BAB 2
TINJAUAN PUSTAKA

2.1 Jaringan Komputer

Jaringan komputer adalah sebuah kumpulan perangkat-perangkat komputer dan jaringan yang saling terhubung. Informasi dan data bergerak melalui kabel-kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar informasi, mencetak pada printer yang sama dan bersama-sama menggunakan *hardware*/*software*. Tiap komputer, printer atau perangkat keras yang terhubung dengan jaringan disebut *node*. Sebuah jaringan komputer dapat memiliki dua, puluhan, ribuan atau bahkan jutaan *node*.

2.1.1 Jenis Jaringan Komputer

Ada 3 macam jenis jaringan, yaitu :

1. *Local Area Network* (LAN)

   LAN adalah jaringan yang dibatasi oleh area yang relatif kecil, umumnya dibatasi oleh area lingkungan seperti sebuah perkantoran di sebuah gedung atau sebuah sekolah dan tidak jauh dari sekitar 1 km persegi.
2. *Metropolitan Area Network* (MAN)

   MAN 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.
3. *Wide Area Network* (WAN)

   WAN adalah jaringan yang lingkupnya sudah menggunakan media satelit atau kabel bawah laut, sebagai contoh keseluruhan jaringan bank yang ada di Indonesia atau yang ada di negara-negara lain.

2.1.2 Arsitektur Jaringan Komputer

Arsitektur sebuah jaringan komputer dibedakan menjadi arsitektur fisik dan arsitektur logik. Arsitektur fisik berkaitan dengan susunan fisik sebuah jaringan komputer, bisa juga disebut dengan topologi fisik jaringan yaitu menjelaskan hubungan perkabelan dan lokasi *node* (simpul) atau *workstation*. Sedangkan arsitektur logik berkaitan dengan logika hubungan masing-masing komputer dalam jaringan atau menjelaskan aliran data dari satu *user* ke *user* lainnya dalam jaringan.

Topologi adalah bentuk hubungan dari suatu jaringan (*map of network*). Topologi fisik dari sebuah jaringan adalah merujuk pada konfigurasi kabel, komputer, dan perangkat lainnya. Terdapat tiga topologi fisik dasar yaitu: topologi *bus,*topologi *star,* dan topologi *ring*.

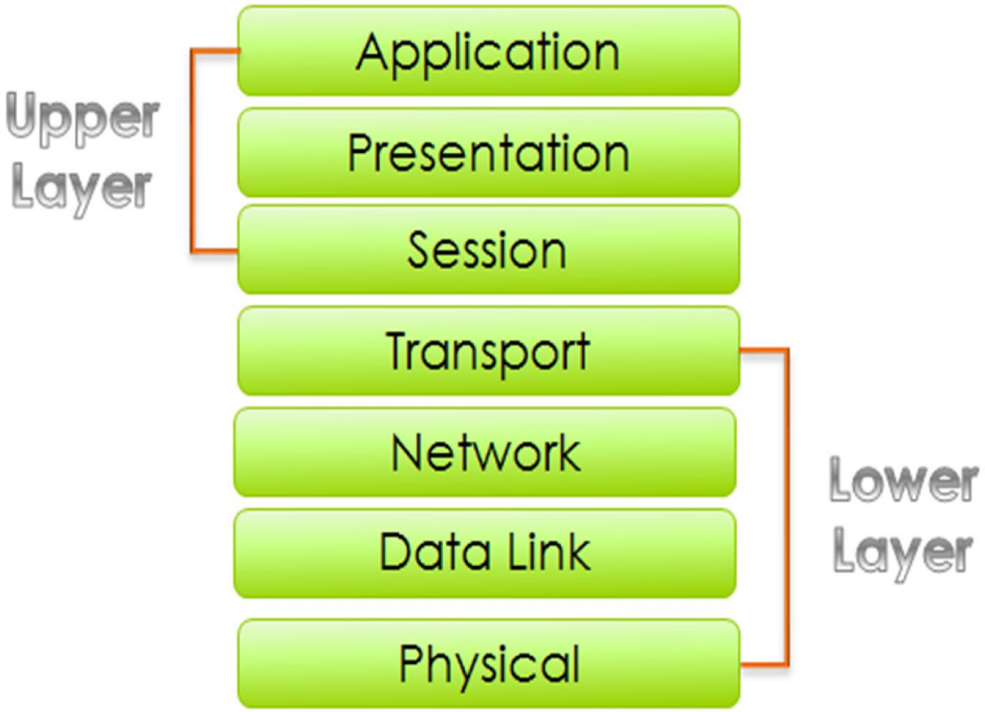
2.2 Protokol Jaringan Komputer

Pada saat *network* baru muncul, kebanyakan komputer hanya dapat berkomunikasi dengan komputer yang dibuat oleh perusahaan yang sama. Sebagai contoh, perusahaan harus menggunakan seluruh solusi dari DECnet atau seluruh solusi dari IBM, tapi tidak bisa kedua – duanya. Pada akhir tahun 1970, *International Organization for Standarization* (ISO) membuat model referensi *Open System Interconnection* (OSI) sebagai solusi untuk mengatasi masalah kompatibilitas ini. Model OSI dimaksudkan untuk membantu para *vendor* (vendor di sini adalah perusahaan pembuat perangkat keras atau pembuat perangkat lunak) agar bisa membuat alat – alat dan perangkat lunak yang bisa saling bekerja sama, dalam bentuk protokol – protokol sehingga *network* dengan *vendor – vendor* yang berbeda bisa saling bekerja sama.

2.2.1 Model Referensi OSI

The International Organization for Standarization (ISO) mengembangkan model referensi *Open System Interconnection* (OSI) sebagai pemandu untuk mendefinisikan sekumpulan protokol terbuka. Walaupun minat pada protokol OSI telah berkurang, model referensi OSI tetap merupakan standar yang paling umum

untuk menggambarkan dan membandingkan rangkaian protokol. Model OSI memiliki tujuh *layer* (lapisan).



Gambar 2.1 OSI Layer

a. *Physical Layer* (Lapisan Fisik)

Berfungsi untuk mendefinisikan media transmisi jaringan, metode pensinyalan, sinkronisasi bit, arsitektur jaringan (seperti halnya *Ethernet* atau *Token Ring*), topologi jaringan dan pengabelan. Selain itu, *level* ini juga mendefinisikan bagaimana *Network Interface Card* (NIC) dapat berinteraksi dengan media kabel atau radio[1].

b. *Data Link Layer* (Lapisan Data Link)

Befungsi untuk menentukan bagaimana bit-bit data dikelompokkan menjadi format yang disebut sebagai *frame*. Selain itu, pada *level* ini terjadi koreksi kesalahan, *flow control*, pengalamatan perangkat keras (seperti halnya Media *Access Control Address* (MAC Address)) dan menentukan bagaimana perangkat-

perangkat jaringan seperti *hub*, *bridge*, *repeater*, dan *switch layer* 2 beroperasi. Spesifikasi IEEE 802, membagi *level* ini menjadi dua *level* anak, yaitu lapisan *Logical Link Control* (LLC) dan lapisan *Media Access Control* (MAC)[1].

c. *Network Layer* (Lapisan Jaringan)

Network layer berfungsi untuk pengendalian operasi *subnet* dengan meneruskan paket-paket dari satu *node* ke *node* lain dalam jaringan. Masalah desain yang penting adalah bagaimana caranya menentukan *route* pengiriman paket dari sumber ke tujuannya. *Route* dapat didasarkan pada tabel statik yang dihubungkan ke *network*. *Route* juga dapat ditentukan pada saat awal percakapan misalnya *session terminal*. *Route* juga sangat dinamik, dapat berbeda bagi setiap paketnya, dan karena itu, *route* pengiriman sebuah paket tergantung beban jaringan saat itu. Bila pada saat yang sama dalam sebuah *subnet* terdapat terlalu banyak paket, maka ada kemungkinan paket-paket tersebut tiba pada saat yang bersamaan. Hal ini dapat menyebabkan terjadinya *bottleneck* (penyempitan di bagian ujung). Pengendalian kemacetan seperti itu juga merupakan tugas *network layer*[2].

d. *Transport Layer* (Lapisan Transport)

Fungsi dasar *transport layer* adalah menerima data dari *session layer*, memecah data menjadi bagian-bagian yang lebih kecil bila perlu, meneruskan data ke *network layer*, dan menjamin bahwa semua potongan data tersebut bisa tiba di sisi lainnya dengan benar. Selain itu, semua hal tersebut harus dilaksanakan secara efisien, dan bertujuan dapat melindungi layer-layer bagian atas dari perubahan teknologi *hardware* yang tidak dapat dihindari.

