BAB 2
LANDASAN TEORI

2.1 Konsep Dasar Jaringan Komputer

Jaringan komputer diartikan sebagai suatu himpunan interkoneksi sejumlah komputer yang dapat saling bertukar informasi. Bentuk koneksinya tidak harus malalui kawat saja melainkan dapat menggunakan serat optik, atau bahkan satelit komunikasi.

Jaringan Komputer adalah merupakan sekelompok komputer otonom yang saling berhubungan antara satu dengan lainnya menggunakan protokol komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, program-program, dan penggunaan bersama perangkat keras seperti printer, harddisk, dan lain sebagainya. (Kristanto : Hal 2 : 2003)

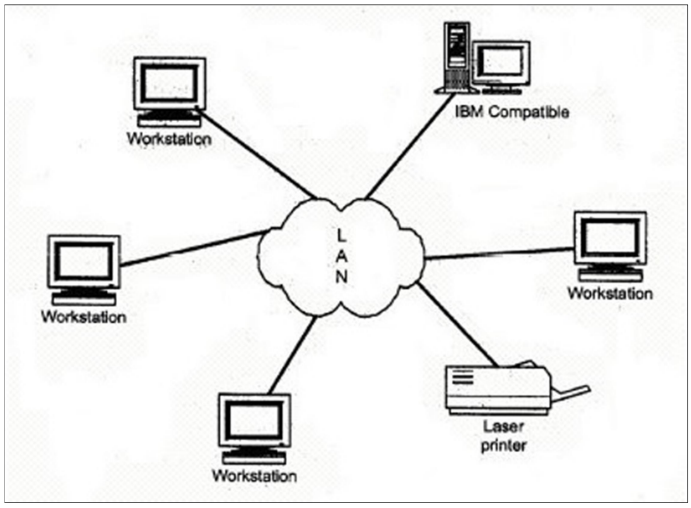
2.1.1 Jaringan Berdasarkan Jangkauan

Dilihat dari ruang lingkup jangkauannya, jaringan komputer dibedakan menjadi: *LocalArea Network* (LAN), *Metropolitan Area Network* (MAN), dan *Wide Area Network* (WAN). (Wahidin : Hal 1 : 2007)

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) dan saling bertukar informasi.

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.

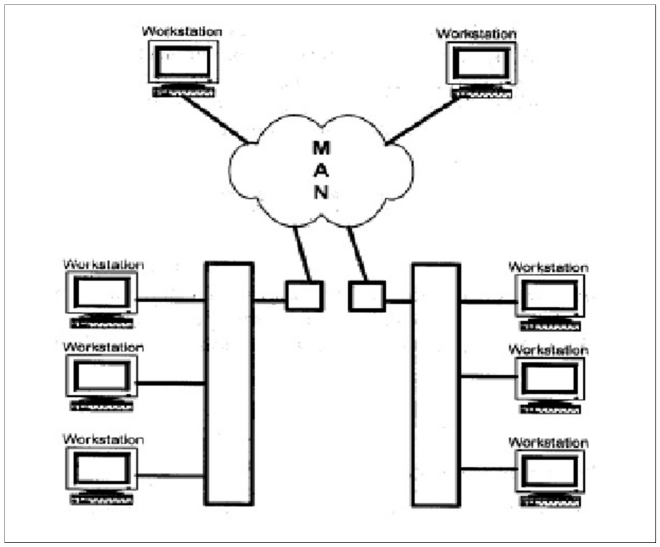


Gambar 2.1 Skema Jaringan LAN

Seperti halnya saluran pelanggan telepon yang dipakai di daerah pedesaan. LAN seringkali menggunakan teknologi transmisi kabel tunggal. LAN tradisional beroperasi pada kecepatan mulai 10 sampai 100 Mbps dengan *delay* rendah (puluhan *microsecond*) dan mempunyai faktor kesalahan yang kecil. LAN-LAN modern dapat beroperasi pada kecepatan yang lebih tinggi, sampai ratusan megabit/detik.

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 letaknya berdekatan atau juga sebuah kota dan dapat dimanfaatkan untuk keperluan pribadi (swasta) atau umum. MAN 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 kabel output. Adanya elemen switching membuat rancangan menjadi sederhana. (Andrew S. Tanenbaum : Hal 8 : 1997)

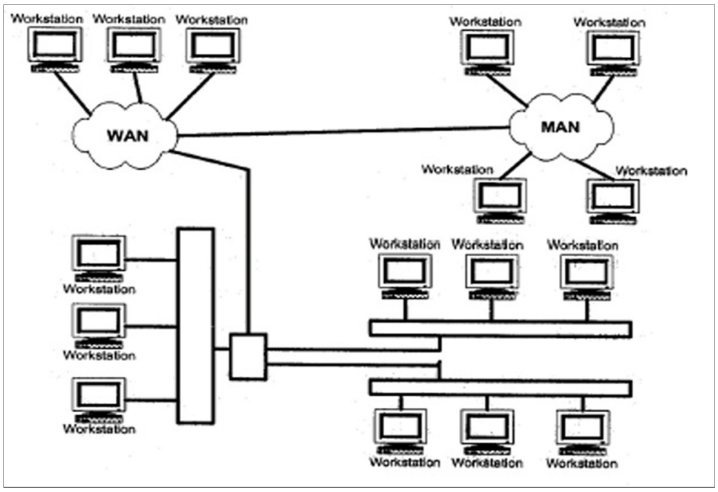


Gambar 2.2 Skema Jaringan MAN

2.1.1.3 ***Wide Area Network (WAN)***

Wide Area Network (WAN) merupakan jaringan yang lebih besar dari MAN dan mencakup daerah geografis yang luas, seringkali mencakup sebuah negara atau benua. WAN terdiri dari kumpulan mesin yang bertujuan unuk menjalankan program-program (aplikasi) pemakai.

Pada sebagian besar WAN, jaringan terdiri dari sejumlah banyak kabel atau saluran telepon yang menghubungkan sepasang router. Bila dua router yang tidak menggunakan kabel yang sama akan melakukan komunikasi, keduanya harus berkomunikasi secara tidak langsung melalui router lainnya. Ketika sebuah paket dikirimkan dari sebuah router ke router lainnya melalui sebuah router perantara atau lebih, maka paket akan diterima router perantara dalam keadaan lengkap, disimpan sampai saluran output menjadi bebas, dan kemudian baru diteruskan. (Andrew S. Tanenbaum : Hal 8-9 : 1997)



Gambar 2.3 Skema Jaringan WAN

2.1.2 Jaringan Berdasarkan Pengakesan Data

Sedangkan jika dilihat dari cara pengaksesan data, jaringan komputer terbagi menjadi: *Client-Server* dan *Peer to Peer.*

2.1.2.1 *Client Server*

Jenis jaringan ini terdapat komputer yang bertindak sebagai server dan komputer yang berperan sebagai *client* (*workstation*). Secara normal komputer server dapat mengontrol sepenuhnya komputer *client*. Server biasanya mempunyai kemampuan yang lebih besar dari pada *client.*

2.1.2.2 *Peer to Peer*

Peer to Peer merupakan jenis jaringan yang tidak memerlukan server secara khusus, karena komputer yang terhubung pada jaringan dapat bertindak sebagai server ataupun *client*. Contoh: hubungan komputer ke komputer. (Wahidin : Hal 1 : 2007)

2.1.3 Jaringan Berdasarkan Media Transmisi

Media adalah suatu bentuk atau benda yang digunakan untuk menyampaikan informasi atau pesan, sedangkan transmisi adalah komunikasi data

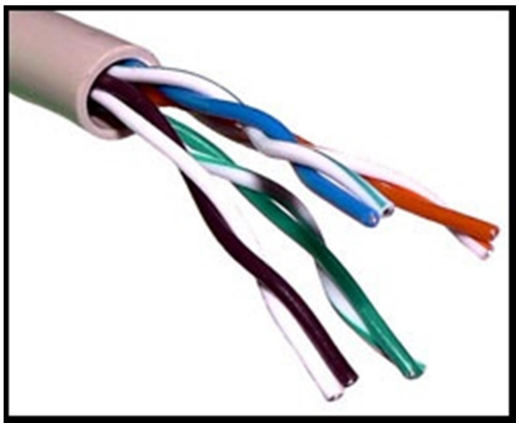
melalui penyebaran dan pemrosesan sinyal-sinyal. Sehingga media transmisi dapat diartikan perangkat yang digunakan sebagai jalur transmisi dari data yang dikirimkan. Berdasarkan media transmisinya, jaringan komputer dapat dibagi menjadi dua jenis, yaitu *Wire Network* dan *Wireless Network*

2.1.3.1 *Wire Network (Guided Media)*

Wire Network adalah jaringan komputer yang menggunakan kabel sebagai media transmisinya, kabel yang digunakan pun bermacam-macam, antara lain:

a. Kabel *Unshielded Twisted Pair*

   Unshielded twisted pair yaitu jenis kabel ini terbuat dari bahan penghantar tembaga, mempunyai isolasi dari plastik & terbungkus oleh bahan isolasi yang dapat melindungi dari api dan juga kerusakan fisik, kabel UTP sendiri terdiri dari 4 pasang inti kabel yang saling berbelit dimana masing-masing pasang mempunyai kode warna berbeda.

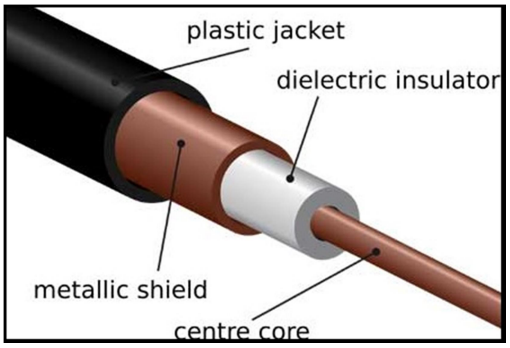


Gambar 2.4 Kabel UTP

b. Kabel *Coaxial*

   Coaxial adalah suatu jenis kabel yang menggunakan dua buah konduktor. Pusatnya berupa inti kawat padat yang dilingkupi oleh sekat yang kemudian dililiti lagi oleh kawat berselaput konduktor. Jenis kabel ini biasa digunakan untuk jaringan dengan *bandwith* yang tinggi. Kabel coaxial mempunyai pengalir tembaga di tengah (*centre core*). Lapisan plastik (*dielectric*

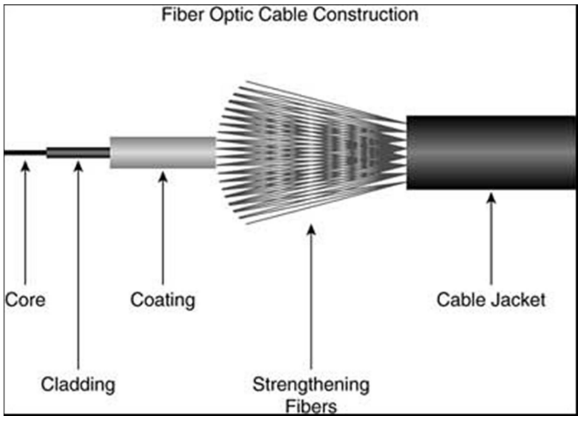
insulator) yang mengelilingi tembaga berfungsi sebagai penebat di antara tembaga dan lapisan metal (*metallic shield)*. Lapisan metal berfungsi untuk menghalang sembarang gangguan luar, sedangkan lapisan paling luar adalah lapisan plastik yang disebut *jacket plastic*. Lapisan ini berfungsi seperti jaket yaitu sebagai pelindung bagian terluar.



