) untuk menghentikan *utility* ini.

```
# tcpdump
tcpdump: listening on eth0
05:22:27.216338 burner.ssh > prime.1035:
P3797249897:3797249949(52) ack 2183278948 win 8576 (DF)
[tos 0x10]
```

Contoh *output* yang berulang diatas menyatakan bahwa salah satu komputer sedang menjalankan ssh *client* untuk koneksi ke *server* ssh di Redhat 7.1 (192.168.0.3) yang mengakibatkan trafik pada *network* (untuk menghasilkan *output* seperti diatas, jalankan ssh *server* pada Redhat 7.1 dan menggunakan ssh *client* untuk melakukan koneksi ke *server* dengan Slackware 8.0 (ini hanya contoh sementara). TCPdump melakukan tugasnya dengan baik. Perintah tcpdump bisa digunakan dengan menggunakan beberapa opsi yang dapat menghasilkan *output* yang lebih spesifik. Manual *pages* dari TCPdump menjelaskan opsi-opsi tersebut secara detil.

Command line berikut ini akan memulai TCPdump dan memperlihatkan *frame Protocol Data Unit* (PDU) yang memindahkan data dari pengirim ke tujuan di dalam *network*, menurut OSI model, PDU yang berada di *datalink layer* disebut sebagai *frame*, PDU yang berada di *network layer* disebut paket, dan PDU yang berada di *transport layer* disebut segmen) yang mengandung IP *address* 192.168.0.2 (mesin Slackware).

```
# tcpdump host 192.168.0.2
tcpdump: listening on eth0
19:16:04.817889 arp who-has tssoss tell prime
19:16:04.818025 arp reply tssoss is-at 0:a0:c9:20:5b:fe
```

```
19:16:04.818182 prime.1219 > tssoss.telnet:
S2506660519:2506660519(0) win 16384 <mss
1460,nop,nop,sackOK> (DF).
```

Perintah berikut ini hanya akan mengeluarkan *output* pada *frame* yang mengandung spesifik IP *address* dan spesifik *port*. Opsi **-nn** akan membuat *disable* translasi nama dan *port*.

```
# tcpdump -nn host 192.168.0.2 and port 23
tcpdump: listening on eth0
19:20:00.804501 192.168.0.1.1221 > 192.168.0.2.23:
S2565655403:2565655403(0) win 16384 <mss
1460,nop,nop,sackOK> (DF)
```

Jika pada perintah di atas ditambahkan opsi **–t**, *display output* akan ditampilkan tanpa keterangan waktu. Sedangkan penambahan opsi **-e** akan menampilkan layer kedua dari informasi *datalink*. Keterangan MAC *address* baik milik mesin pengirim maupun mesin tujuan adalah salah satu contoh dari informasi *datalink*.

```
# tcpdump -nne host 192.168.0.2 and port 23
tcpdump: listening on eth0
19:30:13.024247 0:5:5d:f4:9e:1f 0:a0:c9:20:5b:fe 0800 62:
192.168.0.1.1223 > 192.168.0.2.23:
S2718633695:2718633695(0) win 16384 <mss
1460,nop,nop,sackOK> (DF)
```

TCPDump ini juga mempunyai kemampuan untuk menganalisa PDU yang memulai dan mengakhiri suatu koneksi TCP/IP. TCP mempunyai mekanisme

khusus untuk membuka dan menutup suatu koneksi. Untuk menjamin bahwa *startup* dan *shutdown* koneksi benar-benar terjadi, TCP menggunakan metode dimana ada tiga pesan yang ditukar, metode ini sering juga disebut *three-way-handshake*.

Berikut adalah urutan untuk memulai (*startup*) suatu koneksi:

1. *Host* peminta koneksi 192.168.0.1 mengirimkan bendera sinkronisasi/*synchronization flag* (SYN) dalam segmen TCP untuk membuat suatu koneksi.
2. *Host* penerima permintaan 192.168.0.2 menerima SYN *flag* dan mengirimkan bendera pernyataan/*acknowledgment flag* (ACK) kepada *host* peminta koneksi.
3. *Host* peminta koneksi akan menerima ACK *flag* milik 192.168.0.2 sebagai SYN *flag* dan mengembalikan SYN *flag* yang diterima sebagai ACK *flag* nya sendiri kepada 192.168.0.2.