Dalam keadaan normal, *transport layer* membuat koneksi jaringan yang berbeda bagi setiap koneksi transport yang diperlukan oleh *session layer*. Bila koneksi transport memerlukan *throughput* yang tinggi, maka *transport layer* dapat membuat koneksi jaringan yang banyak. *Transport layer* membagi-bagi pengiriman data ke sejumlah jaringan untuk meningkatkan *throughput*[2].

e. *Session Layer* (Lapisan Sesi)

Session layer mengijinkan para pengguna untuk menetapkan *session* dengan pengguna lainnya. Layer ini membuka, mengatur dan menutup suatu *session* antara aplikasi-aplikasi. Sebuah *session* selain memungkinkan transport data biasa, seperti yang dilakukan oleh *transport layer*, juga menyediakan layanan yang istimewa untuk aplikasi-aplikasi tertentu. Sebuah *session* digunakan untuk memungkinkan seseorang pengguna *log* ke *remote timesharing system* atau untuk memindahkan *file* dari satu mesin ke mesin lainnya.

Sebuah layanan *session layer* yang lain adalah untuk melaksanakan pengendalian dialog. *Session* dapat memungkinkan lalu lintas bergerak dalam bentuk dua arah pada suatu saat, atau hanya satu arah saja. Jika pada suatu saat lalu lintas hanya satu arah (analog dengan rel kereta api tunggal), *session layer* membantu untuk menentukan giliran yang berhak menggunakan saluran. Layanan *session* di atas disebut manajemen *token*. Untuk sebagian protokol, adalah penting untuk memastikan bahwa kedua pihak yang bersangkutan tidak melakukan operasi pada saat yang sama. Untuk mengatur aktivitas ini, *session layer* menyediakan *token-token* yang dapat digilirkan. Hanya pihak yang memegang *token* yang diijinkan melakukan operasi kritis[2].

f. *Presentation Layer* (Lapisan Presentasi)

Presentation layer melakukan fungsi-fungsi tertentu yang diminta untuk menjamin penemuan sebuah penyelesaian umum bagi masalah tertentu. Selain memberikan sarana-sarana pelayanan untuk konversi, format dan enkripsi data, *presentation* layer juga bekerja dengan *file* berformat ASCII, EBCDIC, JPEG, MPEG, TIFF, PICT, MIDI, dan Quick Time.

Presentation layer tidak mengijinkan pengguna untuk menyelesaikan sendiri suatu masalah. Tidak seperti layer-layer di bawahnya yang hanya melakukan pemindahan bit dari satu tempat ke tempat lainnya, *presentation layer* memperhatikan *syntax* dan semantik informasi yang dikirimkan.

Satu contoh layanan *presentation* adalah *encoding* data. Kebanyakan pengguna tidak memindahkan *string* bit biner yang *random*. Para pengguna saling

bertukar data seperti nama orang, tanggal, jumlah uang, dan tagihan. *Item-item* tersebut dinyatakan dalam bentuk *string* karakter, bilangan *integer*, bilangan *floating point*, struktur data yang dibentuk dari beberapa *item* yang lebih sederhana. Terdapat perbedaan antara satu komputer dengan komputer lainnya dalam memberi kode untuk menyatakan *string* karakter contohnya ASCII dan Unicode, integer contohnya komplemen satu dan komplemen dua dan lain sebagainya.

Untuk memungkinkan dua buah komputer yang memiliki *presentation* yang berbeda untuk dapat berkomunikasi, struktur data yang akan dipertukarkan dapat dinyatakan dengan cara abstrak, sesuai dengan ”*encoding standard*” yang akan digunakan “pada saluran”. *Presentation layer* mengatur data struktur abstrak ini dan mengkonversi dari *representation* yang digunakan pada sebuah komputer menjadi ”*representation standard*” jaringan, dan sebaliknya[2].

g. *Application Layer* (Lapisan Aplikasi)

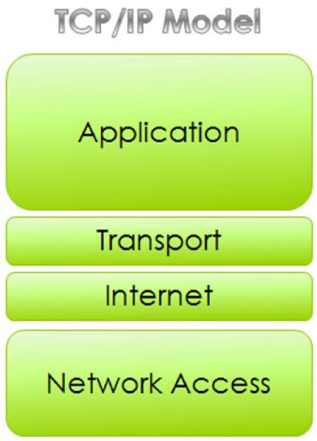
Lapisan ini bertugas memberikan sarana pelayanan langsung ke *user*, yang berupa aplikasi-aplikasi dan mengadakan komunikasi dari program ke program. Jika kita mencari suatu *file* dari *file server* untuk digunakan sebagai aplikasi pengolah kata, maka proses ini bekerja melalui layer ini. Demikian pula jika kita mengirimkan e-mail, *browse* ke internet, *chatting*, membuka telnet *session*, atau menjalankan FTP, maka semua proses tersebut dilaksanakan di layer ini.

Application layer terdiri dari bermacam-macam protokol. Misalnya terdapat ratusan jenis terminal yang tidak kompatibel di seluruh dunia, kemudian kita memerlukan aplikasi yang diharapkan bekerja pada jaringan dengan bermacam-macam terminal, yang masing-masing memiliki *layout* layar yang berlainan, mempunyai cara urutan penekanan tombol yang berbeda untuk penyisipan dan penghapusan teks, memindahkan sensor dan sebagainya. Suatu cara untuk mengatasi masalah seperti di atas adalah dengan menentukan terminal *virtual* jaringan abstrak, sehingga editor dan program-program lainnya dapat ditulis agar saling bersesuaian. Untuk menangani setiap jenis terminal, satu bagian *software* harus ditulis untuk memetakan fungsi terminal *virtual* jaringan ke terminal

sebenarnya. Fungsi lainnya adalah pemindahan file. Sistem file yang satu dengan lainnya memiliki konvensi penamaan yang berbeda, cara menyatakan baris-baris teks yang berbeda, dan sebagainya. Perpindahan *file* dari sebuah sistem ke sistem lainnya yang berbeda memerlukan penanganan untuk mengatasi adanya ketidak-sesuaian ini. Tugas tersebut juga merupakan pekerjaan *application* layer, seperti pada surat elektronik, *remote job entry*, *directory lookup*, dan berbagai fasilitas bertujuan umum dan fasilitas bertujuan khusus lainnya[2].

2.2.2 Protokol TCP/IP

TCP/IP adalah sekumpulan protokol yang terdapat di dalam jaringan komputer (*network*) yang digunakan untuk berkomunikasi atau bertukar data antar komputer.



Gambar 2.2 TCP/IP Model

TCP/IP merupakan standar protokol pada jaringan internet yang menghubungkan banyak komputer yang berbeda jenis mesin maupun system operasinya agar dapat berinteraksi satu sama lain[3].

2.2.2.1 Layanan TCP/IP

Berikut ini adalah layanan "tradisional" yang dilakukan TCP/IP:

1. Pengiriman *file* (*file transfer*). *File Transfer Protocol* (FTP) memungkinkan pengguna komputer yang satu untuk dapat mengirim

ataupun menerima *file* ke komputer jaringan. Karena masalah keamanan data, maka FTP seringkali memerlukan nama pengguna (*username*) dan *password*, meskipun banyak juga FTP yang dapat diakses melalui *anonymous*, alias tidak ber-*password*.

2. *Remote Login*. *Network Terminal Protocol* (telnet) memungkinkan pengguna komputer dapat melakukan *login* ke dalam suatu komputer di dalam suatu jaringan. Jadi hal ini berarti bahwa pengguna menggunakan komputernya sebagai perpanjangan tangan dari komputer jaringan tersebut.
3. *Computer Mail* digunakan untuk menerapkan sistem e-mail (*electronic mail*).
4. *Network File System* (NFS). Pelayanan akses *file-file* jarak jauh yang memungkinkan klien-klien untuk mengakses *file-file* pada komputer jaringan jarak jauh walaupun *file* tersebut disimpan secara lokal.
5. *Remote Execution* Memungkinkan pengguna komputer untuk menjalankan suatu program dari komputer yang berbeda. Biasanya berguna jika pengguna menggunakan komputer yang terbatas, sedangkan ia memerlukan sumber yang banyak dalam suatu sistem komputer. Ada beberapa jenis *remote execution*, ada yang berupa perintah-perintah dasar saja, yaitu yang dapat dijalankan dalam system komputer yang sama dan ada pula yang menggunakan "*procedure remote call system*", yang memungkinkan program untuk memanggil *subroutine* yang akan dijalankan di system komputer yang berbeda. (sebagai contoh dalam Berkeley UNIX ada perintah "rsh" dan "rexec").

2.2.2.2 Arsitektur TCP/IP

Protokol-protokol TCP/IP memberikan dukungan jaringan untuk menghubungkan seluruh tempat dan *host* serta mengikuti aturan standar tentang bagaimana komputer-komputer berkomunikasi dan bagaimana jaringan-jaringan diinterkoneksikan. Protokol-protokol TCP/IP mengikuti model konsep empat

layer yang dikenal sebagai model *Department of Defense* (DOD): *Application*, *Transport*, *Internet*, dan *Network Interface*[4].