Gambar 2.5 Kabel Coaxial

c. Kabel *Fiber Optic*

Kabel *Fiber optic* adalah sebuah kabel yang terbuat dari serat kaca dengan teknologi canggih dan mempunyai kecepatan transfer data yang lebih cepat daripada kabel biasa, biasanya *fiber optic* digunakan pada jaringan *backbone* (Tulang Punggung) karena dibutuhkan kecepatan yang lebih dalam jaringan ini,namun pada saat ini sudah banyak yang menggunakan *fiber optic* untuk jaringan biasa baik LAN, WAN maupun MAN karena dapat memberikan dampak yang lebih pada kecepatan dan bandwith karena *fiber optic* ini menggunakan bias cahaya untuk mentransfer data yang melewatinya dan sudah barang tentu kecepatan cahaya tidak diragukan lagi namun untuk membangun jaringan dengan *fiber optic* dibutuhkan biaya yang cukup mahal dikarenakan dibutuhkan alat khusus dalam pembangunannya.



Gambar 2.6 Kabel Fiber Optic

2.1.3.2 *Wireless Nerwork (Unguided Media)*

Wireless Network adalah jaringan komputer yang menggunakan media gelombang radio, *infrared,* ataupun *microwave* sebagai media transmisinya. Spektrum yang digunakan pada *wireless network* antara 3kHz hingga 900 THz.

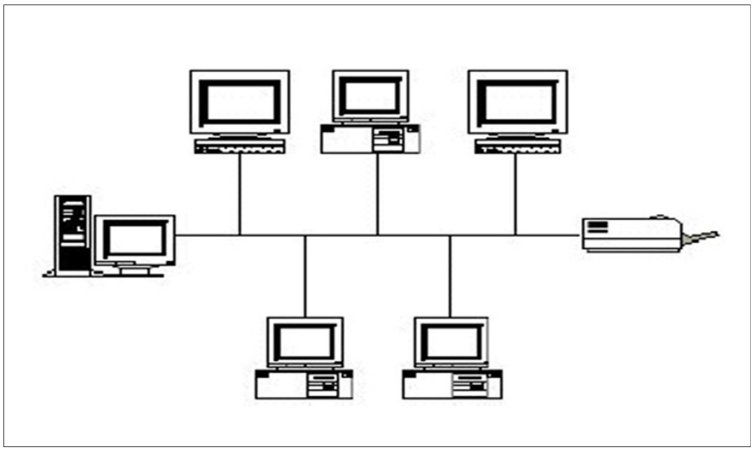
2.2 Topologi Jaringan Komputer

Topologi adalah suatu cara menghubungkan komputer yang satu dengan komputer lainnya sehingga membentuk jaringan. Masing-masing topologi ini mempunyai ciri khas, dengan kelebihan dan kekurangannya. Disini akan dijelaskan beberapa topologi jaringan, seperti: topologi Bus, topologi *Ring* dan topologi *Star*. (Wahana Komputer : Hal 6-7 : 2003)

2.2.1 Topologi Bus

Layout ini termasuk layout umum. Satu kabel utama menghubungkan tiap simpul ke saluran tunggal komputer yang mengaksesnya ujung dengan ujung. Masing-masing simpul dihubungkan ke dua simpul lainnya, kecuali komputer di salah satu ujung kabel, yang masing-masing hanya terhubung ke satu simpul lainnya. Topologi ini seringkali dijumpai pada sistem *client*/server, di mana salah satu komputer pada jaringan tersebut difungsikan sebagai file server, yang berarti bahwa komputer tersebut dikhususkan hanya untuk pendistribusian data dan

biasanya tidak digunakan untuk pemrosesan informasi. Dengan kata lain, pada topologi jenis ini semua terminal terhubung ke jalur komunikasi. Informasi yang akan dikirim akan melewati semua terminal pada jalur tersebut. Jika alamat yang tercantum dalam data atau informasi yang dikirim sesuai dengan alamat terminal yang dilewati, maka data atau informasi tersebut akan diterima dan diproses. Jika alamat tersebut tidak sesuai, maka informasi tersebut akan diabaikan oleh terminal yang dilewati.



Gambar 2.7 Skema Topologi Bus

Berikut adalah kelebihan dan kekurangan dalam menggunakan jaringan topologi bus :

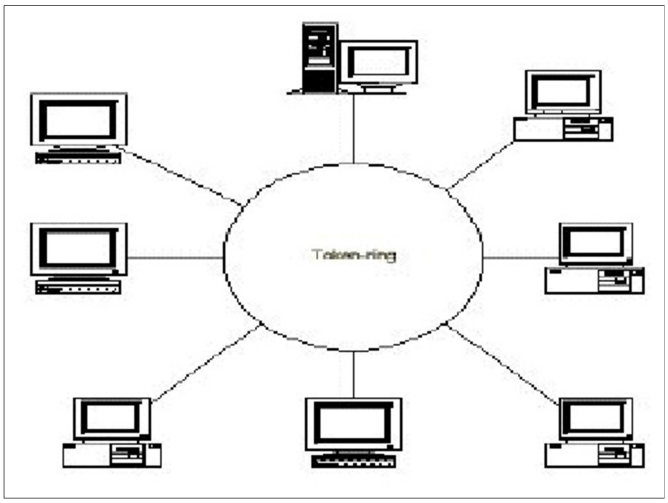
Tabel 2.1 Kelebihan dan Kekurangan Pada Topologi Bus

Kelebihan	Kekurangan
1. Mudah dikembangkan. 2. Jarak LAN tidak terbatas. 3. Keterandalan jaringan tinggi. 4. Kecepatan pengiriman tinggi. 5. Jumlah terminal dapat ditambah ataupun dikurangi tanpa mengganggu operasi yang telah berjalan. 6. Tidak diperlukan pengendalian	1. Jika *traffic* data terlalu tinggi atau padat dapat terjadi kemacetan (*congestion*). 2. Diperlukan *repeater* untuk menguatkan sinyal pada pemasangan jarak jauh. 3. Oprasional jaringan LAN bergantung pada setiap terminal. 4. Jika terjadi gangguan atau

pusat. 7. Kondusif untuk konfigurasi jaringan pada gedung bertingkat	kerusakan pada salah satu *node* dalam jaringan maka akan mempengaruhi jaringan secara keseluruhan, bahkan ada kemungkinan jaringan akan terhenti.

2.2.2 Topologi *Ring*

Topologi ini mirip dengan topologi Bus, tetapi kedua terminal yang berada di ujung saling dihubungkan, sehingga menyerupai seperti lingkaran. Setiap informasi yang diperoleh diperiksa alamatnya oleh terminal yang dilewatinya. Jika bukan untuknya, informasi dilewatkan sampai menemukan alamat yang benar. Setiap terminal dalam jaringan saling tergantung, sehingga jika ada kerusakan pada satu terminal maka seluruh jaringan akan terganggu.



Gambar 2.8 Skema Topologi Ring

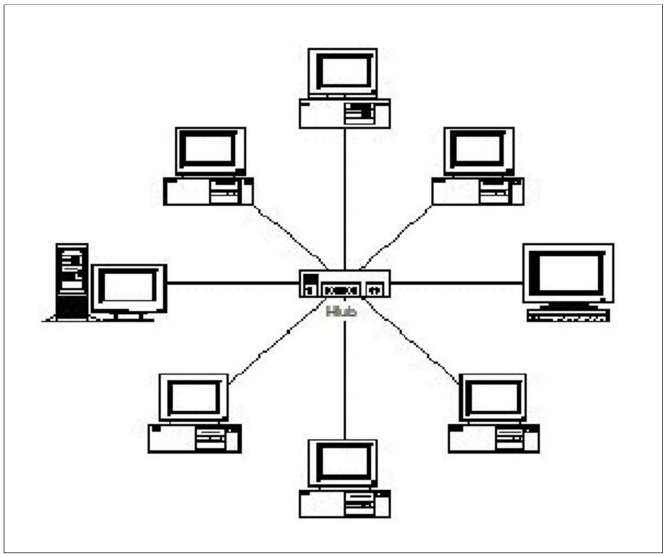
Tabel 2.2 Kelebihan Dan Kekurangan Topologi Ring

Kelebihan	Kekurangan
1. Laju data (*transfer rate*) tinggi. 2. Dapat melayani *traffic* data yang	1. Penambahan atau pengurangan terminal cukup sulit.

padat. 3. Tidak diperlukan *host*, relatif lebih mudah. 4. Dapat melayani berbagai media pengiriman. 5. Komunikasi antar terminal mudah. 6. Waktu yang diperlukan untuk mengakses data optimal.	2. Kerusakan pada media pengiriman data menghentikan kerja seluruh jaringan. 3. Harus ada kemampuan untuk mendeteksi kesalahan dan metode pengisolasian kesalahan. 4. Tidak kondusif untuk pengiriman suara, gambar, dan data.

2.2.3 Topologi *Star*

Dalam topologi *Star*, sebuah terminal pusat bertindak sebagai pengatur dan pengendali semua komunikasi data yang terjadi. Terminal-terminal lain terhubung padanya dan pengiriman data dari satu terminal ke terminal lainnya melalui terminal pusat. Terminal pusat menyediakan jalur komunikasi khusus untuk dua terminal yang akan berkomunikasi. Dengan kata lain semua kontrol dipusatkan pada satu komputer yang dinamakan stasiun primer dan komputer lainnya dinamakan stasiun sekunder. Setelah hubungan jaringan dimulai, setiap stasiun sekunder dapat sewaktu-waktu menggunakan hubungan jaringan tersebut tanpa menunggu perintah dari stasiun primer.



