

BAB 2

LANDASAN TEORI

Pada bab 2 ini, kami akan memaparkan teori-teori umum dalam jaringan komputer dan teori-teori khusus yang berhubungan dengan topik skripsi kami yaitu simulasi jaringan e-KTP dengan menggunakan simulator NS-3. Teori umum jaringan komputer meliputi jenis jaringan, konsep jaringan, internet protocol, media jaringan, dan perangkat jaringan. Sedangkan teori khusus lebih menekankan pada teori-teori yang berhubungan dengan alat simulasi jaringan NS-3. Untuk lebih jelasnya, dapat dilihat pada subbab-subbab dibawah.

2.1 Pengertian Jaringan

Menurut Tanebaum, jaringan komputer merupakan penggabungan teknologi komputer dan berkomunikasi yang merupakan sekumpulan komputer berjumlah banyak yang terpisah-pisah, akan tetapi saling berhubungan dengan melaksanakan tugasnya.(Tanebaum,2003)

Jaringan komputer adalah sebuah sistem yang terdiri atas komputer dan perangakat jaringan lainnya yang bekerja bersama-sama untuk mencapai suatu tujuan yang sama. Tujuan dari jaringan komputer adalah:

- Membagi sumber daya, misalnya membagi printer, CPU, memori ataupun *harddisk*.

- Komunikasi, misalnya *e-mail, instant messaging, dan chatting*.
- Akses informasi, misalnya *web browsing*.

2.2 Klasifikasi Jaringan

Ada 3 macam jenis jaringan, yaitu:

1. *Local Area Network (LAN)*
2. *Metropolitan Area Network (MAN)*
3. *Wide Area Network (WAN)*

Saat ini, jaringan komputer terus berkembang semakin pesat. Hampir setiap perusahaan di berbagai bidang sudah menggunakan jaringan dalam kegiatan usaha mereka. Termasuk pada area perumahan di seluruh dunia untuk saling berkomunikasi satu sama lain.

1. *Local Area Network*

Local Area Network (LAN) adalah sebuah jaringan komputer yang dibatasi oleh area geografis yang relatif kecil dan umumnya dibatasi oleh area lingkungan seperti perkantoran atau sebuah sekolah dan biasanya ruang lingkup yang dicakupnya tidak lebih dari 2 km². Oleh karena itu, kecepatan komunikasinya lebih tinggi dan peluang terjadi kesalahan (*error rate*) lebih rendah. LAN muncul pada awal tahun 1970-an sebagai pengganti dari sistem komputer *mainframe*.

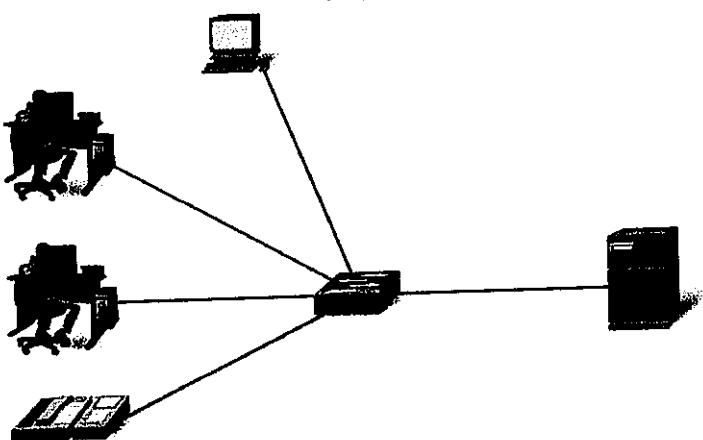
Dalam koneksi jaringan seperti itu, biasanya ada satu komputer yang dijadikan sebuah server. Server tersebut dapat digunakan untuk menyimpan berbagai macam piranti lunak (*software*) yang dapat digunakan untuk mengatur jaringan ataupun sebagai

piranti lunak yang dapat digunakan oleh komputer-komputer yang terhubung dalam jaringan tersebut (komputer klien / *workstation*).

LAN didesain untuk:

- beroperasi pada wilayah geografi yang terbatas,
- memungkinkan banyak user untuk mengakses media dengan kecepatan tinggi,
- menyediakan koneksi ke layanan lokal setiap saat (seperti printer dan file di server),
- menghubungkan peralatan yang berdekatan.

A network serving a home, building or campus is considered a Local Area Network (LAN).