1. *Layer Application*

   Layer Application terletak di atas pada model TCP/IP konsep empat layer dan merupakan tempat program *software* memperoleh akses ke jaringan. Layer ini sesuai dengan layer *Session*, layer *Presentation*, dan layer *Application* pada model OSI. Beberapa layanan dan utiliti TCP/IP beroperasi pada *Layer Application*. Layanan dan utiliti tersebut meliputi:

   a. *HyperText Transfer Protocol* (HTTP). HTTP adalah protocol yang dipakai untuk mayoritas komunikasi *World Wide Web*. Windows 2003 menghadirkan Internet Explorer sebagai *client* HTTP dan *Internet Information Services* (IIS) sebagai *server* HTTP.

   b. *File Transfer Protocol* (FTP). FTP adalah suatu layanan Internet yang mentransfer *file-file* dari satu komputer ke komputer lain. Internet Explorer dan FTP utiliti yang termasuk *command-line* bertindak sebagai *client* FTP. IIS menyediakan *server* FTP.

   c. *Simple Mail Transfer Protocol* (SMTP). SMTP merupakan suatu protokol yang dipakai *server mail* untuk mentransfer e-mail. IIS dapat mengirimkan pesan-pesan dengan memakai protokol SMTP.

   d. Telnet. Telnet adalah suatu protocol yang menyamai terminal yang dapat dipakai untuk *login* ke *host* jaringan yang jauh. Telnet menawarkan para pemakai suatu kapabilitas dalam mengoperasikan program- program secara jauh dan memudahkan administrasi yang jauh. Telnet secara praktis memang disediakan untuk semua sistem operasi dan mengurangi integrasi dalam lingkungan jaringan yang heterogen. Windows 2003 memberikan baik *server* maupun *client* Telnet.

   e. *Domain Name System* (DNS). DNS merupakan seperangkat protocol dan layanan pada suatu jaringan TCP/IP yang mengizinkan para

pemakai jaringan untuk mempergunakan nama-nama hirerarki yang sudah dikenal ketika meletakkan *host* ketimbang harus mengingat dan memakai alamat IP-nya. DNS sangat banyak dipakai di internet dan pada kebanyakan perusahaan pribadi dewasa ini.

f. *Simple Network Management Protocol* (SNMP). SNMP memungkinkan untuk mengelola *node* jaringan seperti *server*, *workstation*, *router*, *bridge*, dan *hub* dari *host* sentral. SNMP dapat dipakai untuk mengonfigurasi *device* yang jauh, memantau unjuk kerja jaringan, mendeteksi kesalahan jaringan atau akses yang tidak cocok, dan mengaudit pemakaian jaringan.

2. Network Application API

TCP/IP Microsoft menyediakan dua *interface* bagi aplikasi-aplikasi jaringan untuk memakai layanan-layanan *stack protocol* TCP/IP:

a. WinSock. Implementasi Windows 2003 sangat banyak memakai Socket API (*application programming interface*). Socket API adalah mekanisme standar untuk mengakses layanan-layanan datagram dan session pada TCP/IP.

b. NetBIOS. Suatu API standar yang dipakai sebagai suatu mekanisme *inter-process communication* (IPC) di lingkungan Windows. Meskipun NetBIOS dapat dipakai untuk menyediakan koneksi standar ke protokol-protokol yang mendukung layanan-layanan penyampaian pesan dan penamaan NetBIOS, misalnya TCP/IP dan NetBEUI, namun NetBIOS disertakan di dalam Windows 2003 terutama sekali untuk mendukung aplikasi-aplikasi peninggalan.

3. Layer Transport

Protokol-protokol transport menghadirkan sesi komunikasi di antara komputer-komputer dan menentukan tipe layanan transport entah berorientasi koneksi (TCP) ataupun berorientasi datagram yang tanpa koneksi (UDP). TCP menyajikan komunikasi berorientasi koneksi yang

dapat diandalkan untuk aplikasi-aplikasi yang secara khusus mentransfer sejumlah besar data pada suatu waktu. TCP juga dipakai untuk aplikasi-aplikasi yang memerlukan pengakuan bagi data yang diterima. Namun demikian, UDP menyediakan komunikasi tanpa koneksi dan tidak menjamin mampu mengirimkan paket. Aplikasi-aplikasi yang memakai UDP secara khusus mentransfer sejumlah kecil data pada suatu waktu. Pengiriman data yang dapat diandalkan merupakan tanggung jawab aplikasi. *Layer Transport* pada model DOD sesuai dengan *Layer Transport* pada model OSI.

4. Layer Internet

Protokol-protokol Internet meng-*encapsulate* paket-paket menjadi datagram internet dan mengoperasikan semua algoritma *routing* yang diperlukan. Fungsi-fungsi *routing* yang dikerjakan layer internet memang dibutuhkan untuk memungkinkan *host* meng-*interoperate* dengan jaringan-jaringan lain. Layer internet sesuai dengan *layer network* pada model OSI. Lima protokol yang diimplementasikan pada layer ini:

a. *Address Resolution Protocol* (ARP), yang menentukan alamat *hardware* pada *host*.

b. *Reverse Address Resolution Protocol* (RARP), yang menyediakan resolusi alamat kebalikan pada *host* yang menerima.

c. *Internet Control Message Protocol* (ICMP), yang mengirimkan pesan-pesan kesalahan ke IP ketika terjadi masalah.

d. *Internet Group Management Protocol* (IGMP), yang menginformasikan *router* tentang ketersediaan anggota-anggota pada grup *multicast*.

e. *Internet Protocol* (IP), yang mengalamatkan dan me-*route* paket-paket.

5. Layer *Interface Network*

Model yang paling mendasar adalah layer *interface network*. Masing-masing tipe *local area network* (LAN), tipe *metropolitan area network*

(MAN), tipe WAN, dan tipe *dial*-up, misalnya *Ethernet*, *Token Ring*, *Fiber Distributed Data Interface* (FDDI), dan ARCnet, mempunyai persyaratan yang berbeda untuk pengkabelan, pembuatan sinyal, dan penyandian data. *Layer interface network* menetapkan persyaratan yang ekuivalen dengan layer *data link* dan layer *physical* pada model OSI. Layer *interface network* bertanggung jawab untuk mengirimkan dan menerima *frame*, yang merupakan paket-paket informasi yang ditransmisikan pada suatu jaringan sebagai suatu unit tunggal. Layer *interface network* meletakkan *frame* pada jaringan dan mengeluarkan *frame* dari jaringan.

2.3 Lapisan Pada Protokol *Network*

Pada sebuah jaringan komputer terdapat beberapa lapisan protokol yang digunakan sebagai format standar untuk berkomunikasi antar komputer.

2.3.1 IP Layer

Pada IP layer, paket IP terdiri dari dua bagian yaitu IP *header* dan IP *body*, untuk lebih jelasnya lihat pada Gambar 2.1.



Gambar 2.1 IP *Header*

Dilihat dari paket filter, IP *header* berisi empat informasi, yaitu :

1. IP *Source address* : memiliki panjang 4 byte seperti 192.168.168.3
2. IP *Destination* : memiliki panjang 4 byte seperti 192.168.168.2
3. IP *Protocol Type* : merupakan identitas dari IP *body* apakah paket tersebut berupa paket TCP atau paket UDP, paket ICMP, atau paket lainnya.
4. IP *options field*

2.3.2 TCP Layer

Pada TCP layer paket dibagi menjadi dua bagian yaitu TCP *header* dan TCP *body*. TCP header terdiri dari tiga infomasi yaitu :

1. TCP *source port* : memiliki panjang dua byte yang khusus yang akan diproses pada sistem *client* atau *server* yang dikirim.
2. TCP *destination port* : memiliki panjang dua byte yang akan diproses pada sistem *client* atau *server* yang diterima.
3. TCP *flag field* : berisi *flag* yang digunakan untuk menandai bentuk atau tipe dari paket.

2.3.3 IP (*Internet Protocol*)

Internet Protocol memiliki beberapa lapisan bawah yang berbeda seperti *ethernet*, *token ring*, PPP, dan lain-lain. IP layer juga memiliki beberapa lapisan atas yang berbeda seperti TCP, UDP, dan ICMP.

2.3.3.1 IP Protocol Lapisan Atas

Internet protocol (IP) merupakan dasar dari semua protokol yang berbeda. Bentuk yang paling umum adalah TCP, UDP dan ICMP.