Gambar 2.9 Skema Topologi Star

Tabel 2.3 Kelebihan Dan Kekurangan Topologi Star

Kelebihan	Kekurangan
1. Kehandalan topologi ini lebih besar dari topologi lainnya. 2. Mudah dikembangkan. 3. Keamanan data tinggi. 4. Kemudahan akses ke jaringan LAN lainnya.	1. *Traffic* data yang padat menyebabkan jaringan lambat. 2. jaringan tergantung pada terminal pusat atau konsentrator.

2.3 Open System Interconnection (OSI)

Open Systems Interconnection (OSI) adalah upaya standarisasi jaringan komputer yang dimulai pada tahun 1982 oleh *International Organization for Standardization* (ISO) bersama *International Telecommunication Union Telecommunication Standardization Sector* (ITU-T).

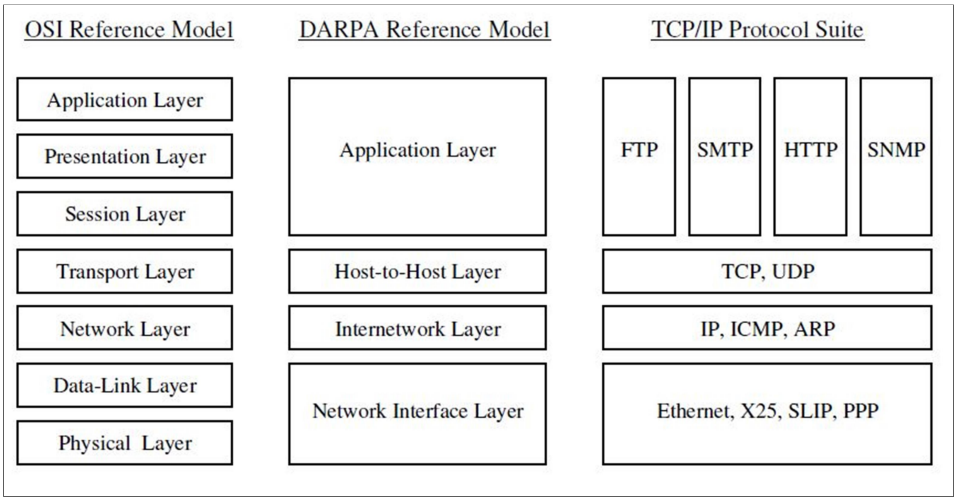
Sebelum OSI, sistem jaringan komputer sangat tergantung kepada pemasok (vendor). OSI berupaya membentuk standar umum jaringan komputer untuk menunjang interoperatibilitas antar pemasok yang berbeda. Dalam suatu

jaringan yang besar biasanya terdapat banyak protokol jaringan yang berbeda. Tidak adanya suatu protokol yang sama, membuat banyak perangkat tidak bisa saling berkomunikasi.

Model referensi jaringan terbuka OSI atau *Reference Model for open networking* adalah sebuah model arsitektural yang dikembangkan oleh badan *International Organization for Standardization (ISO)* di Eropa pada tahun 1974. Model referensi ini pada awalnya ditujukan sebagai basis untuk mengembangkan protokol-protokol jaringan, meski pada kenyataannya inisiatif ini mengalami kegagalan. Kegagalan itu disebabkan oleh beberapa faktor berikut:

1. Standar model referensi ini, jika dibandingkan dengan model referensi DARPA (model internet) yang dikembangkan oleh Internet Engineering Task Force (IETF), sangat berdekatan. Model DARPA adalah model basis protokol TCP/IP yang populer digunakan.
2. Model referensi ini dianggap sangat kompleks. Beberapa fungsi (seperti halnya metode komunikasi *connectionless)* dianggap kurang bagus, sementara fungsi lainnya (seperti *flow control* dan koreksi kesalahan) diulang-ulang pada beberapa lapisan. Pertumbuhan internet dan protokol TCP/IP (sebuah protokol jaringan dunia nyata) membuat OSI Reference Model menjadi kurang diminati.

OSI Reference Model pun akhirnya dilihat sebagai sebuah model ideal dari koneksi logis yang harus terjadi agar komunikasi data dalam jaringan dapat berlangsung. Beberapa protokol yang digunakan dalam dunia nyata, semacam TCP/IP, DECnet dan IBM *Systems Network Architecture* (SNA) memetakan tumpukan protokol *(protocol stack)* mereka ke OSI *Reference Model*. OSI *Reference Model* pun digunakan sebagai titik awal untuk mempelajari bagaimana beberapa protokol jaringan di dalam sebuah kumpulan protokol dapat berfungsi dan berinteraksi.



Gambar 2.10 Hubungan antara OSI, DARPA, dan TCP/IP Protocol Suite

Berikut adalah penjelasan dan fungsi-fungsi setiap *layer* pada model OSI :

1. *Physical Layer* (*Layer* 1)

   Layer ini berfungsi untuk mendefinisikan media transmisi jaringan, metode pensinyalan, sinkronisasi bit, arsitektur jaringan (seperti *Ethernet* atau *token ring*), dan penglabelan. Selain itu, *layer* ini juga mendefinisikan bagaimana *Network Interface Card* (NIC) dapat berinteraksi dengan media kabel atau radio.

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

   Layer ini mendefinisikan bagaimana cara mengirimkan data melalui suatu media, baik media kabel maupun nirkabel dengan *physical addressing*. Tugas utama dari *layer* ini adalah *error checking, flow control,* dan *media access control*untuk mengatur paket yang akan dikirim.

3. *Network Layer* (*Layer* 3)

   Layer ini berfungsi untuk mendefinisikan alamat-alamat IP, membuat *header* untuk paket-paket dan mencari jalur terbaik kemudian melakukan *routing* melalui *internetworking* dengan menggunakan *router* dan *switch* agar data sampai ditujuan dengan benar.

4. *Transport Layer* (*Layer* 4)

   Layer ini mengatur keutuhan data, menerima data dari *session layer* dan meneruskannya ke *network layer* juga sebaliknya. Lapisan ini berfungsi untuk memecah data ke dalam paket-paket data serta memberi nomor urut ke paket-paket tersebut sehingga dapat disusun kembali pada sisi tujuan setelah diterima.

5. *Session Layer* (*Layer* 5)

   Layer ini menyiapkan saluran komunikasi dan terminal dalam hubungan antar terminal, menkoordinasikan proses pengiriman dan penerimaan serta mengatur pertukaran data. Lapisan ini membangun, mempertahankan dan menyelaraskan interaksi antara sistem komunikasi.

6. *Presentation Layer* (*Layer* 6)

   Layer ini bertugas untuk mendefinisikan format data, menampilkan data dan menangani kompresi dan enkripsi data.

7. *Application Layer* (*Layer* 7)

   Layer paling atas ini mengatur interaksi pengguna komputer dengan program aplikasi yang dipakai. *Layer* ini juga mengatur bagaimana aplikasi dapat mengakses jaringan.

2.3.1 Konsep Dasar TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) adalah bukan sebuah protokol tunggal tetapi satu kesatuan protokol dan utility. Setiap protokol dalam kesatuan ini memiliki aturan yang spesifik. Protokol ini dikembangkan oleh ARPA (Advance Research Project Agency) untuk departemen pertahanan Amerika Serikat pada tahun 1969. ARPA menginginkan sebuah protokol yang memiliki karakter sebagai berikut:

1. Mampu menghubungkan berbagai jenis sistem operasi.
2. Dapat diandalkan dan mampu mendukung komunikasi kecepatan tinggi.
3. Routabel dan scalable untuk memenuhi jaringan yang kompleks dan luas.

2.3.2 Dasar Arsitektur TCP/IP

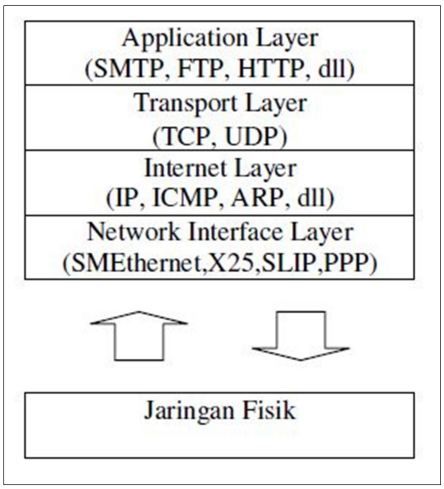
Pada dasarnya, komunikasi data merupakan proses mengirimkan data dari satu komputer ke komputer yang lain. Untuk dapat mengirimkan data, pada komputer harus ditambahkan alat khusus, yang dikenal sebagai *network interface* (*interface* jaringan). Jenis *interface* jaringan ini bermacam-macam, bergantung pada media fisik tersebut. Dalam proses pengiriman data ini terdapat beberapa masalah yang harus dipecahkan. Pertama, data harus dapat dikirimkan ke komputer yang tepat, sesuai tujuannya. Hal ini akan menjadi rumit jika komputer tujuan transfer data ini tidak berada pada jaringan lokal, melainkan di tempat yang jauh. Jika lokasi komputer yang saling berkomunikasi jauh (secara jaringan) maka terdapat kemungkinan data rusak atau hilang. Karenanya, perlu ada mekanisme yang mencegah rusaknya data ini. Hal lain yang perlu diperhatikan ialah, pada komputer tujuan transfer data mungkin terdapat lebih dari satu aplikasi yang menunggu datangnya data. Data yang dikim harus sampai ke aplikasi yang tepat, pada komputer yang tepat, tanpa kesalahan.

Untuk menangani semua masalah komunikasi data, keseluruhan aturan-aturan yang telah tersusun harus bekerja sama satu dengan yang lainnya. Sekumpulan aturan untuk mengatur proses pengiriman data ini disebut sebagai protokol komunikasi data. Protokol ini diimplementasikan dalam bentuk program komputer (*software*) yang terdapat pada komputer dan peralatan komunikasi data lainnya.