Gambar 2.1 Topologi LAN

(<http://www.cisco.com>)

2. *Metropolitan Area Network*

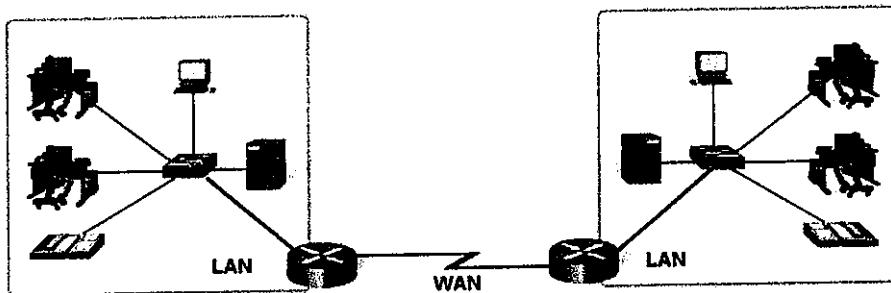
Sebuah MAN (*Metropolitan Area Network*), biasanya mencakup area yang lebih besar dari LAN, misalnya antar wilayah dalam satu propinsi. MAN juga dapat menghubungkan beberapa LAN menjadi suatu bagian jaringan yang lebih besar lagi. Cakupan geografis dari MAN itu sendiri tidak menghubungkan area geografis yang berbeda.

3. *Wide Area Networks*

Wide Area Networks (WAN) adalah jaringan yang ruang lingkupnya sudah terpisahkan oleh batas geografis dan biasanya sebagai penghubungnya sudah menggunakan media Satelit ataupun kabel bawah laut.

WAN muncul pada akhir tahun 1960-an, digunakan sebagai proyek riset akademis agar tersedia layanan komunikasi yang efektif antara situs, memperbolehkan berbagi *hardware* dan *software* secara ekonomis antar pengguna. WAN yang pertama kali didesain dan dikembangkan adalah Arpanet (*Advanced Research Project Agency Network*) yang pada akhirnya menjadi cikal bakal dari Internet. Situs-situs dalam WAN tersebut pada area geografis yang luas. Oleh karena itu, komunikasi berjalan relatif lambat dan reliabilitas tidak terjamin. Hubungan antara *link* yang satu dengan yang lain dalam jaringan diatur oleh *communication processor*.

LANs separated by geographic distance are connected by a network known as a Wide Area Network (WAN).



Gambar 2.2 Topologi WAN

(<http://www.cisco.com>)

WAN didesain untuk:

- beroperasi pada wilayah geografis yang sangat luas,
- memungkinkan akses melalui *interface* serial yang beroperasi pada kecepatan yang rendah,
- menyediakan koneksi *fulltime* dan *parttime*,
- menghubungkan peralatan yang dipisahkan oleh wilayah yang luas, bahkan secara global.

2.3 Konsep *Networking Model*

Pada saat *network* baru muncul, kebanyakan komputer hanya bisa berkomunikasi dengan komputer yang dibuat oleh perusahaan sama. Untuk itu, *International Organization for Standardization* membuat model referensi *Open System Interconnection*(OSI) sebagai solusi untuk mengatasi masalah kompatibilitas ini.

2.3.1 Pengenalan *Layer*

Konsep *layer* digunakan untuk menjelaskan bagaimana komputer berkomunikasi satu sama lainnya. Konsep *layer* menjelaskan bagaimana jaringan komputer mendistribusikan informasi dari sumber ke tujuan. Ketika komputer mengirimkan informasi melalui *network*, semua komunikasi di atur oleh sumber kemudian dikirimkan ke tempat tujuan.

Informasi yang ada dalam jaringan secara umum disebut dengan data atau paket. Sebuah paket secara logika merupakan sekumpulan unit informasi yang bergerak diantara sistem komputer. Setiap kali data melewati *layer*, informasi ditambahkan dari setiap *layer* yang akan mengefektifkan komunikasi dengan *layer* penerima pada komputer tujuan.

Model *Open System Interconnection* (OSI) dan *Transmission Control Protocol/Internet Protocol* (TCP/IP) memiliki *layer* yang menjelaskan bagaimana data mengalir dari komputer satu dengan komputer lainnya. Model tersebut memiliki perbedaan pada jumlah dan fungsi *layer* yang dimilikinya. Tetapi, setiap model dapat digunakan untuk menjelaskan dan menyediakan keterangan lengkap tentang aliran informasi dari sumber ke tujuan.

2.3.2 Model OSI *Layer*

Model Referensi OSI adalah model jaringan komputer yang paling utama, meskipun ada beberapa model jaringan komputer lainnya. Model Referensi OSI adalah model jaringan komputer yang paling banyak diadopsi oleh beberapa pembuat peralatan jaringan komunikasi.

Model Referensi OSI memungkinkan kita untuk melihat fungsi dari jaringan pada setiap *layer*-nya. Yang lebih penting lagi, model OSI adalah sebuah *framework* yang dapat mempermudah mengerti bagaimana perjalanan sebuah informasi ke dalam jaringan. Model Referensi OSI dapat digunakan untuk menvisualisasikan bagaimana informasi atau paket data berjalan di dalam aplikasi atau program melalui media komunikasi menuju ke aplikasi atau program lainnya yang terletak pada komputer yang lain pada satu jaringan meskipun penerima maupun pengirim memiliki media jaringan yang berbeda.