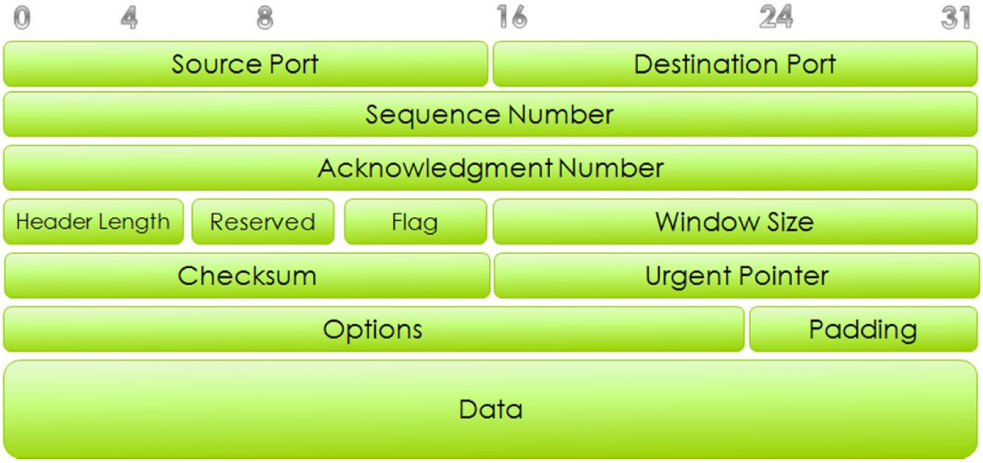
2.3.3.1.1 *Transmission Control Protocol* (TCP)

Protokol TCP pada umumnya digunakan pada layanan internet seperti telnet, FTP, SMTP, NNTP, dan HTTP. Protokol TCP adalah protokol yang menghubungkan antara dua *endpoint*. TCP adalah protokol yang dapat memastikan bahwa tujuan dalam keadaan aktif dan data yang diterima tidak rusak.

Protokol TCP akan melakukan pemutusan untuk suatu koneksi yang melanggar dari aturan. Sebagai contoh, jika dalam perjalanan paket TCP hilang atau rusak pada waktu pengiriman maka paket tersebut tidak akan diterima. Jika pengiriman diulangi kembali dan tetap ada data yang hilang atau rusak maka koneksi akan diputus secara otomatis. TCP terbagi menjadi dua bagian yaitu TCP *header* dan TCP *option*.

1. *Header* TCP

   Header TCP mampu menampilkan seluruh mekanisme protokol yang cukup besar dengan panjang maksimum 20 byte. Elemen-elemen dari TCP *header* dapat dilihat pada Gambar 2.2.



Gambar 2.2 *Header* TCP

Elemen-elemen pada *header* TCP akan dijelaskan pada Tabel 2.1.

Tabel 2.1 Elemen TCP Header

Elemen	Penjelasan
Source Port (16 bit)	Port number dari sumber
Destination Port (16 bit)	Port number dari tujuan
Sequence Number (32 bit)	Nomor urut data pertama di dalam segmen kecuali bila flag SYN diset
Acknowledgment Number (32 bit)	Nomor urut data berikutnya yang merupakan tempat entitas TCP
Data offset (4bit)	Berjumlah 32 bit kata di dalam header
Reserved	Bertujuan untuk pengguna-pengguna berikutnya
Flag (6 bit)	Tanda untuk sebuah paket URG,ACK, PSH, RST, SYN, FIN

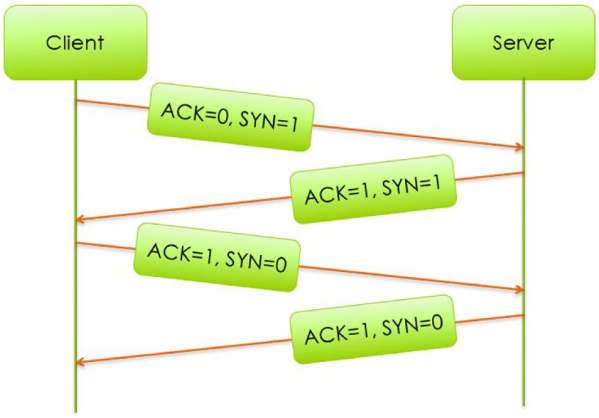
Window Size	Pengalokasian kredit kontrol alur dalam octet. Memuat sejumlah octet data yang dimulai dengan satu balasan bahwa pengirim ingin menerima data
Checksum (16 bit)	Elemen dari one complement dari sum modulo $2^{16} - 1$ dari seluruh word 16 bit di dalam segmen ditambah dengan pseudo header
Urgen Pointer	Menunjuk pada byte terakhir dalam urutan data urgen. Hal ini memungkinkan penerima mengetahui berapa banyak data urgen yang datang
Data	Berisi data yang dikirimkan

2. TCP *Option*

TCP *option* berisi beberapa *flag* atau tanda. Beberapa *flag* tersbut diantaranya yaitu :

1. URG (*Urgent*)
2. ACK (*Acknowledgment*)
3. PSH (*Push*)
4. RST (*Reset*)
5. SYN (*Synchronize*)
6. FIN (*Finish*)

URG dan PSH digunakan untuk melakukan identifikasi pada data. PSH menerima *buffer* dan beberapa program dari data, sedangkan URG secara umum menandai data dari pengirim yang disebut dengan *out of band data*. ACK dan SYN digunakan untuk membuat *handshake* pada protokol TCP dengan tiga cara. Pada Gambar 2.3 diperlihatkan bagaimana prokol TCP melakukan proses *handshaking*.

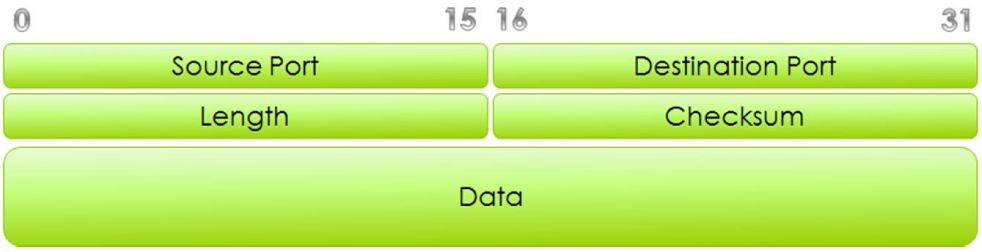


Gambar 2.3 *Handshaking* Pada Protokol TCP

RST dan FIN merupakan cara untuk menutup suatu koneksi. RST berfungsi sebagai tanda menutup koneksi secara tidak normal karena terjadi kesalahan atau *error*, sedangkan FIN berfungsi sebagai tanda untuk menutup koneksi secara normal.

2.3.3.1.2 User Datagram Protocol (UDP)

UDP menyediakan layanan koneksi untuk prosedur-prosedur pada *level* aplikasi. Pada dasarnya UDP merupakan suatu layanan protokol yang kurang bisa diandalkan karena kurang memberikan perlindungan dalm pengiriman dan duplikasi data. Datagram merupakan suatu paket *switching*, sebuah paket terpisah-pisah dari paket lain yang membawa informasi yang memadai untuk ro*u*ting dari *Data Terminal Equipment* (DTE) sumber ke DTE tujuan tanpa harus menetapkan koneksi antara DTE dan jaringan[5].



Gambar 2.4 User Datagram Protocol

Pada dasarnya struktur UDP hampir sama dengan TCP, UDP *header* berisi *source* dan *destination port number*, seperti halnya *source* TCP dan *destination port number*, hanya saja pada UDP *header* tidak terdapat *flag*, *sequence number*, atau bit ACK. Berikut ini adalah gambar dari struktur UDP yang diperlihatkan pada Gambar 2.4.

2.3.3.1.3 *Internet Control Message Protocol* (ICMP)

Internet Control Message Protocol (ICMP) adalah protokol pada TCP/IP yang bertugas mengirimkan pesan-pesan kesalahan dan kondisi lain yang memerlukan perhatian khusus. Hal ini dapat dilakukan dengan mengevaluasi pesan yang dihasilkan oleh ICMP. Jenis pesan pada ICMP ada dua yaitu ICMP *error message* dan ICMP *query message*. Dengan memanfaatkan kedua pesan ini seorang *hacker* dimungkinkan untuk mengumpulkan informasi sebanyak-banyaknya sebelum melakukan penyerangan seperti ICMP *sweep*, *traceroute* dan OS *fingerprinting* dan menyiapkan strategi yang diperlukan untuk melakukan penyerangan seperti ICMP *route redirect*, ICMP *informational message*, ICMP *router discovery message* dan ICMP *flood*[5].