TCP/IP terdiri atas sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi data. Atas prinsip ini, tugas masing-masing protokol menjadi jelas dan lebih sederhana. Protokol yang satu tidak perlu mengetahui cara kerja protokol yang lain, sepanjang ia masih bisa saling mengirim dan menerima data. (Onno W. Purbo : Hal 21-22 : 1998)

Karena penggunaan prinsip ini, TCP/IP menjadi protokol komunikasi data yang fleksibel. Protokol TCP/IP dapat diterapkan dengan mudah disetiap jenis komputer dan interface jaringan, karena sebagian besar isi protokol ini tidak spesifik terhadap satu komputer atau peralatan jaringan tertentu. Agar TCP/IP

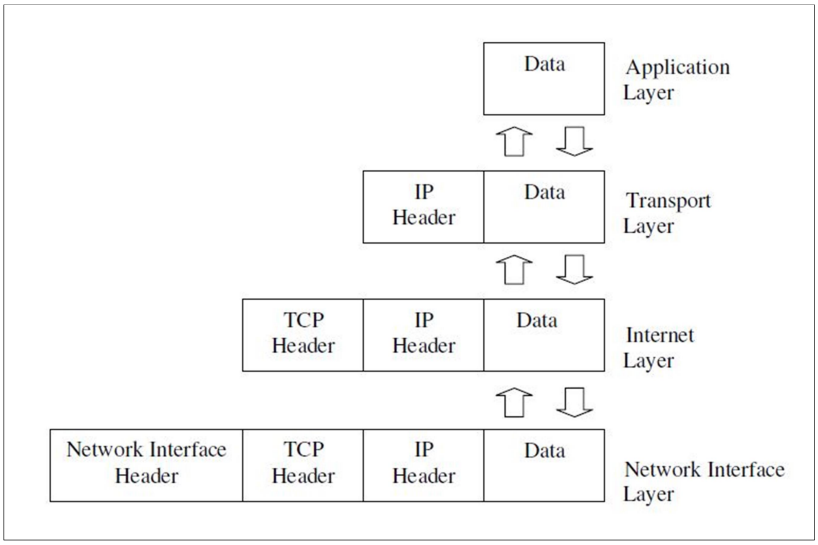
dapat berjalan di atas jaringan interface jaringan tertentu, hanya perlu dilakukan perubahan pada protokol yang berhubungan dengan interface jaringan saja.



Gambar 2.11 Layer TCP/IP

Dalam TCP/IP, terjadi penyimpangan data dari protokol yang berada dalam satu layer ke protokol yang berada di layer lain. Setiap protokol memperlakukan semua informasi yang diterimanya pada protokol lain sebagai data. Jika suatu protokol menerima data dari protokol lain di layer atasnya, ia akan menambahkan informasi tambahan miliknya ke data tersebut. Informasi ini memiliki fungsi yang sesuai dengan protokol tersebut. Setelah itu, data ini akan diteruskan lagi ke protokol pada layer di bawahnya.

Hal yang sebaliknya terjadi jika suatu protokol menerima data dari protokol lain yang berada pada layer di bawahnya. Jika data ini dianggap *valid*, protokol akan melepas informasi tambahan tersebut, untuk kemudian meneruskan data itu ke protokol lain yang berbeda pada layer di atasnya. (Onno W. Purbo : Hal 24 : 1998)



Gambar 2.12 Pergerakan data dalam layer TCP/IP

TCP/IP terdiri atas empat lapis kumpulan protokol yang bertingkat. Setiap lapisan yang dimiliki oleh protocol suite TCP/IP diasosiakan dengan protokolnya masing-masing. Protokol utama dalam protokol TCP/IP adalah sebagai berikut:

1. *Application Layer* : bertanggung jawab untuk menyediakan akses kepada aplikasi terhadap layanan jaringan TCP/IP. Protokol ini mencakup protokol *Dynamic Host Configuration Protocol (DHCP), Domain Name System (DNS), Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), Telnet, Simple Mail transfer Protocol (SMTP), Simple NetworkManagement Protocol (SNMP),* dan masih banyak protokol lainnya. Dalam beberapa inplementasi *stack* protokol, seperti halnya Microsoft TCP/IP, protokol-protokol lapisan aplikasi berinteraksi dengan menggunakan antarmuka *Windows Sockets (winsock)* atau NetBIOS over TCP/IP (NetBT).
2. *Transport Layer* : berguna untuk membuat komunikasi menggunakan sesi koneksi yang bersifat *connection-oriented* atau *broadcast* yang bersifat *connectionless*. Protokol dalam lapian ini adalah *Transmission Control Protocol* (TCP) dan *User Datagram Protocol* (UDP).

3. *Internet Layer* : bertanggung jawab untuk melakukan pemetaan (routing) dan enkapsulasi paket-paket dan jaringan menjadi paket-paket IP. Protokol yang bekerja dalam lapisan ini adalah *Internet protocol* (IP), *Address Resolution Protocol* (ARP), *Internet Control Message protocol* (ICMP), dan *Internet Group Message Protocol* (IGMP).
4. *Network Interface Layer* : bertanggung jawab untuk meletakkan frame-frame jaringan di atas media jaringan yang digunakan. TCP/IP dapat bekerja dengan banyak teknologi transport, mulai dari teknologi transport dalam LAN (seperti halnya Ethertnet dan *Token Ring*).

2.3.3 *Internet Protocol Address Version 4* (IPv4) dan *Subnet Mask*

Dalam jaringan yang menggunakan protokol TCP/IP, setiap host akan memiliki alamat IP atau *IP address*. Format *IP address* adalah nilai biner berukuran 32 bit yang diberikan ke setiap host dalam jaringan. Nilai ini digunakan untuk mengenali jaringan di mana host tersebut dan mengenali nomor unik host bersangkutan di jaringan tertentu. Setiap host yang terhubung jadi satu pada sebuah internet work harus memiliki satu alamat unik TCP/IP. Konsep ini serupa dengan cara kantor pos mengantarkan surat. Setiap rumah di sepanjang jalan menggunakan nama jalan (nama jaringan) yang sama tetapi memiliki nomor rumah (nomor host) yang berbeda. Sewaktu-waktu komputer ingin mengirimkan data ke komputer lain, maka kiriman tersebut harus dilengkapi dengan alamat yang tepat. Jika tidak maka yang menerima atau jaringan akan kebingungan harus dikirim ke mana jaringan tersebut. Pemberian alamat ini menjadi tanggung jawab pengirim. Setiap alamat terbagi atas dua komponen, yaitu :

1. *Network ID* adalah bagian dari alamat IP yang mewakili jaringan fisik dari host (nama jalan rumah). Setiap komputer dalam segmen jaringan tertentu akan memiliki ID jaringan yang sama.

2. *Host ID* adalah bagian yang mewakili bagian individu dari alamat (nomor rumah). Bila komputer di segmen jaringan memiliki alamat, maka jaringan tersebut perlu tahu milik siapa paket itu.

Jika dilihat dari bentuknya, IP *address* terdiri atas 4 buah bilangan biner 8 bit. Nilai terbesar dari bilangan biner 8 bit ialah 255 ($=2^7+ 2^6+ 2^5+ 2^4+ 2^3+ 2^2+ 2^1+1$). Karena IP address terdiri atas 4 buah bilangan 8 bit, maka jumlah IP address yang tersedia ialah 255 x 255 x 255 x 255.

IP address sebanyak ini harus dibagi-bagikan ke seluruh pengguna jaringan internet di seluruh dunia.

Untuk mempermudah proses pembagiannya, IP *address* dikelompokkan dalam kelas-kelas. Dasar pertimbangan pembagian IP *address* ke dalam kelas-kelas adalah untuk memudahkan pendistribusian pendaftaran IP *address*. Dengan memberikan sebuah ruang nomor jaringan (beberapa blok IP address) kepada ISP (Internet Service Provider) di suatu area diasumsikan penanganan komunitas lokal tersebut akan lebih baik, dibandingkan dengan jika setiap pemakai individual harus meminta IP address ke otoritas pusat, yaitu Internet Assigned Numbers Authority (IANA).

IP address ini dikelompokkan dalam lima kelas : Kelas A, Kelas B, Kelas C, Kelas D, dan Kelas E. Perbedaan pada tiap kelas tersebut adalah pada ukuran dan jumlahnya. IP kelas A dipakai oleh sedikit jaringan namun jaringan ini memliki anggota yang besar. Kelas C dipakai oleh banyak jaringan, namun anggota masing-masing jaringan sedikit. Kelas D dan E juga didefinisikan, tetapi tidak digunakan dalam penggunaan normal. Kelas D diperuntukkan bagi jaringan multicast, dan kelas E untuk keperluan eksperimental. (Onno W. Purbo : Hal 65-66 : 1998)

Kelas	Rentang IP Address	Format Penulisan
A	1-126	1.0.0.0 s/d 126.255.255.255
B	128-191	128.0.0.0 s/d 191.255.255.255
C	192-223	192.0.0.0 s/d 223.255.255.255
D	224-239	224.0.0.0 s/d 239.255.255.255
E	240-255	240.0.0.0 s/d 254.255.255.255

Gambar 2.13 Kelas IP Address

Setiap komputer di sebuah jaringan biasanya ingin mengirim data langsung ke komputer lainnya. Komputer pengiriman harus memastikan bahwa si penerima berada di jaringan yang sama atau di luar itu. Subnet mask digunakan oleh protokol TCP/IP untuk menentukan bahwa *host* yang akan berkomunikasi berada di jaringan local yang sama atau berada di jaringan remote.

Tabel 2.4 Klasifikasi Subnet Mask

Class	***Subnet Mask***
A	255.0.0.0
B	255.255.0.0
C	255.255.255.0

2.4 Perangkat Jaringan

Perangkat jaringan merupakan alat atau piranti keras atau lunak yang digunakan untuk membangun suatu sistem jaringan. Masing-masing perangkat jaringan memiliki fungsi dan tujuan tersendiri didalam suatu sistem jaringan. Pemilihan perangkat-perangkat jaringan yang diperlukan dapat disesuaikan dengan kebutuhan sistem jaringan yang akan dibangun.

2.4.1 Switch

Switch adalah alat penghubung jaringan dengan *forwarding* berdasarkan alamat MAC. *Switch* membagi *collision domain*tetapi tidak membagi *broadcast domain. Switch* bekerja pada *layer 2* pada model OSI (*Data Link Layer*) da nada juga yang bekerja pada *layer* 3 (*Network layer*) pada model OSI. Perbedaan yang

mendasar antara *switch layer* 2 dan *switch layer* 3 adalah kemampuan *switch layer* 3 dapat melakuka proses *routing.*



Gambar 2.14 Switch

2.4.2 Hub

Hub mirip dengan *switch* yaitu sebagai konsentrator. Namun *hub* tidak "secerdas" *switch*. Jika informasi dikirim ke *host* target melalui hub maka informasi akan mengalir ke semua *host*. Kondisi semacam ini dapat menyebabkan beban *traffic* yang tinggi. Oleh sebab itu, sebuah *hub* biasanya hanya digunakan pada jaringan berskala kecil.