Pada setiap *layer* pada OSI, memiliki sebuah set fungsi yang dikerjakan agar suatu paket data dapat dikirim dari sumber ke tujuan pada suatu jaringan.

1. Layer 7 (Layer Application)

Layer ini adalah *layer* yang paling dekat dengan *user/pengguna*. *Layer* ini menyediakan sebuah layanan jaringan kepada pengguna aplikasi. *Layer* ini berbeda dengan *layer* lainnya yang dapat menyediakan layanan kepada *layer* lain. Sebagai contoh : program pengolah kata, *email*, FTP, dll.

2. Layer 6 (Layer Presentation)

Layer ini mengelola informasi yang disediakan oleh *layer* aplikasi (*application layer*) supaya informasi yang dikirimkan dapat dibaca oleh *layer* aplikasi pada sistem lain. Jika diperlukan, pada *layer* ini dapat menerjemahkan beberapa data format yang berbeda, kompresi, dan enkripsi.

3. Layer 5 (Layer Session)

Sesuai dengan namanya, *layer* ini berfungsi untuk menyelenggarakan, mengatur dan memutuskan sesi komunikasi. *Layer session* menyediakan servis kepada *layer presentation*. *Layer* ini juga mensinkronisasi dialog diantara dua *host layer presentation* dan mengatur pertukaran data.

4. Layer 4 (Layer Transport)

Layer transport mensegmentasi data dari pengirim dan merakit kembali data ke dalam sebuah data *stream* pada komputer penerima. Pada *layer* ini juga menyediakan servis komunikasi. Dalam menyediakan sebuah servis yang *reliable*, *layer* ini menyediakan *error detection* dan *recovery* serta *flow control*.

5. Layer 3 (Layer Network)

Layer network lebih sedikit kompleks dalam menyediakan koneksi. *Layer* ini dapat melakukan pemilihan jalur terbaik dalam komunikasi jaringan yang terpisah secara geografis (*Path Selection*).

6. Layer 2 (Layer Data Link)

Layer data link berfungsi menghasilkan alamat fisik (*physical addressing*), pesan-pesan kesalahan (*error notification*), pemesanan pengiriman data (*flow control*). Jika ingin mengambil istilah yang mudah diingat pada *layer* ini, gunakanlah istilah '*media access control*'.

7. Layer 1 (Layer Physical)

Layer physical berkaitan dengan karakteristik tinggi tegangan, periode perubahan tegangan, lebar jalur komunikasi (*bandwidth*), jarak maksimum komunikasi, konektor, dll. Jika ingin mengambil istilah yang mudah diingat pada *layer* ini, gunakanlah istilah '*signals*' atau '*media*'.

2.3.3 Model TCP/IP Layer

Model Referensi TCP/IP diciptakan oleh Departement Pertahanan Amerika (DARPA) karena mereka menginginkan jaringan yang dapat bertahan dalam kondisi apapun, sekalipun perang nuklir. Mereka menginginkan *network* yang dapat mengirimkan paket setiap saat, dalam kondisi apapun, dari satu titik ke titik lainnya. Masalah desain yang sangat sulit inilah yang menghasilkan Model TCP/IP, yang mana menjadi *standard* pertumbuhan internet.

Model TCP/IP Memiliki 4 *layer*: *Layer Aplikasi*, *Layer Transport*, *Layer Internet*, dan *Layer Network Access*. Penting untuk diperhatikan bahwa beberapa *Layer* pada Model TCP/IP memiliki nama yang sama dengan *layer* pada Model OSI.

1. Layer 4 (Layer Application)

Desainer TCP/IP merasa bahwa *level* protokol yang lebih tinggi harus memasukan detil *layer session* dan presentasi. Mereka hanya menciptakan

layer aplikasi yang menangani protokol tingkat tinggi, hal-hal mengenai representasi, *encoding*, dan dialog kontrol. TCP/IP menggabungkan semua hal – hal yang berhubungan dengan aplikasi menjadi satu *layer*. Dan memastikan data itu dibungkus dengan tepat untuk *layer* berikutnya.

2. Layer 3 (*Layer Transport*)

Layer transport berurusan dengan hal-hal mengenai kualitas layanan, reliabilitas, *flow control*, dan *error correction*. Layer ini memiliki dua protocol, yaitu TCP dan UDP. *Transmission Control Protokol* (TCP), menyediakan cara yang sempurna dan fleksible untuk menciptakan jaringan komunikasi yang dapat diandalkan, mengalir dengan baik, dan memiliki *error* yang rendah. TCP adalah protokol yang bersifat *Connection-Oriented*. TCP melakukan komunikasi antar sumber dan tujuan ketika membungkus informasi *layer application* menjadi unit yang disebut *segment*. *Connection-Oriented* tidak berarti bahwa ada sebuah *circuit* diantara komputer yang berkomunikasi, tapi memiliki arti bahwa *segment layer 4* berjalan bolak-balik diantara kedua *host* untuk mengetahui terdapat koneksi secara logika untuk beberapa waktu. Hal ini dikenal dengan nama *packet-switching*.