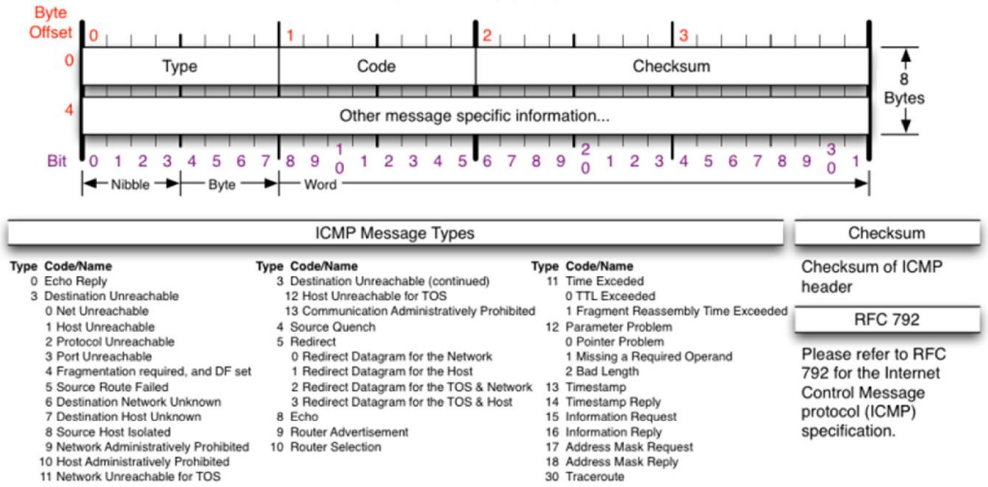
Contoh-contoh dari paket ICMP dijelaskan pada Tabel 2.2.

Tabel 2.2 Paket ICMP

Paket ICMP	Penjelasan
Echo request	Apa yang dikirim host ketika user menjalankan perintah ping
Echo response	Apa yang direspon host terhadap echo request
Time exceeded	Apa yang dikembalikan router ketika menentukan bahwa suatu paket terjadi looping
Destination unreachable	Apa yang router kembalikan ketika tujuan dari suatu paket tidak bisa dicapai oleh beberapa alasan, misalkan suatu sistem sedang down

Paket ICMP	Penjelasan
Redirect	Respon apa yang router kirim ke host suatu paket telah dikirim ke router yang lain

ICMP berada pada *level* yang sama pada *internet protocol* yang berada pada arsitektur TCP/IP. Pesan yang telah disusun dan kemudian ditransfer dalam lintasan *internet protocol* dan membungkus pesan tersebut dengan sebuah *header* IP dan kemudian mentransmisikan datagram yang dihasilkan dengan cara ditransfer biasa, dikarenakan pesan dari ICMP ditransmisikan di dalam datagram IP sehingga pengirim tidak dijamin keamanannya. Oleh karena itu penggunaan ICMP tidak bisa diandalkan. Pada Gambar 2.5 diperlihatkan format ICMP message.



Sumber:http://www.siongboon.com/projects/2006-03-06_serial_communication/

Gambar 2.5 ICMP *Message*

2.4 Perangkat Jaringan Komputer

Dalam jaringan komputer terdapat beberapa perangkat keras yang dibutuhkan agar dapat menghubungkan komputer satu dengna komputer lainnya. Perangkat-perangkat keras itu diantaranya : kabel UTP, *switch*, *switch manageable*, *network interface card* (NIC).

2.4.1 Kabel *Unshielded Twisted Pair* (UTP)

Kabel UTP digunakan sebagai penghubung antar perangkat keras satu ke perangkat keras yang lain agar perangkat tersebut dapat berhubungan. Kabel ini memiliki delapan jalur yang dapat digunakan. Kabel UTP memiliki beberapa jenis kategori, seperti yang dijelaskan pada Tabel 2.3.

Tabel 2.3 Kategori Kabel UTP

Kategori	Keterangan
Cat 1	Digunakan diperangkat telephone pada jalur ISDN (*Integrated Service Digital Network*), juga untuk menghubungkan modem dengan line telephone
Cat 2	Transmisi suara hingga 4 Mbps, digunakan pada topologi Token Ring
Cat 3	Transmisi data hingga 10 Mbps, digunakan pada topologi Token Ring atau 10BaseT
Cat 4	Transmisi data hingga 16 Mbps, digunakan pada topologi Token Ring
Cat 5e	Transmisi data hingga 100 Mbps, digunakan pada topologi Star dan Tree. Teknologi yang disupport ethernet 16 Mbps,Fast Ethernet 100 Mbps, Token Ring 16 Mbps
Cat 6	Transmisi data hingga 2,5 Gbps, Jarak hingga 200 meter
Cat 7	Transmisi data hingga 10 Gbps

Ada dua jenis pemasangan kabel UTP yang umum digunakan yaitu *straight* dan *cross-over*. *Straight* biasanya digunakan untuk menghubungkan perangkat-perangkat keras yang berbeda, contoh komputer dan *switch*. Kabel *straight* ini dipasang sejajar antar pin pada konektor RJ45. *Cross-over* biasanaya digunakan untuk menghubungkan perangkat-perangkat keras yang sama, contohnya komputer dengan komputer. Kabel *Cross* ini dipasang dengan menghubungkan pin 1 ke pin 3, pin 2 ke pin 6, pin 3 ke pin 1, pin 6 ke pin 2 dan pin-pin yang lain tetap dihubungkan sesuai dengan nomor pin pada RJ45.

2.4.2 *Switch*

Switch adalah perangkat jaringan komputer yang digunakan untuk menghubungkan beberapa *network segment*. Istilah *switch* juga merujuk pada *multi-port network bridge* (Jembatan jaringan *multi-port*) yang memproses dan mengirimkan data pada layer kedua pada OSI. *Switch* yang juga memproses data pada layer ketiga OSI disebut juga *multilayer switch*. Fungsi dari *switch* adalah membuat jalur distribusi data tersendiri tanpa mengganggu jalannya distribusi data lain yang sedang berjalan. Sehingga data bisa berjalan *full duplex* (kirim terima secara bersamaan) dan mempunyai *bandwidth* tersendiri. Hal ini berbeda dengan *switch* yang menjalankan data dalam *half duplex* (kirim-terima secara bergantian) dan membagi semua *bandwidth* ke semua jalur, yang bisa menyebabkan terjadinya tabrakan data. Misal ada dua komputer A dan B, dan dua lagi C dan D. Keduanya dihubungkan dengan *switch* yang sama. Maka komputer A dan B dapat saling berkomunikasi tanpa mengganggu C dan D, begitu pula sebaliknya. Dari jenis konfigurasi datanya *switch* dapat dibedakan menjadi :

1. *Unmanaged switch*

   Switch jenis ini tidak dapat dikonfigurasi dan bersifat *plug and play*. *Switch* jenis ini bisa ditemukan di rumah atau di perusahaan atau jaringan kecil.



Gambar 2.6 *Unmanaged Switch*

2. *Managed switch*

   Mempunyai fitur untuk memodifikasi operasi dari *switch*. Biasanya cara untuk mengaturnya lewat CLI (*Command Line Interface*) yang diakses lewat *console*, telnet atau *secure shell*. Biasanya juga ditanamkan SNMP (*Simple*

Network Management Protocol) yang diatur melalui *web browser*. Jenis ini terbagi menjadi dua jenis :

1. *Smart Switch* (*Intelligent Switch*)
   Adalah *managed switch* dengan fitur manajemen yang terbatas. Pengaturannya melalui *web browser*, pengaturan tersebut antara lain terdiri dari VLAN, *port-bandwith* dan *duplex*.
2. *Enterprise Managed Switch* (*Full Managed Switch*)
   Adalah managed *switch* dengan fitur manajemen yang lengkap. Pengaturan dilakukan melalui *web browser* atau CLI, dengan pengaturan tambahan sepertu *backup* konfigurasi *switch*.



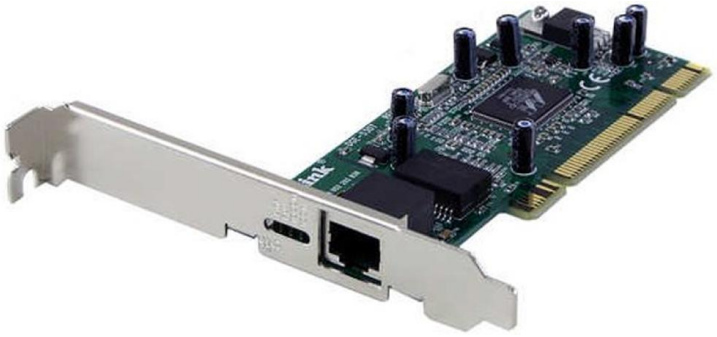
Gambar 2.7 *Managed Switch*

Dibanding *Smart Switch*, *switch* ini memiliki pengaturan lebih lengkap. Contoh perangkat keras ini yaitu Cisco Catalyst 2960. Pada jenis *switch* ini mendukung untuk aplikasi SPAN (*Switched Port Analyzer*), SPAN ini digunakan untuk mengirimkan salinan paket data pada *switch port* (atau seluruh VLAN) ke *switch port* lain yang terhubung pada suatu *network monitoring* seperti IDS. *Administrator* menggunakan fitur ini untuk menganalisa data pada jaringan. Proses ini membantu *administrator* unutk *memonitoring* performa *network* dan akan membantu memperingatkan ketika terjadi suatu masalah.