Proses *handshake* yang sama juga terjadi saat menutup (*shutdown*) koneksi, tetapi bendera atau *flag* yang digunakan adalah bendera selesai atau *finish flag* (FIN). Untuk melakukan koneksi, *host* yang meminta koneksi akan membuat suatu segmen (PDU di *transport layer*) yang mengandung alamat IP dan nomor *port* dari *host* yang dituju. Segmen tersebut mengandung SYN *flag* dan *host* peminta akan menginisialisas urutan nomor/*sequence number*. Sebelum data dikirim, data tersebut akan dipecah menjadi bagian-bagian kecil, dan fungsi

sequence number adalah untuk membantu perakitan kembali data yang telah dipecah di *host* tujuan.

Pada *output* TCPdump, dapat dilihat data-data tersebut :

```
#tcpdump -nn host 192.168.0.2 and port 23
20:06:32.845356 192.168.0.1.1249 > 192.168.0.2.23:
S 3263977215:3263977215(0) win 16384 <mss
1460,nop,nop,sackOK> (DF).
```

Host yang diminta akan merespon permintaan dengan mengirimkan SYN *flag* dan *sequence number* miliknya sendiri kepada host peminta, respon ini juga mengandung ACK *flag* (3263977215 (SYN *flag* milik peminta) + 1) untuk menyatakan bahwa permintaan SYN *flag* peminta sudah diterima.

```
20:06:32.845725 192.168.0.2.23 > 192.168.0.1.1249: S
48495364:48495364(0) ack 3263977216 win 32120 <mss
1460,nop,nop,sackOK> (DF).
```

Kemudian *host* peminta akan merespon SYN *flag* yang dikirim oleh *host* yang diminta dengan cara mengirimkan segmen lain yang mengandung **.** *flag* dan ACK *flag* 20:06:32.845921 192.168.0.1.1249 > 192.168.0.2.23: . ack 1 win 17520 (DF)

flag yang ditampilkan oleh *output* TCPdump ada 5, yaitu:

S: SYN (*Synchronize sequence numbers - Connection establishment*)

F: FIN (*Ending of sending by sender - Connection termination*)

R: RST (*Reset connection*)

P: PSH (*Push data*)

.: (No flag is set)

Untuk menutup koneksi segment yang mengandung SYN *flag* akan dikirimkan dari *host* 192.168.0.2 ke *host* yang membuka/meminta koneksi.

2.15 NMAP

Nmap (*Network Mapper*) adalah sebuah aplikasi atau *tool* yang berfungsi untuk melakukan *port scanning*, eksplorasi dan audit keamanan jaringan. Nmap dirancang untuk memeriksa jaringan besar secara cepat, meskipun dapat pula bekerja terhadap host tunggal. Nmap menggunakan paket IP raw dalam cara yang canggih untuk menentukan host mana saja yang tersedia pada jaringan, layanan (nama aplikasi dan versi) apa yang diberikan, sistem operasi (dan versinya) apa yang digunakan, apa jenis *firewall*/*filter* paket yang digunakan, dan sejumlah karakteristik lainnya. Meskipun Nmap umumnya digunakan untuk audit keamanan, 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.

Output Nmap adalah sebuah daftar target yang diperiksa, dengan informasi tambahannya tergantung pada opsi yang digunakan. Hal kunci di antara informasi itu adalah "tabel *port*". Tabel tersebut berisi daftar angka *port* dan protokol, nama layanan, dan status. Statusnya adalah `terbuka (open)`, `difilter (filtered)`, `tertutup (closed)`, atau `tidak difilter (unfiltered)`. `Terbuka` berarti bahwa aplikasi pada mesin target sedang mendengarkan (*listening*) untuk koneksi/paket pada *port* tersebut. `Difilter` berarti bahwa sebuah *firewall*, filter,

atau penghalang jaringan lainnya memblokir *port* sehingga Nmap tidak dapat mengetahui apakah ia `terbuka` atau `tertutup`. `Tertutup` *port* tidak memiliki aplikasi yang sedang mendengarkan, meskipun mereka dapat terbuka kapanpun. *Port* digolongkan sebagai `tidak difilter` ketika mereka menanggapi *probe* Nmap, namun Nmap tidak dapat menentukan apakah mereka terbuka atau tertutup. Nmap melaporkan kombinasi status `open|filtered` dan `closed|filtered` ketika ia tidak dapat menentukan status manakah yang menggambarkan sebuah *port*. Tabel *port* mungkin juga menyertakan detil versi software ketika diminta melakukan pemeriksaan versi. Ketika sebuah pemeriksaan protokol IP diminta (`-sO`), Nmap memberikan informasi pada protokol IP yang didukung *port-port* yang merespon.