Sedangkan UDP adalah salah satu protokol lapisan transpor TCP/IP yang mendukung komunikasi yang tidak andal (*unreliable*), tanpa koneksi (*connectionless*) antara host-host dalam jaringan yang menggunakan TCP/IP.

UDP memiliki karakteristik-karakteristik berikut:

- *Connectionless* (tanpa koneksi): Pesan-pesan UDP akan dikirimkan tanpa harus dilakukan proses negosiasi koneksi antara dua host yang hendak berukar informasi.
- *Unreliable* (tidak andal): Pesan-pesan UDP akan dikirimkan sebagai datagram tanpa adanya nomor urut atau pesan acknowledgment. Protokol lapisan aplikasi yang berjalan di atas UDP harus melakukan pemulihan terhadap pesan-pesan yang hilang selama transmisi. Umumnya, protokol lapisan aplikasi yang berjalan di atas UDP mengimplementasikan layanan keandalan mereka masing-masing, atau mengirim pesan secara periodik atau dengan menggunakan waktu yang telah didefinisikan.
- UDP menyediakan mekanisme untuk mengirim pesan-pesan ke sebuah protokol lapisan aplikasi atau proses tertentu di dalam sebuah host dalam jaringan yang menggunakan TCP/IP. *Header* UDP berisi *field* Source Process Identification dan Destination Process Identification.
- UDP menyediakan penghitungan checksum berukuran 16-bit terhadap keseluruhan pesan UDP.

3. *Layer 2 (Layer Internet)*

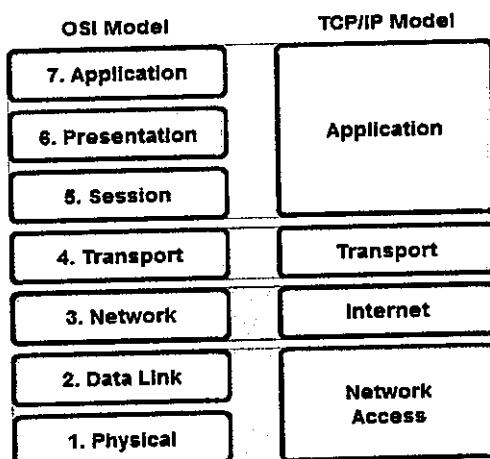
Tujuan dari *layer internet* adalah untuk mengirimkan paket dari sumber menuju jaringan manapun pada *internetwork* dan meyampaikannya di

tujuan dengan tidak tergantung pada jalur dan jaringan yang mereka ambil untuk sampai di sana. Protokol spesifik yang mengatur layer ini disebut *Internet Protocol* (IP). Pemilihan jalur terbaik dan *packet switching* terjadi pada *layer* ini.

4. Layer 1 (Layer Network Access)

Nama dari *layer* ini memiliki arti yang luas dan seringkali membingungkan. *Layer* ini disebut juga *layer host-to-network*. Adalah *layer* ini yang berhubungan dengan hal-hal yang memungkinkan paket IP untuk membuat hubungan secara fisik, dan kemudian membuat hubungan fisik lainnya. Termasuk didalamnya teknologi LAN dan WAN, dan semua detil dari *layer physical* dan *layer data link* pada *layer* OSI.

Comparing the OSI and TCP/IP models



The key parallels are in the Transport and Network layers.

Gambar 2.3 Perbandingan OSI dengan TCP/IP

(<http://www.cisco.com>)

2.4 Internet Protocol (IP)

Internet Protocol adalah protokol jaringan (*layer 3 / layer Network* pada OSI) yang digunakan di internet. Ketika sebuah informasi mengalir ke bawah pada *layer Model OSI*, data di enkapsulasi pada setiap *layer*. Pada *layer network*, data dienkapsulasi dalam paket – paket (atau disebut juga datagram), IP menentukan bentuk dari *header* paket (yang mana termasuk pengalamatan atau *addressing* dan informasi kontrol lainnya) tetapi tidak peduli mengenai data yang sebenarnya, dia menerima apapun yang di berikan oleh *layer* diatasnya.

Pada jaringan TCP/IP, sebuah komputer diidentifikasi dengan alamat IP. Tiap-tiap komputer memiliki alamat IP yang unik, masing-masing berbeda satu dengan yang lainnya. Hal ini dilakukan agar mencegah kesalahan pada transfer data. Secara umum protokol ini bertugas untuk menangani pendekripsi kesalahan pada transfer data.