2.4.3 *Network Interface Card* (NIC)

Network Interface Card adalah perangkat keras pada jaringan komputer yang digunakan sebagai *interface* komputer untuk berhubungan dengan jaringan

komputer. Pada NIC terdapat *port ethernet* yang digunakan untuk memasukan konektor RJ45.



Gambar 2.8 *Network Interface Card*

NIC menentukan kecepatan data komunikasi yang dapat diproses oleh komputer.

2.5 Keamanan Jaringan Komputer

Keamanan jaringan secara umum adalah komputer yang terhubung ke jaringan, mempunyai ancaman keamanan yang lebih besar dibandingkan komputer yang tidak terhubung ke jaringan. Namun keamanan komputer biasanya bertentangan dengan *network access*, dimana bila *network access* semakin mudah, maka *network security* semakin rawan, dan bila *network* semakin baik maka *network access* semakin tidak nyaman. Suatu *network* dirancang sebagai komunikasi data dengan tujuan meningkatkan akses ke sistem komputer, sementara *security* dirancang untuk mengendalikan akses. Sebagai analogi, keamanan di rumah dilakukan dengan memberi kunci di pintu rumah, tidak dengan cara memblokir jalan di depan rumah. Hal seperti ini juga diterapkan pada *network security*. Keamanan dijaga untuk setiap *host-host* tertentu, tidak langsung pada *network*-nya.

Langkah awal dalam mengembangkan rencana *network security* yang efektif adalah dengan mengenali ancaman yang mungkin datang. Dalam RFC 1244, *Site security Handbook*, dibedakan tiga tipe ancaman, yaitu :

1. Akses tidak sah oleh orang yang tidak memiliki wewenang.

2. Kesalahan informasi, segala masalah yang dapat menyebabkan diberikannya informasi yang penting atau sensitif kepada orang yang salah, yang seharusnya tidak boleh mendapatkan informasi tersebut.
3. Penolakan terhadap *service*, segala masalah mengenai *security* yang menyebabkan sistem mengganggu pekerjaan-pekerjaan yang produktif.

2.5.1 Tujuan Keamanan Jaringan Komputer

Pada dasarnya tujuan dari keamanan komputer yaitu :

1. *Confidentiality*, merupakan usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. *Confidentiality* biasanya berhubungan dengan informasi yang diberikan ke pihak lain.
2. *Integrity*, Keaslian pesan yang dikirim melalui sebuah jaringan dan dapat dipastikan bahwa informasi yang dikirim tidak dimodifikasi oleh orang yang tidak berhak dalam perjalanan informasi tersebut.
3. *Availability*, aspek *availability* atau ketersediaan berhubungan dengan ketersediaan informasi ketika dibutuhkan.

Tabel 2.4 CIA

	Definition	**Tools**
Confidentiality	a. Concealment of info and resource b. Hide existence of info and resource	a. Encryption b. Access control
Integrity	a. Trustworthiness off info and resource (Authentication) b. Correctness of data (data integrity)	a. Preventions b. Detection
Availability	Ability to use info and resources	a. System design b. Statstical models of use

2.5.2 Tipe Ancaman Keamanan Sistem Komputer

Fungsi sistem komputer dapat digunakan sebagai dasar untuk menentukan model tipe ancaman dari suatu sistem komputer. Berdasarkan fungsi sistem

komputer sebagai penyedia informasi, ancaman terhadap sistem komputer dikategorikan menjadi empat, yaitu:

1. *Interruption*, merupakan suatu ancaman terhadap *availability*, informasi atau data yang ada dalam sistem komputer dirusak, dihapus, sehingga jika dibutuhkan maka sudah tidak ada lagi[8].
2. *Interception*, merupakan ancaman terhadap kerahasiaan (*secrecy*). Informasi yang ada di dalam sistem disadap oleh orang yang tidak berhak[8].
3. *Modification*,merupakan ancaman terhadap integritas. Orang yang tidak berhak berhasil menyadap lalu-lintas informasi yang sedang dikirim lalu mengubahnya sesuai keinginan orang itu[8].
4. *Fabrication*, merupakan ancaman terhadap integritas. Orang yang tidak berhak berhasil meniru atau memalsukan suatu informasi sehingga orang yang menerima informasi tersebut menyangka informasi tersebut berasal dari orang yang dikehendaki oleh si penerima informasi tersebut[8].

2.5.3 Bentuk Ancaman Jaringan Komputer

Berikut beberapa ancaman yang umum ditemui pada jaringan nirkabel:

1. MAC *Spoofing*

   Penyerang berusaha mendapatkan koneksi ke dalam jaringan dengan mengambil alamat NIC dari suatu perangkat komputer pada jaringan tersebut[6].
2. ARP *Spoofing*

   Penyerang menangkap penyebaran paket ARP dari *access point* dan kemudian mengirimkan balasan ARP fiktif sehingga informasi perangkat dari penyerang akan terpetakan ke dalam tabel ARP untuk kemudian mendapatkan hak akses kedalam jaringan[6].
3. *Man in the Middle Attack*

   Metode serangan ini biasanya didahului dengan ARP spoofing kemudian penyerang menempatkan perangkat yang dimilikinya sebagai sebuah komputer fiktif yang akan terlihat resmi dari sisi access point[6].

4. *Denial of Service*

   Metode serangan dengan mengirimkan paket data dalam jumlah yang sangat besar terhadap jaringan yang menjadi targetnya secara terus-menerus. Hal ini dapat mengganggu lalu-lintas data bahkan kerusakan sistem jaringan[6].

2.6 Jenis Serangan Jaringan Komputer

Jenis dan teknik serangan yang mengganggu jaringan komputer beraneka jenis, diantaranya adalah :

2.6.1 *Port Scanning*

Merupakan suatu proses untuk mencari port yang terbuka pada suatu jaringan komputer. Dari hasil *scanning* akan didapat letak kelemahan sistem tersebut. Pada dasarnya sistem *port scanning* mudah untuk dideteksi, tetapi penyerang akan menggunakan beberapa cara metode untuk menyembunyikan serangan[7].

2.6.2 *Teardrop*

Teknik ini menggunakan tekni penyerangan dengan mengeksploitasi proses *disassembly- reassembly* paket data. Dalam jaringan internet sering kali data harus dipotong menjadi paket yang lebih kecil untuk menjamin reliabilitas dan proses multiple akses jaringan. Pada proses pemotongan data paket yang normal, setiap potongan diberi informasi offset data yang berbunyi “Potongan byte ini merupakan potongan 600 byte dari total 800 byte paket yang dikirimkan”. Program memanipulasi potongan data sehingga terjadi *overlapping* antara paket yang diterima di bagian penerima setelah potongan-potongan paket disusun kembali[7].

2.6.3 IP *spoofing*

Teknik ini bekerja dengan mengganti alamat IP pengguna yang lain yang bukan penyerang sebenarnya. Hal ini terjadi karena salah rancang (*design flaw*) bagian urutan nomor (*sequence number*) dari paket TCP/IP. Dalam beberapa

kasus, penyerang menggunakan satu alamat IP sumber yang spesifik pada semua paket IP yang keluar untuk membuat semua pengembalian paket IP dan pesan ICMP ke pemilik alamat tersebut[7].

2.6.4 ICMP *flood*

Penyerang melakukan eksploitasi dengan tujuan untuk membuat target *host* menjadi terganggu, yang disebabkan oleh pengiriman sejumlah paket yang besar ke arah target *host*. Eksploitasi sistem ini dilakukan dengan mengirimkan suatu perintah "ping" dengan tujuan *broadcast* atau *multicast* dimana pengirim dibuat seolah-olah adalah target host. Semua balasan dikembalikan ke target *host*. Hal inilah yang menyebabkan *host* target menjadi terganggu dan menurunkan kinerja jaringan bahkan dapat menyebabkan *denial of service*[7].

2.6.5 UDP *flood*

UDP *flood* mengaitkan dua sistem tanpa disadari. Dengan cara *spoofing*, *User Datagram Protocol* (UDP) *flood attack* akan menempel pada servis UDP di salah satu mesin yang untuk keperluan "percobaan" akan mengirimkan sekelompok karakter ke mesin lain, yang diprogram untuk meng- *echo* setiap kiriman karakter yang diterima melalui servis. Karena paket UDP tersebut di-*spoofing* diantara ke dua mesin tersebut maka yang terjadi adalah "banjir" tanpa henti paket kiriman karakter yang tidak berguna diantara diantara kedua mesin tersebut. Untuk menanggulangi UDP *flood*, *disable* semua servis UDP di semua mesin di jaringan, atau dengan menyaring semua servis UDP yang masuk pada *firewall*[7].