Gambar 2.15 Hub

2.4.3 Perbedaan Switch dan Hub

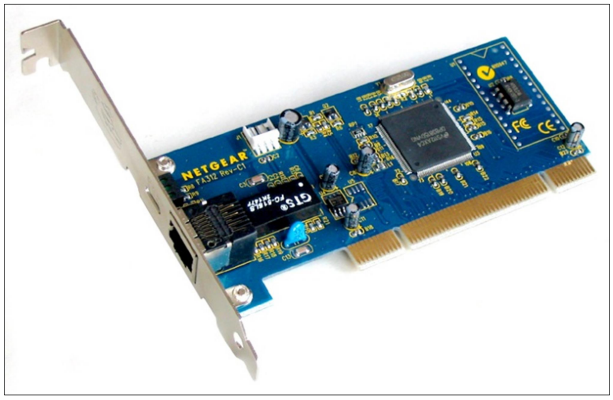
Perbedaan *hub* dan *switch*terletak dari bagaimana *packet data* / informasi yang dikirim kepada mereka diproses. Ketika data masuk atau datang ke *hub*, *hub* akan mengambil data tersebut dan akam mengtransmisikannya ke setiap komputer yang terhubung dalam jaringan. Tetapi lain halnya dengan *switch, switch* akan

menerima data tersebut dan hanya mengirimkan ke komputer yang berkepentingan menerima data tersebut.

Penggunaan *switch* akan memotong penggunaan *bandwith* jaringan secara signifikan, terutama bila kita memiliki jaringan dengan banyak komputer dan semua komputer tersebut sibuk untuk mengirim dan menerima data disaat yang bersamaan.

2.4.4 *Network Interface Card*

Kartu jaringan (NIC) merupakan salah satu dari perangkat keras yang menyediakan media untuk menghubungkan antar komputer. Kebanyakan kartu jaringan adalah kartu internal. Yaitu kartu jaringan yang dipasang pada slot ekspansi didalam komputer. Kartu jaringan yang banyak terpakai saat ini adalah kartu jaringan *Ethernet, local talk* konektor, dan kartu jaringan *token ring*.



Gambar 2.16 Network Interface Card

2.4.5 Router

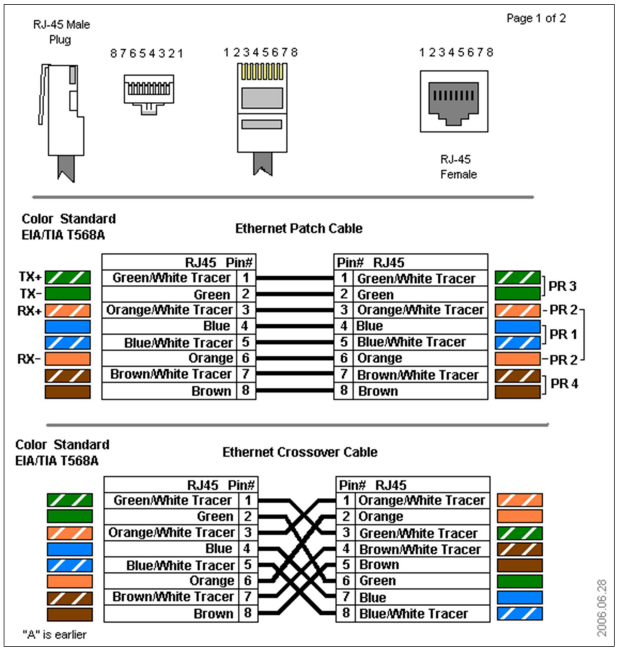
Router merupakan perangkat keras jaringan yang dapat digunakan untuk menghubungkan beberapa jaringan yang sama atau berbeda. *Router* juga dapat disebut sebagai alat untuk mengirimkan paket data melalui jaringan atau *internet* untuk dapat menuju tujuannya atau proses *routing*. *Routing* itu sendiri terjadi pada lapisan 3 dari OSI *layer. Router* terkadang digunakan untuk menghubungkan 2 buah jaringan yang menggunakan media berbeda, seperti dari Ethernet menuju ke token ring.



Gambar 2.17 Router

2.4.6 Konektor RJ45

Konektor RJ45 adalah penghubung atau konektor kabel *Ethernet* yang digunakan dalam jaringan. Konektor RJ45 biasanya terdapat pada ujung kabel UTP (*unshielded twisted pair*) yang menghubungkan ke pemancar penerima (*transciever*) atau kartu jaringan.dalam meng konfigurasikan konektor RJ45 terdapat dua konfigurasi yaitu *straight through* dan *cross over.*



Gambar 2.18 RJ45

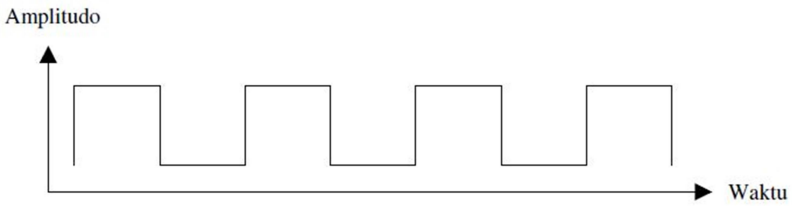
2.5 Sinyal Informasi

Data adalah suatu jenis informasi yang disimpan atau didapatkan kembali pada sebuah komputer. Oleh karena itu, jaringan mentransfer data dari satu komputer ke komputer yang lain. Data tersebut dapat berupa pesan *e-mail, file, web page, video*, musik dan lain sebagainya.

Sistem komunikasi jaringan pada komputer melambangkan data dengan menggunakan kode-kode yang diwakili secara efisien alat-alat elektronik dan gelombang radio. Sinyal tersebut membawa informasi melalui sistem dari satu titik ke titik yang lain. Sinyal tersebut dapat berupa sinyal digital atau analog. (Geier : 2005)

2.5.1 Sinyal Digital

Sinyal pada komputer memiliki irama amplitudo yang berubah dari waktu ke waktu. Sinyal digital pada gambar 2.19 biasanya berupa bilangan biner (dua digit), sehingga disebut dengan rangkaian digit biner (bit) atau data biner. Untaian digital dalam komputer dengan mudah menyimpan dan mengolah sinyal-sinyal digital tersebut ke dalam bentuk biner. (Geier : Hal 62 :2005)



Gambar 2.19 Sinyal Ideal Pada Komputer

Bilangan biner merupakan sebuah sistem yang hanya menggunakan 0 dan 1 untuk merepresentasikan angka-angka. Konversi dari sistem bilangan desimal ke bilangan biner mudah dijalankan tinggal menyimpan bilangan biner tersebut.

Salah satu kelebihan sinyal digital adalah lebih mudah diperbaiki. Saat sinyal merambat melalui medium udara, sinyal tersebut dapat berbenturan dengan suara atau gelombang yang dapat merubah sinyal. Untuk mengatasi memperbaiki sinyal tersebut, maka untaian digital dapat mendeteksi jika ada denyut dalam

periode waktu tertentu dan membuat denyut baru yang sama dengan denyut digital yang dikirim sebelumnya. Sinyal digital dapat menjangkau jarak jauh melalui periodik repeater sambil melindungi integritas informasi.

Berikut ini hal-hal yang menetapkan karakteristik penting pada sinyal digital:

1. Kecepatan Data

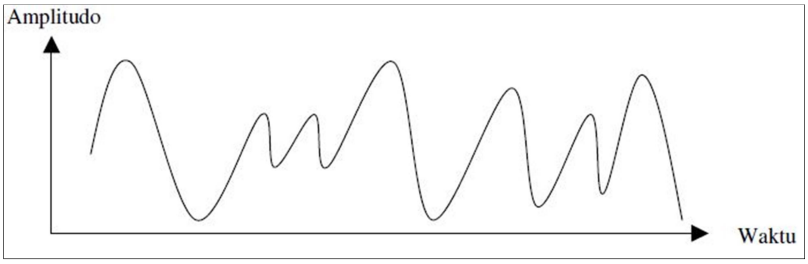
   Kecepatan data menyesuaikan dengan kecepatan yang ditransfer sinyal *digital*. Karena itu kecepatan data pada sinyal *digital* memberikan beberapa informasi mengenai lamanya pengiriman data dari satu titik ke titik yang lain dan mengidentifikasi jumlah bandwidth yang harus disuplai medium untuk mendukung sinyal secara efektif. Kecepatan data dari sinyal sama dengan waktu tempuh jumlah total *bit* yang ditransmisikan. Ukuran untuk kecepatan *bit* adalah *bits per second* (bps).

2. Throughput

   Throughput sama dengan kecepatan data, akan tetapi kalkulasi *throughput* mengabaikan bit-bit yang berhubungan dengan *overhead* pada protocol komunikasi. Tidak ada standar untuk merepresentasikan *throughput*, kecuali jika memasukkan informasi aktual yang dikirim menyebrangi jaringan. Karena itu, *throughput* memberikan solusi yang akurat untuk merepresentasikan performa dan efisiensi jaringan yang sebenarnya.

2.5.2 Sinyal Analog

Sinyal analog seperti ditunjukkan gambar 2.20, merupakan salah satu amplitudo sinyal yang berubah secara terus-menerus dari waktu ke waktu.



Gambar 2.20 Sinyal Analog Membawa Informasi Melalui Udara

Pada permulaan komunikasi elektronik, sebagian besar komunikasi elektronik mengolah sinyal dalam bentuk analog karena input informasinya berasal dari manusia. Sinyal analog memiliki amplitudo, *voltase*, energi, dan frekuensi. (Geier : Hal 65 : 2005)

2.6 Bentuk Komunikasi

Pada penulisan ini, peneliti akan menjelaskan dua jenis bentuk komunikasi, yaitu: *simplex* dan *duplex*.

2.6.1 Simplex

Simplex adalah salah satu bentuk komunikasi antara dua belah pihak, dimana sinyal-sinyal dikirim secara satu arah. Metode transmisi ini berbeda dengan metode *full-duplex* yang mampu mengirim sinyal dan menerima secara sekaligus dalam satu waktu atau half-duplex yang mampu mengirim sinyal dan meneriam sinyal meski tidak dalam satu waktu. Transmisi secara *simplex* terjadi di dalam beberapa teknologi komunikasi, seperti siaran televisi atau siaran radio.

Transmisi *simplex* tidak digunakan dalam komunikasi jaringan karena node-node dalam jaringan umumnya membutuhkan komunikasi secara dua arah. Memang, beberapa komunikasi dalam jaringan, seperti *video streaming*, terlihat seperti *simplex*, tapi sebenarnya lalu lintas komunikasi terjadi secara dua arah, apalagi jika protokol TCP yang digunakan protokol lapisan transportnya.