Salah satu hal penting dalam IP, dalam pengiriman informasi adalah metode pengalamatan pengirim dan penerima. Saat ini terdapat standar pengalamatan yang sudah digunakan yaitu IPv4 dengan alamat terdiri dari 32 bit.

2.4.1 Pengalamatan IP

Pengalamatan bertujuan bagaimana supaya data yang dikirim sampai pada mesin yang sesuai dan bagaimana hal tersebut dapat dilakukan oleh operator dengan mudah. Untuk itu, maka data dari suatu *host* harus dilewatkan ke jaringan menuju *host* tujuan, dan dalam komputer tersebut data akan disampaikan ke *user* atau proses yang sesuai. Pengalamatan (IP Addressing) memiliki panjang 32 bit. Mengandung dua bagian utama, nomor *network* dan nomor *host*.

Karena hampir mustahil bagi orang untuk mengingat 32 *bit biner*, IP address di groupkan menjadi 8 *bit* per *group* sehingga menjadi 4 *group*, dipisahkan oleh titik, dan masing-masing *group* ditampilkan dalam desimal dan bukan format *biner*. Hal ini dikenal juga dengan format “*Dotted Decimal*”.

Ada 3 kelas yang dari pengalamatan IP yang bisa diterima oleh sebuah organisasi dari ARIN (*American Registry for Internet Number*). Kelas – kelas tersebut adalah Kelas A, B, dan C. ARIN menyediakan kelas A untuk pemerintah di seluruh dunia (Meskipun beberapa perusahaan besar seperti HP, dahulu pernah menerima satu) dan Kelas B ditunjukan untuk perusahaan berukuran menengah. Permintaan lainnya dimasukan kedalam pengalamatan Kelas C.

1. Kelas A

Ketika ditulis dalam format *biner*, *bit* pertama (yang paling kiri) dari alamat IP Kelas A selalu 0. *Oktet* (8 bit) pertama menandakan nomor *network* yang di berikan ARIN. *Internal Network Administrator* menetapkan 24 *bit* selanjutnya. Untuk mempermudah pengenalan alamat Kelas A, untuk kelas A *oktet* pertama selalu bernilai antara 0 – 126 (127 memang berawal *bit* 0,namun sudah di sediakan untuk tujuan *special* yaitu pengalamatan balik), semua alamat IP kelas A hanya menggunakan 8 *bit* pertama untuk mengidentifikasi *network* , 24 *bit* selanjutnya dapat digunakan untuk bagian alamat *host*. Setiap Jaringan yang menggunakan Alamat IP kelas A bisa memiliki 2 pangkat 24 minus 2 ($2^{24} - 2$) atau 16,777,214 Alamat IP yang mungkin yang dapat digunakan untuk peralatan yang terhubung ke jaringan tersebut.

2. Kelas B

Dua *bit* pertama dari Alamat kelas B selalu 10 (satu dan nol). Contoh dari kelas B adalah 143.234.23.8. Dua *oktet* pertama mengidentifikasi nomor *network* yang diberikan ARIN. *Internal Network Administrator* kemudian dapat menentukan 16 *bit* berikutnya. *Oktet* pertama dari alamat kelas B selalu bernilai antara 128 – 191. Alamat IP kelas B menggunakan 16 *bit* pertama untuk mengidentifikasi nomor *network* dan 16 *bit* selanjutnya dapat digunakan untuk bagian *host*. Setiap jaringan yang menggunakan kelas B dapat memiliki 2 pangkat 16 *minus* 2 atau 65,534 alamat IP yang mungkin digunakan untuk peralatan yang tersambung ke jaringan tersebut.

3. Kelas C

Tiga *bit* pertama dari alamat IP kelas C selalu 110(satu, satu , nol). Contoh alamat IP kelas C adalah 194.23.23.4. Tiga *oktet* pertama mengidentifikasi nomor *network* yang diberikan ARIN. *Internal Network Administrator* dapat menentukan 8 *bit* selanjutnya. Nilai *oktet* pertama Alamat IP kelas C selalu bernilai antara 192 – 223. Setiap jaringan yang menggunakan alamat IP kelas C dapat memiliki 2 pangkat 8 *minus* 2 atau 254 alamat IP yang mungkin untuk peralatan yang tersambung ke jaringan tersebut. Untuk setiap kelas diatas, jumlah alamat IP yang dapat dibuat selalu dikurangi dua, hal ini karena jika seluruh bagian *host* bernilai 0 (misalnya 10.0.0.0 untuk kelas A), maka alamat IP sama dengan alamat *network* sehingga tidak bisa digunakan sebagai alamat IP *host*. Dan jika seluruh bagian *host* bernilai satu (misalnya

10.255.255.255 untuk kelas A) maka alamat tersebut khusus digunakan untuk *broadcast* dalam jaringan tersebut.