2.16 Wireshark

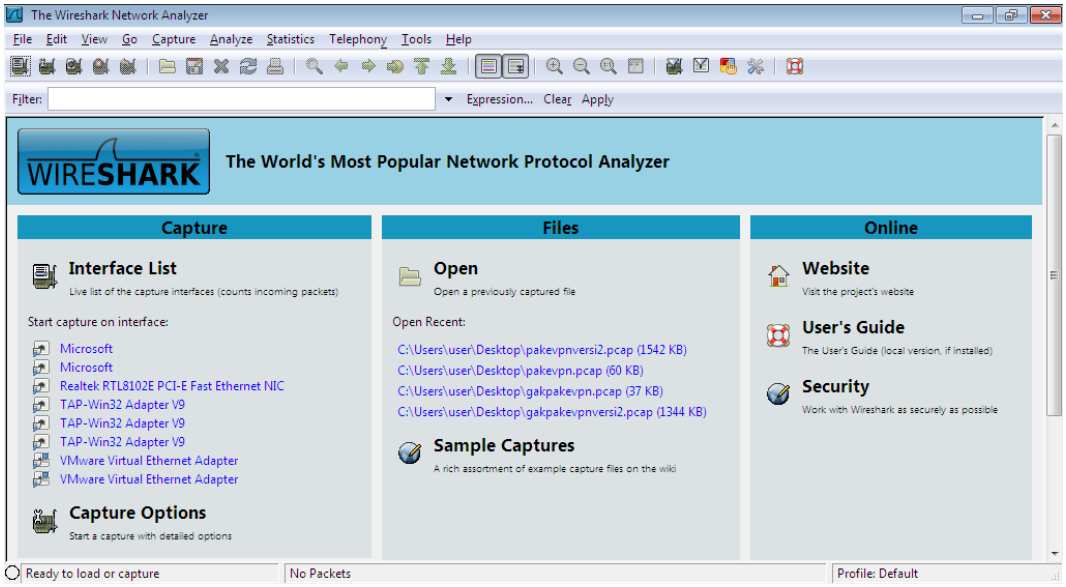
Wireshark adalah program yang berfungsi untuk mengetahui kejadian yang terjadi pada saat kita melakukan interaksi pada suatu jaringan komputer. Wireshark merupakan salah satu *network analysis tool*, atau disebut juga dengan *protocol analysis tool* atau *packet sniffer*. Wireshark dapat digunakan untuk *troubleshooting* jaringan, analisis, pengembangan software dan protokol, serta untuk keperluan edukasi.

Wireshark mampu menangkap paket-paket data atau informasi yang berjalan dalam suatu jaringan yang dimonitor. Hampir semua jenis paket informasi dalam berbagai format protokol dapat ditangkap dan dianalisa.

Packet sniffer sendiri diartikan sebagai sebuah program atau tool yang memiliki kemampuan untuk ‘mencegat’ dan melakukan pencatatan terhadap *traffic* data dalam jaringan. Selama terjadi aliran data dalam, *packet sniffer* dapat menangkap *protocol data unit* (PDU), melakukan dekoding serta melakukan analisis terhadap isi paket berdasarkan spesifikasi RFC atau spesifikasi-spesifikasi yang lain.

Wireshark sebagai salah satu packet *sniffer* diprogram sedemikian rupa untuk mengenali berbagai macam protokol jaringan. Wireshark mampu menampilkan hasil enkapsulasi dan *field* yang ada dalam PDU.

Berikut adalah tampilan utama Wireshark:



Gambar 2.25 *Tampilan Utama Aplikasi Wireshark*