2.6.2 Duplex

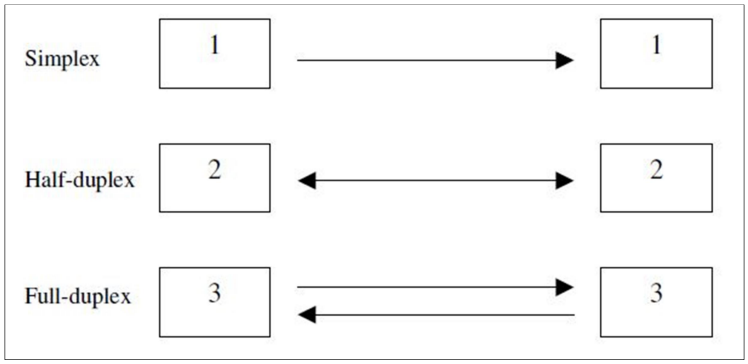
Duplex adalah sebuah istilah dalam bidang telekomunikasi yang merujuk kepada komunikasi dua arah. Terdapat dua metode duplexing, yaitu:

1. Half duplex

   Half duplex merupakan sebuah mode komunikasi dimana data dapat ditransmisikan atau diterima secara dua arah tapi tidak dapat secara bersama-sama. Contoh paling sederhana adalah *walkie-talkie*, dimana dua penggunanya harus menekan sebuah tombol untuk berbicara dan melepaskan tombol tersebut untuk mendengar, Ketika dua orang menggunakan *walkie-talkie* untuk berkomunikasi pada satu waktu tertentu, hanya salah satu diantara mereka yang dapat berbicara sementara pihak lainnya mendengar. Jika kedua-duanya mencoba untuk berbicara secara serentak, kondisi *collision* (tabrakan) pun terjadi dan kedua pengguna walkie-talkie tersebut tidak dapat saling mendengarkan apa yang keduanya kirimkan.

2. Full duplex

   Dalam komunikasi *full duplex*, dua pihak yang saling berkomunikasi akan mengirimkan informasi dan menerima informasi dalam waktu yang sama, dan umumnya membutuhkan dua jalur komunikasi. Komunikasi *full duplex* juga dapat diraih dengan menggunakan teknik *multiplexing*, dimana sinyal yang berjalan dengan arah yang berbeda akan diletakkan pada slot waktu (time slot) yang berbeda. Kelemahan teknik ini adalah bahwa teknik ini memotong kecepatan transmisi yang mungkin menjadi setengahnya.



Gambar 2.21 Cara kerja transmisi Simplex, Half-duplex, dan Full-duplex

2.7 Web Server

Web-server merupakan perangkat lunak yang melayani permintaan HTTP dari web-browser dan mengirimkan kode-kode dinamis ke server plikasi. Server aplikasi inilah yang menerjemahkan dan memproses kode-kode dinamis menjadi kode-kode statis HTML dalam suatu halaman statis yang kemudian dikirimkan ke browser oleh web-server. Web-server biasanya disebut juga sebagai HTTP server karena basisnya menggunakan protokol HTTP. Ketika user mengetik sebuah URL atau meng-klik sebuah link, maka web-browser akan mengirimkan permintaan halaman web. Melalui alamat URL, diketahui server mana yang menyimpan halaman web tersebut, selanjutnya server menerima permintaan dan mencari halaman web yang diminta, apabila halaman web yang diminta terdapat di webserver tersebut, maka server akan mengirimkan alaman web yang diminta melalui Jaringan Internet atau Intranet ke browser di komputer klien. Ketika halaman tersebut tiba di komputer, web-browser segera menerjemahkan bahasa hypertext dan menampilkannnya di komputer klien, proses ini hanya berjalan dalam hitungan detik. komputer klien dapat mengakses homepage dan halaman web secara online apabila terkoneksi ke suatu jaringan komputer, sebab web-server selalu dikoneksikan melalui jaringan komputer. Melalui jaringan inilah komputer klien dapat berinteraksi langsung dengan jutaan web-server yang tersebar di dunia, begitu juga dengan web-server yang akan melayani ribuan bahkan jutaan permintaan halaman web setiap harinya. (Kleinman : 2010)

Banyak sekali aplikasi *Web Server* saat ini baik yang berbayar ataupun yang gratis. Diantara web server yang gratis adalah :

1. Apache
2. Internet Information Services (IIS)
3. Nginx .

2.7.1 Web Server Apache

Apache merupakan *web server* yang paling banyak dipergunakan di Internet. Program ini pertama kali didesain untuk sistem operasi lingkungan UNIX. Namun demikian, pada beberapa versi berikutnya Apache mengeluarkan programnya yang dapat dijalankan di Windows NT. Berdasarkan sejarahnya, Apache dimulai oleh veteran developer NCSA httpd (*National Center for Supercomputing Application*). Saat itu pengembangan NCSA httpd sebagai web server mengalami stagnasi. ROB MC COOL meninggalkan NCSA dan memulai sebuah proyek baru bersama para webmaster lainnya, menambal bug, dan menambahkan fitur pada NCSA httpd. Mereka mengembangkan program ini lewat mailing list. Dengan berpijak pada NCSA httpd versi 1.3, Team Apache mengeluarkan rilis pertama kali secara resmi Apache versi 0.6.2. Tim inti pengembang Apache waktu itu :

1. BRIAN BEHLENDORF
2. ROY T. FIELDING
3. ROB HARTILL
4. DAVID ROBINSON
5. CLIFF SKOLNICK
6. RANDY TERBUSH
7. ROBERT S. THAU
8. ANDREWWILSON

Nama Apache diambil dari kata "*A Patchy Server*", server perbaikan yang penuh dengan tambalan *(patch)*. Tambalan yang dimaksud adalah penambahan fitur dan penambalan bug dari NCSA httpd Versi 1.3. Saat ini Apache dipergunakan secara luas. Hal ini disebabkan karena programnya yang gratis,

dengan kinerja relatif stabil. Dalam pengembangannya pun mempergunakan sistem Bazaar, yakni tiap orang dibuka kesempatan seluas-luasnya untuk dapat memberikan kontribusi dalam mengembangkan program. Apache mempunyai program pendukung yang cukup banyak.

Tabel 2.5 Kelebihan Dan Kekurangan Apache

Kelebihan	**Kekurangan**
Open Source	Mudah diserang oleh DoS (pada Apache versi 1.3 dan versi 2 sampai versi 2.0.36)
Mudah untuk dikustomisasi	Tidak bisa mengatur load seperti IIS sehingga akan terus memfork proses baru sampai dalam batas yang diijinkan OS. Akan memudahkan penyerang karena RAM akan cepat habis.
Bisa digunakan di berbagai platform mesin dari mainframe sampai embedded system	
Keamanannya bagus dan bisa menggunakan SSL (Secure Socket Layer)	

2.7.2 Web Server Nginx

Nginx merupakan *web server* dengan cepat memberikan konten statis dengan penggunaan efisien sumber daya sistem.Hal ini dapat menyebarkan dinamis HTTP konten di jaringan menggunakan FastCGI handler untuk script, dan dapat berfungsi sebagai perangkat lunak yang sangat mampu menjadi penyeimbang beban. Nginx menggunakan asynchronous -event pendekatan untuk menangani permintaan yang diprediksi memberikan kinerja yang lebih bawah beban, kontras dengan Apache HTTP server model yang menggunakan berulir atau proses yang berorientasi pada pendekatan-permintaan penanganan. (Kleinman : 2010)

Tabel 2.6 Kelebihan Dan Kekurangan Nginx

Kelebihan	**Kekurangan**
Performanya yang tinggi, stabil, memiliki banyak fitur	Fast-CGInya tidak berfungsi maximal
Menggunakan hanya sedikit sumberdaya pada server	Belum support IPV6
Mudah dikonfigurasi	
Tidak bergantung kepada thread untuk melayani klien	

2.7.3 Web Server IIS

Internet Information Service (IIS) adalah komponen yang dapat digunakan untuk mengelola web, File Transfer Protocol (FTP), Ghoper, dan NNTP. Komponen IIS terdapat pada sistem operasi Windows NT ,Windows 2000, Windows XP, 7 dan Windows Server 2003. Pada Windows 98, supaya dapat mengelola Web diperlukan komponen Personal Web Server (PWS) walaupun tidak terdapat fasilitas untuk FTP. PWS juga merupakan bagian dari IIS. Komponen PWS terdapat pada CD master Windows 98 dan terletak pada direktori \add-ons\pws. Instalasi PWS dapat dilakukan dari <Drive CD ROM>:\ADD-ONS\PWS\SETUP. (H Garnieri : 2010)

Tabel 2.7 Kelebihan Dan Kekurangan IIS

Kelebihan	**Kekurangan**
Lebih kompatibel dengan Windows	Tidak gratis (web server berbayar)
Unjuk kerja untuk PHP lebih stabil, handal, dan cepat	Port 80 (Port untuk layanan web) sangat mudah diserang oleh cracker
Untuk platform .NET, user hanya bisa menggunakan IIS	Keamanan file log juga sangat mudah ditembus sehingga system password pun akan mudah didapatkan
Memiliki score tertinggi dari web	

server lainnya untuk masalah bug	

2.8 Sistem Operasi Komputer

Sistem operasi komputer merupakan bagian yang terpenting dalam suatu sistem komputer, merupakan jantung dari komputer itu sendiri. Sistem operasi komputer merupakan program yang figunakan untuk mengendalikan sistem *input/output*, seperti *keyboard, diskdrive,* dan juga digunakan untuk membaca dan menjalankan program atau aplikasi. Sistem operasi juga merupakan sekumpulan mekanisme dan kebijakan yang membantu mendefinsikan pengendalian sumberdaya yang di-*share* (Sidik, hal 1 : 2004).

Sistem operasi dibedakan berdasarkan kemampuan untuk menangani proses dan pemakaian komputer pada saat yang bersamaan menjadi sistem operasi *standalone* dan *multiuser*. Pada sistem operasi *standalone*, komputer hanya dapat digunakan untuk satu pemakai saja pada saat yang bersamaan, tetapi proses yang di tanganinya bias lebih dari satu pada saat yang bersamaan. Sedangkan dengan sistem operasi *multiuser*, komputer dapat digunakan untuk melayani proses dari banyak pemakai pada saat yang bersamaan.

2.8.1 Windows

Microsoft Windows atau sering disebut dengan Windows merupakan sistem operasi yang dibuat dan dikembangkan oleh Microsoft Corporation, perusahaan perangkat lunak yang didirikan oleh Bill Gates pada tahun 1975 dan berkantor pusat di Redmond, Washington, Amerika Serikat.