2.4.2 Pemakaian IP

Alamat IP yang berada didalam ruangan alamat pribadi dikenal juga dengan alamat pribadi atau *private address/IP private*. Adapun IP *private* tersebut yaitu:

- 10.0.0.0 sampai 10.255.255.255 (10.0.0.0 /8)

Jaringan pribadi ini memiliki 24 *bit host* yang dapat digunakan untuk skema *subnetting* di dalam sebuah organisasi *private*.

- 172.16.0.0 sampai 172.31.255.255 (172.16.0.0 /12)

Jaringan pribadi ini memiliki 20 *bit* yang dapat ditetapkan sebagai *host identifier*, yang dapat digunakan dengan menggunakan skema *subnetting identifier* yang dapat digunakan dengan menggunakan skema *subnetting identifier* didalam sebuah organisasi *private*.

- 192.168.0.0 sampai 192.168.255.255 (192.168.0.0 /16)

Jaringan pribadi ini memiliki 16 *bit* yang dapat ditetapkan sebagai *host identifier* yang dapat digunakan dengan menggunakan skema *subnetting identifier* apapun di dalam sebuah organisasi *private*.

2.5 Jenis-jenis Media Transmisi

Media transmisi adalah suatu jalur antara pemancar dan penerima dalam sistem transmisi data. Media transmisi dapat diklasifikasikan menjadi *guided* (dengan perantara) dan *unguided* (tanpa perantara). Jenis media transmisi *guided* yaitu:

1. Kabel UTP

UTP merupakan jenis kabel jaringan yang menggunakan bahan dasar tembaga, yang tidak dilengkapi dengan *shield* internal. UTP merupakan jenis kabel yang paling umum yang sering digunakan di dalam jaringan lokal (LAN), karena memang harganya yang rendah, fleksibel dan kinerja yang ditunjukkannya relatif bagus.

2. 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 *bandwidth* yang tinggi.

3. *Fiber Optik*

Fiber optik merupakan saluran transmisi atau sejenis kabel yang terbuat dari kaca atau plastik yang sangat halus dan lebih kecil dari sehelai rambut, dan dapat digunakan untuk mentransmisikan sinyal cahaya dari suatu tempat ke tempat lain.

Sedangkan media transmisi *unguided* yaitu:

1. Gelombang Mikro

Gelombang mikro atau Mikrogelombang (*microwave*) adalah gelombang elektromagnetik dengan frekuensi super tinggi (*Super High Frequency, SHF*).

2. Gelombang Mikro Satelit

Satelit komunikasi adalah sebuah stasiun *relay* gelombang mikro. Dipergunakan untuk menghubungkan dua atau lebih *transmitter/receiver* gelombang mikro pada bumi, yang dikenal sebagai stasiun bumi atau *ground station*.

3. Radio *Broadcast*

Perbedaan-perbedaan utama diantara siaran radio dan gelombang mikro yaitu, dimana siaran radio bersifat segala arah (*broadcast*) sedangkan gelombang mikro searah (*point-to-point*). Karena itu, siaran radio tidak memerlukan antena parabola, dan antena tidak perlu mengarah ke arah persis sumber siaran.

4. Sinar Inframerah

Inframerah adalah radiasi elektromagnetik dari panjang gelombang lebih panjang dari cahaya tampak, tetapi lebih pendek dari radiasi gelombang radio. Komunikasi inframerah dicapai dengan menggunakan *transmitter/receiver (transceiver)* yang modulasi cahaya yang *koheren*.

Dengan mempertimbangkan desain sistem transmisi data, perhatian ditekankan pada *rate* data dan jarak. Semakin besar *rate* data dan semakin kecil jarak, maka akan

semakin baik. Sejumlah faktor-faktor perancangan yang berkaitan dengan media transmisi dan sinyal yang menentukan *rate* data dan jarak adalah :

- *Bandwidth*: Selama faktor lain tetap konstan, maka semakin besar *bandwidth* suatu sinyal, akan semakin tinggi *rate* data yang diperoleh
- Gangguan Sinyal: Gangguan, seperti misalnya *atenuasi*, membatasi jarak.
- Interferensi: Interferensi dari sinyal-sinyal yang berkompetisi dalam *band* frekuensi yang saling bertumpuk dapat mengubah atau menghapuskan sinyal.
- Jumlah *Receiver*: Suatu media *guided* bisa dipergunakan untuk membangun suatu hubungan titik ke titik atau hubungan terbagi pada alat-alat tambahan. Masing-masing alat tambahan akan memunculkan beberapa *atenuasi* dan distorsi dengan segera, serta membatasi jarak dan/atau *rate* data.