2.6.6 *Packet interception*

Gangguan jenis ini dilakukan dengan membaca paket di saat paket tersebut sedang mengalami *packet sniffing*. *Packet interception* merupakan cara penyerang untuk mendapatkan informasi yang ada di dalam paket tersebut. Hal ini dapat dicegah dengan mengenkripsi terlebih dahulu, sehingga penyerang akan mengalami kesulitan untuk membuka paket tersebut[7].

2.6.7 Smurf *attack*

Gangguan jenis ini biasanya dilakukan dengan menggunakan IP *spoofing*, yaitu mengubah nomor IP dari datangnya *request*. Dengan menggunakan IP *spoofing*, respons dari ping tadi dialamatkan ke komputer yang IP-nya di-*spoof*. Akibatnya, komputer tersebut akan menerima banyak paket. Hal ini dapat mengakibatkan pemborosan penggunaan *bandwidth* jaringan yang menghubungkan komputer tersebut[7].

2.6.8 DOS (*Denial Of Service*)

DOS merupakan salah satu serangan yang banyak ditemui dalam dunia *networking* saat ini. Kita tidak pernah akan tahu kapan kita akan mendapat serangan ini. Serangan DOS dapat terjadi kapan saja pada jaringan, yang dapat kita lakukan adalah menyusun dan menerapkan metode penangan untuknya. Serangan ini dapat ditujukan kepada siapa saja, bahkan ke personal. Namun yang paling sering terkena dampaknya adalah *server- server* besar seperti yahoo, google, serta perbankan yang secara langsung memberikan pelayanannya melalui jaringan. Ada beberapa contoh misal pada anggota parpol, mungkin bila anda seorang yang berkecimpung di dunia politik atau pengurus parpol, anda akan berpikir serangan ini ditujukan untuk menjatuhkan citra politik anda dan parpol. Bila anda seorang pengusaha serangan ini bisa saja ditujukan oleh saingan bisnis anda. Bila kita memutuskan untuk memberikan pelayanan melalui jalur *networking* maka kita juga harus siap menanggung resikonya, salah satunya adalah serangan DOS (*Denial Of Service*).

Faktor penyebab yang paling banyak ditemukan di dunia nyata sebenarnya karena keisengan belaka atau dengan kata lain ”its just for fun”. Pembuktian jati diri di kalangan *underground* juga bukanlah alasan yang salah, sebagian komunitas mungkin berpendapat bila seseorang telah dapat melumpuhkan sebuah sistem dengan cara yang bermacam – macam maka ia baru bisa diterima dalam komunitas tersebut. ”welcome to Underground World”. Seorang *Elite* yang beretika tentunya tidak diharapkan untuk melakukan serangan *Denial Of Service* ini. Serangan ini biasanya bertujuan untuk mematikan pelayanan dari komputer

atau jaringan yang diserang. Dampaknya akan sangat besar bagi perusahaan atau instansi yang menyediakan jasa terutama bagi perbankan. Korban yang terkena serangan ini tidak dapat memberikan pelayanan yang seharusnya. Serangan Dos ini dapat menghambat bahkan mematikan pelayanan pada sebuah sistem, sehinggan pengguna yang absah tidak dapat menerima atau mendapatkan pelayanan yang seharusnya. Bayangkan saja bila sebuah perusahaan bank tidak dapat memberikan pelayanan kepada nasabahnya, maka akan sangat fatal akibatnya bagi kelangsungan perusahaan tersebut. Atau sebuah *provider* internet yang tidak dapat memberikan *bandwith*-nya kepada klien, maka akan berdampak tidak hanya kepada provider tersebut yang tentunya harus membayar jaminan koneksi sesuai yang mereka terapkan namun juga kepada penyewa koneksi. Bila salah satu penyewanya adalah sebuah warnet atau bahkan banking, tentunya akan berdampak luas.

Serangan *Denial Of Service* ini pada dasarnya sulit untuk dideteksi, kecuali bila penyerang telah melakukan beberapa kali percobaan dengan alamat ip yang sama. Tentu akan sangat mudah untuk membloknya. DOS cukup sulit untuk diatasi karena serangan ini pada dasarnya juga berkaitan dengan pelayanan yang diberikan, sebuah sistem dengan tingkat keamanan yang tinggi biasanya memberikan kenyamanan yang rendah bagi penggunanya. Bayangkan bila *server* yahoo dijadikan perantara untuk menyerang, tentu akan sangat membingungkan bagi *administrator* disebuah penyedia jasa internet. Sang *administrator* tidak bisa semerta- merta memblok alamat ip dari yahoo karena yang akan terkena dampaknya adalah pengguna jasa internet tersebut. Maka perlu dicari solusi yang tepat oleh *provider* internet tersebut.

Saat seseorang ingin melakukan serangan, ada banyak sekali cara yang dapat dilakukan. Klasifikasi dari serangan DOS ini, diantaranya *syn attack* dan UDP *flood*. Serangan DOS biasa saat ini sudah tidak terlalu besar lagi dampaknya, karena para ahli jaringan telah banyak menemukan metode pengamanan terbaru dan menambal *patch- patch* pada sebuah sistem yang dapat memungkinkan seseorang dengan mudahnya melakukan DOS.

Serangan DOS bukan merupakan hal yang sulit untuk dilakukan, namun bila penyerang ingin memberikan dampak yang besar, maka dibutuhkan pengetahuan dan sumber daya yang memadai untuk melakukan hal tersebut. Pengamatan tentang target juga merupakan hal – hal yang tidak boleh dilupakan. Mengatasi DOS secara kasar dapat dilakukan dengan cara mematikan layanan yang diserang atau mungkin mematikan server kita, namun akan berakibat kepada user absah yang ingin mengakses *server* kita (”*destination unreachable*”).

2.6.9 SQL *Injection*

Pada dasarnya SQL *Injection* merupakan cara mengeksploitasi celah keamanan yang muncul pada *level* atau “layer” *database* dan aplikasinya. Celah keamanan tersebut ditunjukkan pada saat penyerang memasukkan nilai “string” dan karakter-karakter yang ada dalam instruksi SQL, perintah tersebut hanya diketahui oleh sejumlah kecil individu yang berusaha untuk mengeksploitasinya. Karena tipe data yang dimasukkan tidak sama dengan yang seharusnya, maka terjadi sebuah aktivitas “liar” yang tidak terduga yang dapat mengakibatkan individu yang tidak berhak dapat masuk ke dalam sistem yang telah terproteksi. Dikatakan sebagai sebuah “injeksi” karena aktivitas penyerangan dilakukan dengan cara memasukkan string khusus untuk melewati filter logika hak akses pada *website* atau sistem komputer yang dimaksud[8].

2.6.10 *Cross Site Scripting*

Cross Site Scripting (XSS) adalah suatu serangan dengan mekanisme *injection* pada aplikasi *web* dengan memanfaatkan metode HTTP GET atau HTTP POST. *Cross Site Scripting* biasa digunakan oleh pihak-pihak yang berniat tidak baik dalam upaya mengacaukan konten website dengan memasukkan naskah program sebagai bagian dari teks masukan melalui formulir yang tersedia. XSS memanfaatkan lubang kelemahan keamanan yang terjadi pada penggunaan teknologi *dynamic page*. Serangan jenis ini dapat diakibatkan oleh kelemahan yang terjadi akibat ketidakmampuan *server* dalam memvalidasi input yang diberikan oleh pengguna. Hal ini memungkinkan halaman yang dihasilkan menyertakan perintah yang sebenarnya tidak diperbolehkan[8].

2.7 *Intrusion Detection System*

Intrusion Detection System (IDS) dapat didefinisikan sebagai *tool*, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi dan memberikan laporan terhadap aktifitas jaringan komputer. Ada beberapa alasan menggunakan IDS, diantaranya adalah :

1. Mencegah ancaman keamanan yang terus meningkat, karena banyak ditemukan kegiatan illegal yang diperbuat oleh orang-orang yang tidak bertanggung jawab.
2. Mendeteksi serangan dan pelanggaran keamanan sistem jaringan yang tidak bias dicegah oleh sistem yang umum dipakai, seperti *firewall*.
3. Mendeteksi serangan awal (biasanya *network probe* dan aktivitas *doorknob rattling*), Langkah awal dari suatu serangan pada umumnya adalah melakukan penyelidikan atau menguji sistem jaringan yang akan menjadi target, untuk mendapatkan titik-titik dimana mereka akan bisa masuk.
4. Mengamankan *file* yang keluar dari jaringan.
5. Sebagai pengendali *security design* dan administrator terutama bagi perusahaan yang besar.
6. Menyediakan informasi terhadap gangguan secara langsung, meningkatkan diagnosis, *recovery* dan mengkoreksi faktor-faktor penyebab serangan.