Sistem operasi Windows merupakan pengembangan dari MS-DOS, sebuah sistem operasi berbasis modul teks dan command-line atau CLI (Command Line Interface). MS-DOS yang menjadi awal kesuksesan Microsoft berasal dari Q-DOS (Quick and Dirty Operating System) yang ditulis perusahaan pembuatan komputer Seattle Computer Products (SCP) dan dikepalai oleh Tim Paterson pada tahun 1980 dan dirancang untuk prosesor intel 8086. Tahun 1981 Microsoft membeli lisensi Q-DOS seharga US$ 50.000 dan mengganti nama sistem operasi tersebut menjadi MS-DOS, sementara yang dijual ke perusahaan komputer IBM

dan langsung dibundle dengan komputer diberi nama IBM PC-DOS (Internal Business Machine Personal Computer Disk Operating System). Windows pertama yang menggunakan sistem GUI adalah Windows Graphic Encirontment 1.0 atau yang biasa dikenal dengan Windows 1.0 dikenalkan pertama kali pada 10 novembar 1983. Windows 1.0 merupakan perangkat lunak 16-bit tambahan (bukan merupakan sistem operasi) yang berjalan di atas MS-DOS (dan beberapa varian dari MS-DOS), sehingga ia tidak dapat berjalan tanpa adanya sistem operasi DOS.



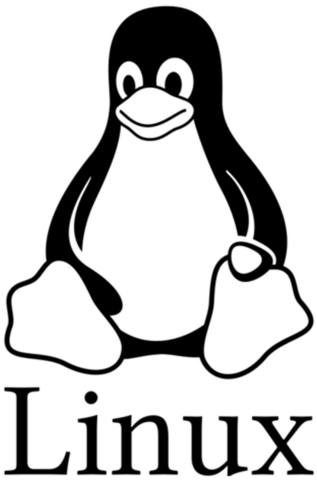
Gambar 2.22 Logo Windows

2.8.2 Linux

Linux pada awalnya dibuat oleh seorang mahasiswa Finlandia yang *bernama Linus Benedict Torvalds*. Dulunya Linux merupakan proyek hobi yang diinspirasikan dari Minix, yaitu sistem UNIX kecil yang dikembangkan oleh Andrew Tanenbaum. Linux *versi* 0.01 dikerjakan sekitar bulan Agustus 1991. Kemudian pada tanggal 5 Oktober 1991, Linus mengumumkan versi resmi Linux, yaitu *versi* 0.02 yang hanya dapat menjalankan *shell bash(GNU Bourne Again Shell*) dan gcc (*GNU C Compile*r).

Terdapat banyak distribusi Linux (lebih dikenali sebagai distro) yang dibuat oleh individu, grup, atau lembaga lain. Masing-masing disertakan dengan program sistem dan program aplikasi tambahan, di samping menyertakan suatu program yang memasang keseluruhan sistem di komputer (installer program). Inti

di setiap distribusi Linux adalah kernel, koleksi program dari proyek GNU (atau proyek lain), cangkang (shell), dan aturcara utilitas seperti pustaka (libraries), kompilator, dan penyunting (editor). Kebanyakan sistem juga menyertakan cara dan utilitas yang bukan-GNU. Bagaimanapun, utilitas tersebut dapat dipisahkan dan sistem ala UNIX masih tersedia. Beberapa contoh adalah aturcara dan utiliti dari BSD dan sistem grafik-X (X-Window System). X menyediakan antarmuka grafis (GUI) yang umum untuk Linux. Contoh-contoh distribusi Linux : Debian, Ubuntu, SuSE, Fedora, Mandriva, Slackware, Debian, PCLinuxOS, Knoppix, Xandros, CentOS. (Noris : 2009)



Gambar 2.23 Logo Linux

2.8.2.1 Red Hat

Red Hat Linux (RHL), buatan perusahaan Red Hat, Inc., adalah sistem operasi Linux yang populer sampai produksinya dihentikan pada tahun 2004. Red Hat Linux 1.0 ini dirilis pada 3 November 1994. Awalnya disebut "Red Hat Commercial Linux" merupakan distro linux pertama yang menggunakan sistem RPM Package Manager, yang kemudian diikuti beberapa distro lain, seperti Mandriva Linux dan SUSE Linux.

Sejak 2003, Red Hat telah menghentikan produksi Red Hat Linux namun mengeluarkan Red Hat Enterprise Linux (RHEL) untuk lingkungan perusahaan (tidak gratis), serta Fedora (yang dikembangkan oleh masyarakat dengan

dukungan Fedora Project dan disponsori oleh Red Hat) sebagai versi gratis bagi lingkungan rumahan. Red Hat Linux 9, rilis terakhir, secara resmi diakhiri pada 30 April 2004, meskipun dukungan masih diberikan oleh proyek Fedora Legacy sampai awal 2007.



Gambar 2.24 Logo Red Hat

2.8.2.2 CentOS

CentOS adalah singkatan dari *Community ENTerprise Operating System*, dikembangkan oleh sebuah komunitas kontributor dan pengguna sistem operasi Linux. Pengguna tipikal Centos adalah organisasi dan individu yang tidak terlalu membutuhkan layanan khusus *(strong commercial support)* untuk menjalankan sistemnya dengan sukses. CentOS merupakan 100% kompatibel *dengan Red Hat Enterprise Linux (RHEL)* karena merupakan kembaran dari Distro tersebut, dan _full compliance‘ dengan persyaratan redistribusi *Red Hat*. Dalam kondisi normal tidak akan ada tambahan patch terhadap paket-paket yang didapat dari distribusi asalnya. Perubahan paling mendasar akan dibuat agar sesuai dengan kebijakan pembuat distribusi asal berkaitan dengan merek dagang nama dan logo. Perubahan lain-lain akan selalu dijelaskan dalam *Release Notes* untuk masing-masing produk *CentOS. CentOS* ditargetkan untuk siapa saja yang membutuhkan stabilitas _enterprise class *operating system stability* tanpa biaya lisensi dan *subscription* (layanan dukungan) bila tidak dibutuhkan sertifikasi maupun dukungan dari *Red Hat.*



Gambar 2.25 Logo Centos

2.9 Shell Script

Shell script adalah suatu *file* yang berisi perintah-perintah yang harus dikerjakan oleh *shell* dan kemudian dieksekusi dengan cara memanggil *file* tersebut. Perintah dalam suatu *script* bisa berupa perintah yang dapat diberikan dari *prompt* atau perintah kendali alir (*flow control commands*). Perintah kendali alir adalah perintah untuk mengatur urutan eksekusi perintah-perintah dalam suatu *script*.

Shell script umumnya ditulis dengan menggunakan teks editor, misalnya vim, nano atau emacs, yang biasanya terdiri dari perintah-perintah dan komentar dan umumnya diberikan ekstensi .sh. Setiap *file script* biasanya diawali dengan baris seperti : #!/bin/bash yang artinya *script* dibawah baris tersebut akan dijalankan dengan interpreter /bin/bash. Komentar ditandai dengan tanda # dan terdiri dari teks yang memberitahu apa yang terjadi.



Gambar 2.26 Shell Script

2.10 PostgreSQL

PostgreSQL adalah sebuah sistem basis data yang disebarluaskan secara bebas menurut Perjanjian lisensi BSD. Piranti lunak ini merupakan salah satu basis data yang paling banyak digunakan saat ini, selain MySQL dan Oracle. PostgreSQL menyediakan fitur yang berguna untuk replikasi basis data. Fitur-fitur yang disediakan PostgreSQL antara lain DB Mirror, PGPool, Slony, PGCluster, dan lain-lain. PostgreSQL adalah sistem database yang kuat untuk urusan relasi, open source. Memiliki lebih dari 15 tahun pengembangan aktif dan sudah terbukti segala rancangan arsitekturnya telah mendapat reputasi tentang “kuat”, “handal”, “integritas data”, dan “akurasi data”.

2.11 Unison

Unison merupakan sebuah tool sinkronisasi file dua arah yang berjalan pada sistem operasi Unix (termasuk Linux, Mac OS X, dan Solaris) dan Windows. Hal ini memungkinkan melakukan replikasi dari kumpulan beberapa file dan direktori untuk disimpan pada *host* yang berbeda (atau pada tempat penyimpanan data yang berbeda (*storage*) pada *host* yang sama), diubah secara

terpisah, dan kemudian diperbaharui dengan menyebarkan perubahan di setiap replika yang lain.

2.12 Cron Job

Cron Jobs adalah software yang bertugas untuk melakukan penjadwalan yang memungkinkan mengatur tugas secara terjadwal. Jadi kita bisa men-setting tugas secara terjadwal seperti download file, download email dan lain sebagainya file konfigurasi yang khusus menangani command agar berjalan secara periodik pada jangka waktu yang sudah ditentukan. File crontab disimpan dimana daftar tugas dan instruksi lainnya tersimpan. Pengguna bisa membuat file crontab nya sendiri.

Berikut contoh penggunaan Cron Job :

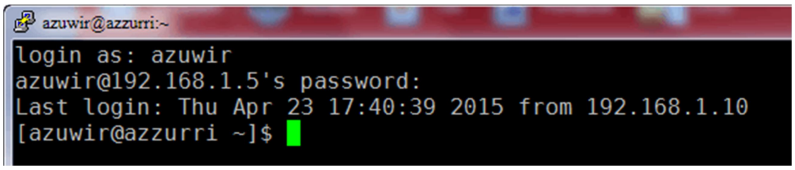
```
# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
0 */3 * * * /root/clearcache.sh
```

Pada command diatas mengartikan bahwa cronjob akan melakukan eksekusi bash script dengan file yang berada pada direktori /root/clearcache.sh setiap 3 menit sekali.

2.13 Open SSH

OpenSSH adalah satu set aplikasi komputer yang bisa mendukung sesi komunikasi terenkripsi pada jaringan komputer menggunakan protokol SSH. Awalnya aplikasi ini dikembangkan sebagai aplikasi open source yang menjadi alternatif dari aplikasi serupa yang berbayar. OpenSSH dikembangkan dan merupakan bagian dari proyek OpenBSD.

Dalam perkembangan selanjutnya, OpenSSH tidak hanya memberikan aplikasi remote shell melalui SSH atau remote eksekusi program saja, tapi bisa digunakan untuk tunnelling atau setup vpn antara dua jaringan.