2.6 Perangkat Jaringan

Dalam sebuah jaringan komputer terdapat *device-device* yang berfungsi sebagai penghubung antar *device* ataupun dengan jaringan lainnya. Adapun *device-device* tersebut yaitu:

1. PC

Merupakan alat yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan.

2. Bridge

Bridge berjalan pada lapisan kedua pada model OSI (lapisan data-link), dan menggunakan skema pengalaman yang digunakan pada lapisan itu,

yakni MAC *address*.

3. Router

Router berjalan pada lapisan ketiga pada model OSI (lapisan jaringan), dan menggunakan skema pengalamatan yang digunakan pada lapisan itu, seperti halnya alamat IP.

4. Server

Merupakan sebuah sistem komputer yang menyediakan jenis layanan (*service*) tertentu dalam sebuah jaringan komputer.

2.7 Topologi Jaringan

Topologi dapat didefinisikan sebagai struktur dari sebuah jaringan. Ada dua bagian penting dari definisi topologi :

- *Physical Topology*, dimana kondisi sebenarnya jaringan dihubungkan secara langsung.
- *Logical Topology*, dimana didefinisikan bagaimana cara media jaringan dapat diakses oleh komputer.

1. *Physical Topology*

Physical Topology (topologi fisik) yang sering dipakai adalah *Bus*, *Ring*, *Star*, *Extended Star*, *Hierarchical*, dan *Mesh*.

- a. *Bus Topology* menggunakan "single backbone segment" sebagai

penghubung semua komputer yang ada pada jaringan. Semua komputer tersebut terhubung secara langsung ke kabel tersebut.

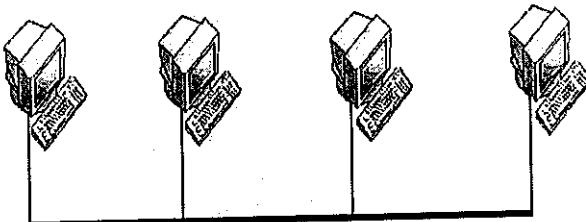
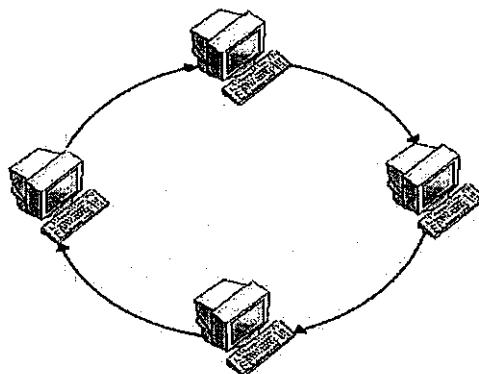


Figure 1: Bus topology

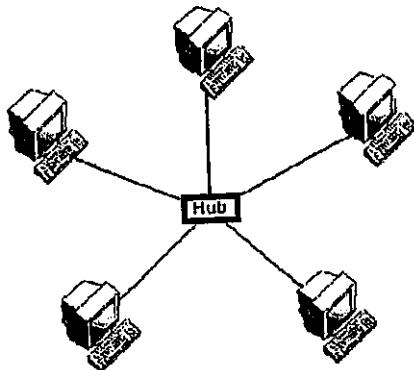
Gambar 2.4 Topologi Bus

- b. *Ring Topology* menghubungkan satu komputer dengan komputer berikut, dan seterusnya sehingga komputer paling akhir akan kembali terhubung ke komputer yang pertama (akan membentuk seolah-olah menjadi sebuah bentuk lingkaran/cincin).



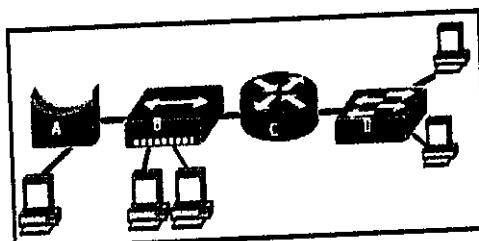
Gambar 2.5 Topologi Ring

- c. *Star Topology* menghubungkan semua kabel ke satu buah titik pusat. Titik pusat ini biasanya berupa hub atau switch sehingga seolah-olah komputer yang terhubung berbentuk seperti bintang.



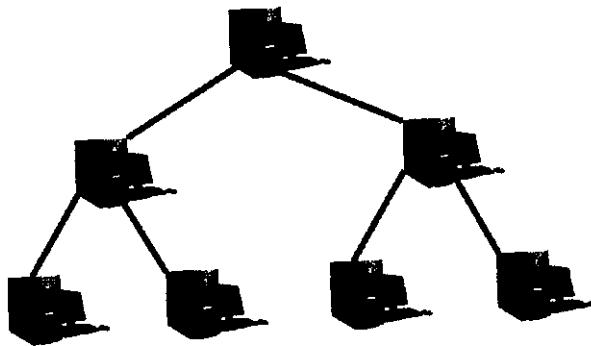
Gambar 2.6 Topologi Star

- d. *Extended Star Topology* menggabungkan beberapa topologi star menjadi satu. Hub/Switch yang dipakai untuk menghubungkan beberapa komputer pada satu jaringan dengan menggunakan topologi *star*, akan dihubungkan lagi ke hub/switch utama.