2.7.1 Jenis *Intrusion Detection System*

Pada dasarnya IDS terbagi menjadi dua jenis, yaitu :

1. *Host Based*

   IDS *host based* bekerja pada *host* yang akan dilindungi. IDS jenis ini dapat melakukan berbagai macam tugas untuk mendeteksi serangan yang dilakukan pada *host* tersebut. Keunggulan IDS *Host based* adalah pada tugas-tugas yang berhubungan dengan keamanan *file*. Misalnya ada atau tidaknya *file* yang telah diubah atau adakah usaha untuk mendapatkan akses ke *file-file* yang *sensitive*.

2. *Network Based*

   IDS *Network based* biasanya berupa mesin yang khusus dipergunakan untuk melakukan *monitoring* seluruh segmen dari jaringan. IDS ini akan mengumpulkan paket-paket data yang terdapat pada jaringan dan kemudian menganalisanya serta menentukan apakah paket itu berupa suatu paket yang normal atau suatu serangan.

2.7.2 Metode Pada *Intrusion Detection System*

Ada dua metode yang digunakan pada *intrusion detetion system*, diantaranya:

1. *Signature Based*

   IDS yang berbasis pada *signature* akan melakukan pengawasan terhadap paket-paket dalam jaringan dan melakukan pembandingan terhadap paket-paket tersebut dengan basis data *signature* yang dimiliki oleh sistem IDS ini atau atribut yang dimiliki oleh percobaan serangan yang pernah diketahui. Cara ini hampir sama dengan cara kerja aplikasi antivirus dalam melakukan deteksi terhadap *malware*. Intinya adalah akan terjadi keterlambatan antara terdeteksinya sebuah serangan di internet dengan *signature* yang digunakan untuk melakukan deteksi yang diimplementasikan didalam basis data IDS yang digunakan. Jadi bisa saja basis data *signature* yang digunakan dalam sistem IDS ini tidak mampu mendeteksi adanya sebuah percobaan serangan terhadap jaringan karena informasi jenis serangan ini tidak terdapat dalam basis data *signature* sistem IDS ini. Selama waktu keterlambatan tersebut sistem IDS tidak dapat mendeteksi adanya jenis serangan baru[9].

2. *Anomaly Based*

   IDS jenis ini akan mengawasi *traffic* dalam jaringan dan melakukan perbandingan *traffic* yang terjadi dengan rata-rata *traffic* yang ada (stabil). Sistem akan melakukan identifikasi apa yang dimaksud dengan jaringan "normal" dalam jaringan tersebut, berapa banyak *bandwidth* yang biasanya digunakan di jaringan tersebut, protokol apa yang digunakan, *port-port*

dan alat-alat apa saja yang biasanya saling berhubungan satu sama lain didalam jaringan tersebut, dan memberi peringatan kepada administrator ketika dideteksi ada yang tidak normal, atau secara signifikan berbeda dari kebiasaan yang ada[9].

2.8 Snort

Snort merupakan perangkat lunak untuk mendeteksi penyusup dan mampu menganalisis paket yang melintasi jaringan secara *realtime*, serta mampu membuat *log* ke dalam *database*. Snort merupakan *software open source* yang dapat digunakan pada berbagai macam sistem operasi. Snort adalah suatu aplikasi *intrusion detection system* yang banyak dipakai orang di dunia. Snort dapat dioperasikan dengan tiga *mode*, yaitu :

1. Paket *Sniffer*, digunakan untuk melihat paket yang lewat di jaringan.
2. Paket *logger*, digunakan untuk mencatat semua paket yang lewat di jaringan untuk dianalisis dikemudian hari.
3. NIDS, deteksi penyusup pada *network*. Pada *mode* ini snort akan berfungsi untuk mendeteksi serangan yang dilakukan melalui jaringan komputer. Untuk menggunakan mode IDS ini diperlukan *setup* dari berbagai *rules* atau aturan yang akan membedakan sebuah paket normal dengan paket yang membawa serangan.

2.8.1 Komponen Snort

Snort terdiri dari komponen – komponen yang mempunyai tugas dan fungsinya sendiri – sendiri yaitu:

1. *Packet capture library*

   Packet capture library adalah sebuah perangkat lunak yang terpisah yang mengambil paket data dari NIC. Paket – paket data itu adalah paket data Lapisan *Data Link* (OSI *model*) yang biasanya disebut *frame* yang masih belum diproses. Pada sistem Linux dan UNIX, Snort

menggunakan libpcap, sedangkan pada system Windows, Snort menggunakan winpcap [6].

2. *Packet decoder*

   Packet decoder mengambil *frame* lapisan 2 (Data Link) yang dikirimkan oleh packet capture library dan kemudian memecahnya. Pertama – tama komponen ini membaca kode sandi terhadap frame lapisan 2, kemudian paket lapisan 3 (protokol IP), lalu kemudian paket lapisan 4 (paket TCP atau UDP). Setelah proses selesai dilakukan, snort mempunyai semua informasi masing – masing protokol untuk pemrosesan lebih lanjut [6].

3. *Preprocessor*

   Preprocessor pada Snort memiliki beberapa fitur tambahan yang dapat dimatikan atau dinyalakan. *Preprocessor* bekerja pada paket yang sudah dibaca kode sandinya dan kemudian melakukan transformasi pada data itu supaya lebih mudah untuk diproses oleh snort [6].

4. *Detection engine*

   Komponen ini mengambil informasi dari *packet decoder* dan *preprocessor* yang kemudian memproses data itu pada lapisan *Transport* dan *Application*, membandingkan data yang terkandung dalam paket dengan aturan-aturan yang juga merupakan fitur tambahan dari komponen ini [6].

5. *Output*

   Ketika *preprocessor* terpancing karena adanya data yang cocok dengan definisi jenis serangan, Snort kemudian menghasilkan peringatan dan kemudian melakukan pencatatan. Snort mendukung beberapa macam keluaran, seperti keluaran dalam format teks atau biner. Pencatatan juga bisa dilakukan ke dalam penyimpanan data ataupun *syslog* [6].

2.9 Basis Data

Dalam pengertiannya basis data sendiri terdiri dari 2 kata yaitu basis yang diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya.[21] Basis data sendiri dapat didefinisikan dalam beberapa sudut pandang seperti:

a. kelompok data himpunan yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
b. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
c. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronis.

2.10 PHP

PHP (akronim dari PHP:Hypertext Preprocessor) adalah bahasa pemrograman yang berfungsi untuk membuat website dinamis maupun aplikasi web. Berbeda dengan HTML yang hanya bisa menampilkan konten statis, PHP bisa berinteraksi dengan database, file dan folder, sehingga membuat PHP bisa menampilkan konten yang dinamis dari sebuah website. [22]

Blog, Toko Online, CMS, Forum, dan Website Social Networking adalah contoh aplikasi web yang bisa dibuat oleh PHP. PHP adalah bahasa scripting, bukan bahasa tag-based seperti HTML. PHP termasuk bahasa yang cross-platform, ini artinya PHP bisa berjalan pada sistem operasi yang berbeda-beda (Windows, Linux, ataupun Mac). Program PHP ditulis dalam file plain text (teks biasa) dan mempunyai akhiran ".php".

PHP diciptakan pertama kali oleh Rasmus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada homepage-nya. Rasmus Lerdorf adalah seorang pendukung open source. Oleh karena itu, ia mengeluarkan Personal Home Page versi 1.0 secara gratis, kemudian menambah kemampuan PHP 1.0 dan meluncurkan PHP 2.0.[22]

Pada Tahun 1996, PHP telah banyak digunakan dalam website didunia. Sebuah kelompok pengembang software yang terdiri dari Zeew Suraski, Andi Gutman, Stig Bakken, Shane Caraveo dan Jim Winstead bekerja sama untuk menyempurnakan PHP 2.0. Akhirnya, pada tahun 1998 PHP 3.0 diluncurkan. Penyempurnaan terus dilakukan sehingga pada tahun 2000 dikeluarkan PHP 4.0.

2.11 CSS

Cascading Style Sheet adalah suatu teknologi bahasa pemograman yang digunakan untuk memperindah tampilan halaman website. Singkatnya dengan menggunakan metode CSS ini dengan mudah mengubah secara keseluruhan warna dan tampilan yang ada di situs, sekaligus memformat ulang situs anda atau merubahnya. Jelas, dengan sedikit berfikir dan menggunakan logika maka akan didapatkan hasil yang memuaskan. Cascading Style Sheet (CSS) technology support hampir semua web Browser. Karena CSS telah di standarkan oleh World Wide Web Consortium(W3C) untuk di gunakan di web browser.[24]

2.12 MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL B membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. Tidak sama dengan proyek-proyek seperti Apache, dimana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki

oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.