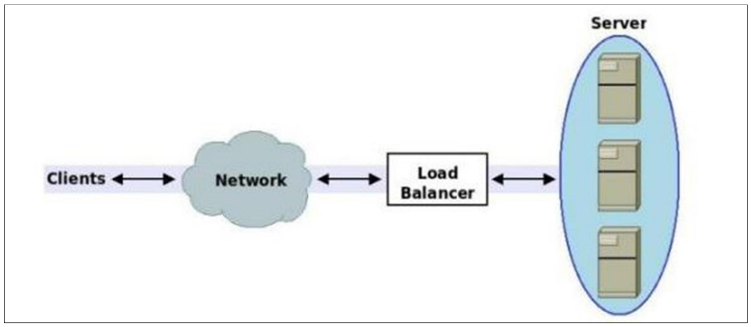


Gambar 2.27 Open SSH

2.14 Sistem Penyeimbang Beban *(Load Balancing)*

Penyeimbang beban (*Load Balancing*) adalah suatu proses dan teknologi yang mendistribusikan trafik situs di antara beberapa server dengan menggunakan perangkat berbasis jaringan. Sesuai kriteria dasar penyeimbang beban proses ini mampu mengurangi beban kerja setiap *server* sehingga tidak ada server yang kelebihan beban, memungkinkan *server* untuk menggunakan *bandwidth* yang tersedia secara lebih efektif, dan menyediakan akses yang cepat ke situs-situs yang *di-hosting*. Penyeimbang beban dapat diimplementasikan dengan menggunakan perangkat keras, perangkat lunak, atau gabungan keduanya.

Penyeimbang beban merupakan proses fleksibel yang dapat diciptakan dengan berbagai cara dan metode. Proses ini tidak dapat dilakukan oleh sebuah perangkat tertentu atau sebuah perangkat lunak khusus saja. Cukup banyak cara dan pilihan untuk mendapatkan jaringan yang dilengkapi dengan sistem penyeimbang beban. Cara kerja dan prosesnya berdeda-beda satu sama lainnya. Pada dasarnya penyeimbang beban adalah suatu teknik yang digunakan untuk memisahkan antara dua atau banyak *link* jaringan. Dengan mempunyai banyak *link* maka optimalisasi utilisasi sumber daya, *throughput* atau waktu respon akan semakin baik karena mempunyai lebih dari satu *link*



Gambar 2.28 Load Balancing

2.14.1 Algoritma Penjadwalan *Load Balancing*

Dalam memilih *real server* dari system *load balancing* diterapkan tiga algoritma penjadwalan, antara lain Round-Robin, Weighted Round-Robin, dan Least-Connection.

1. Penjadwalan Round-Robin

Algoritma penjadwalan Round-Robin, dalam arti kata, mengarahkan koneksi jaringan pada server yang berbeda dalam bentuk round-robin, yang memperlakukan semua *real server* sama menurut jumlah koneksi atau waktu respon.

2. Penjadwalan Weighted Round-Robin

Penjadwalan weighted round-robin ini memperlakukan *real server* dengan kapasitas proses yang berbeda. Masing-masing server dapat diberikan bobot, yaitu bilangan integer yang menunjukkan kapasitas proses (bobot awal adalah 1).

3. Penjadwalan Least-Connection

Algoritma penjadwalan least-connection mengarahkan koneksi jaringan pada server dengan jumlah koneksi aktif yang paling sedikit. Penjadwalan ini merupakan salah satu algoritma penjadwalan dinamik, karena memerlukan penghitungan koneksi aktif untuk masing-masing *real server* secara dinamik. Pada virtual server dimana terdapat sekumpulan server dengan kinerja yang mirip, penjadwalan least-connection baik untuk melancarkan pendistribusian ketika beban permintaan bervariasi banyak, karena semua permintaan panjang tidak akan mempunyai kesempatan untuk diarahkan pada server.

2.15 Reverse Proxy

Pengelompokan sistem penyeimbang beban terdapat beberapa jenis terdapat *proxy* yang merupakan bagian dari sistem penyeimbang beban yang bisa bertindak sebagai pembagi beban, *filtering*, dan *caching. Proxy* adalah sebuah system komputer atau program aplikasi yang melayani permintaan dari klien dengan meminta layanan ke *server* lain. *Proxy server* memiliki 3 fungsi utama yaitu :

1. *Connection sharing* : perantara klien dan *server*.
2. *Filtering* : bekerja pada layer aplikasi yang dapat mem-block paket-paket tertentu.
3. *Caching* : mampu menyimpan informasi yang pernah di akses dari server-server.

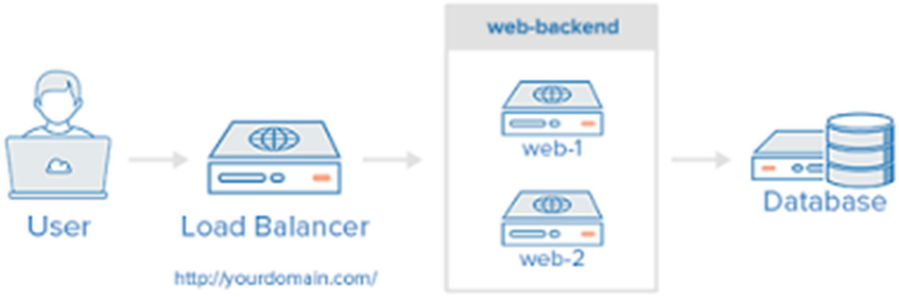
Proxy di bagi menjadi 2 yaitu *forward proxy* dan *reverse proxy. Proxy forward* adalah *proxy* yang meneruskan data ke *host* tujuan. *Reverse proxy* adalah *proxy* yang menampung permintaan klien dari Internet dan meneruskan ke *server-server* yang berada satu LAN *(local area network)* dengan *proxy* tersebut. Aplikasi yang berfungsi sebagai *reverse proxy c*ontohnya adalah Haproxy.

2.15.1 Haproxy

Haproxy adalah produk *opensource* yang mendukung keperluan penyeimbang beban dan *failover web server*, banyak digunakan untuk keperluan *reverse proxy* di *site-site* yang trafik hariannya tinggi. Pada Haproxy memiliki beberapa parameter di dalamnya yaitu sebagai berikut :

1. Global parameter : berisi parameter manajemen proses dan keamanan, *tuning* kinerja, *debugging*, dan *userlist*
2. *Proxy*
3. Konfigurasi *Server*
4. Manipulasi HTTP
5. *Accesslist*
6. *Logging*
7. Statistik dan monitoring

Berikut adalah contoh topologi dengan Haproxy :



Gambar 2.29 Mekanisme Haproxy

Keterangan pada gambar diatas user mengakses domain web yaitu disini adalah load balancer server, pada load balancer server sudah terinstall Haproxy. Haproxy akan membagi beban pada web-backend server yaitu web-1 dan web-2, lalu pada web-backend masing akan memproses apa yang user inginkan pada database.

Tabel 2.8 Kelebihan Dan Kekurangan Haproxy

Kelebihan	**Kekurangan**
Cepat	Hanya berjalan pada sistem operasi open source seperti linux, solaris, free BSD, Open BSD
Open Source (tidak berbayar)	Tidak dapat men-caching web
Dapat bekerja pada layer 4 (TCP) dan layer 7 (HTTP)	Hanya bertindak sebagai load balancer saja
Konfigurasi sangat mudah dalam *routing* dan *load balancing*	

2.16 Pgpool-II

Pgpool-II merupakan sebuah aplikasi pihak ketiga atau middleware yang bekerja diantara database PostgreSQL server dan aplikasi client yang memiliki kemampuan sebagai berikut :

1. *Connection Pooling*

   Pgpool-II akan membuat sekumpulan koneksi ke database PostgreSQL dan tidak akan memutus / mematikan koneksi itu. Apabila applikasi anda ingin berhubungan dengan database PostgreSQL, harus melalui pgpool-II ini. Jumlah koneksi yang dibuat oleh pgpool dapat di tentukan jumlahnya, dan akan selalu tetap jumlahnya sampai pgpool-II di restart. Koneksi yang ada ini dapat dipergunakan berulang ulang pada semua koneksi client yang terhubung ke pgpool-II. Hal ini akan mengurangi pemakaian resource yang berlebih dari proses terciptanya koneksi yang berulang ulang dan meningkatkan beban data yang lewat di dalam sistem secara keseluruhan.

2. *Replication*

   Pgpool-II dapat berperan sebagai alat untuk replikasi data memanfaatkan beberapa server PostgreSQL untuk menciptakan backup data secara realtime. Sehingga database akan selalu berfungsi jika terjadi kerusakan data.

3. *Load Balance*

   Jika database ter-replikasi, perintah SELECT pada server manapun akan memiliki hasil yang sama. pgpool-II akan memanfaatkan kemampuan replikasinya untuk menguraki beban dari setiap server dengan cara menyebar perintah SELECT ke beberapa servernya, hal ini akan meningkatkan kemampuan sistem. Load Balancing ini akan berfungsi dengan baik apabila terdapat banyak user yang mengeksekusi perintah sql secara bersamaan.

4. *Limiting Exceeding Connections*

   Ada pembatasan dalam membuat jumlah maksimal kumpulan koneksi ke database PostgreSQL. Meningkatkan jumlah maksimal koneksi ini akan berbanding lurus dengan besarnya pemakaian resource dan berpengaruh terhadap kemampuan sistem. Apabila jumlah koneksi yang ada telah dipergunakan semuanya, maka koneksi baru yang ada akan masuk dalam daftar antrian.

5. *Parallel Query*

   Fasilitas ini akan memecah perintah SELECT agar dapat di eksekusi ke semua server replikasi sehingga akan mempercepat waktu pemrosesan data. Parallel Query ini baik untuk mencari data dalam skala besar.

2.17 Siege Benchmark

Siege adalah aplikasi yang dibuat untuk keperluan stress test dan benchmark webserver. Stress test dan benchmarking diperlukan untuk mengetahui seberapa baik performa aplikasi web, dan seberapa banyak client atau koneksi yang bisa ditangani oleh aplikasi web. Contoh penggunaan Siege Bench :

```
[root@centos7 ~]# siege -b -c 20 -t 5M http://nama.domain.com/blog/
```

Keterangan :

1. -b = Benchmark, dengan pilihan ini tidak akan ada waktu jeda dari satu test ke test
2. -c = Banyaknya koneksi secara bersamaan yang akan dijalankan, default 10.
3. -t = Lama test akan dijalankan, dalam contoh di set 5 menit.

2.18 FirewallD

FirewallD di CentOS 7. Firewall merupakan tingkat dasar keamanan untuk server. Aplikasi ini bertanggung jawab untuk menolak traffic ke setiap port pada server dengan pengecualian untuk port/service yang telah diizinkan. Tidak seperti Linux CentOS 6, di CentOS 7 firewall disebut firewalld. Sebuah tool yang disebut firewall-cmd dapat digunakan untuk mengkonfigurasi kebijakan firewall Anda. Strategi dasar ini akan mengunci segala sesuatu aktivitas yang tidak memiliki alasan yang baik untuk mencoba mengakses server.

Contoh penggunaan perintah firewallD.

1. sudo firewall-cmd --permanent --remove-service=ssh
2. sudo firewall-cmd --permanent --add-port=5555/tcp