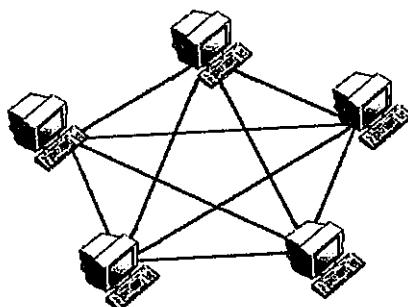
Gambar 2.7 Topologi Extended Star

- e. *Hierarchical Topology* dibuat mirip dengan topologi *extended star* tetapi pada system jaringan yang dihubungkan dapat mengontrol arus data pada topologi.



Gambar 2.8 Topologi Hierarki

- f. *Mesh Topology* digunakan ketika dibutuhkan dibutuhan dalam jaringan yang sangat penting yang tidak boleh ada sedikitpun kesalahan dalam komunikasi, contohnya system kontrol pembangkit tenaga nuklir. Jadi seperti yang bisa anda liat pada gambar. Setiap host memiliki hubungan langsung dengan semua *host* lainnya dalam jaringan. Hal ini juga merefleksikan internet, yang memiliki banyak jalur ke satu titik.



Gambar 2.9 Topologi Mesh

2. Logical Topology

Topologi logikal dari jaringan adalah bagaimana sebuah *host* berkomunikasi melalui *medium*. Dua tipe topologi logikal yang sering digunakan adalah *Broadcast* dan *Token Passing*.

- a. Topologi *Broadcast* berarti setiap *host* yang mengirim paket akan mengirimkan paket ke semua *host* pada media komunikasi jaringan. Tidak ada aturan khusus siapa yang akan menggunakan jaringan berikutnya. Peraturannya sederhana “yang pertama datang , yang pertama dilayani” (FCFS) dan ini adalah bagaimana cara *ethernet* bekerja.
- b. *Token-passing*, mengendalikan akses jaringan dengan mempass-kan sebuah token elektronik yang secara sekuensial akan melalui masing-masing anggota dari jaringan tersebut. Ketika sebuah komputer mendapatkan *token* tersebut, berarti komputer tersebut diperbolehkan mengirimkan data pada jaringan. Jika komputer tersebut tidak memiliki data yang akan dikirim, maka *token* akan dilewatkan ke komputer berikutnya. Kejadian akan berulang-ulang terus.

2.8 Pengertian Simulasi

Simulasi merupakan suatu teknik meniru operasi-operasi atau proses-proses yang terjadi dalam suatu sistem dengan bantuan perangkat komputer dan dilandasi oleh beberapa asumsi tertentu sehingga sistem tersebut bisa dipelajari secara ilmiah (Law and Kelton, 1991).

Dalam simulasi digunakan komputer untuk mempelajari sistem secara numerik, dimana dilakukan pengumpulan data untuk melakukan estimasi statistik untuk mendapatkan karakteristik asli dari sistem.

Simulasi merupakan alat yang tepat untuk digunakan terutama jika diharuskan untuk melakukan eksperimen dalam rangka mencari *command* terbaik dari komponen-komponen sistem. Hal ini dikarenakan sangat mahal dan memerlukan waktu yang lama jika eksperimen dicoba secara nyata. Dengan melakukan studi simulasi maka dalam waktu singkat dapat ditentukan keputusan yang tepat serta dengan biaya yang tidak terlalu besar karena semuanya cukup dilakukan dengan komputer.

Pendekatan simulasi diawali dengan pembangunan model sistem nyata. Model tersebut harus dapat menunjukkan bagaimana berbagai komponen dalam sistem saling berinteraksi sehingga benar-benar menggambarkan apa yang dilakukan oleh sistem. Setelah model dibuat maka model tersebut ditransformasikan ke dalam program komputer sehingga memungkinkan untuk disimulasikan.

2.9 Sekilas tentang Teknologi CSMA

Carrier Sense Multiple Access atau disingkat CSMA adalah mekanisme mendengarkan aliran data oleh sebuah *Network Interface Card* (disingkat : NIC, atau lebih dikenal di tanah air dengan *LAN-Card* - Kartu Jaringan) sebelum NIC tersebut mengirimkan data (atau *frame*) pada *shared broadcast media Ethernet* dalam sebuah jaringan komputer.

Karena berhubungan dengan *frame* (atau proses *framing*, yaitu perubahan paket data kedalam bentuk aliran listrik) - maka sederhananya mekanisme CSMA ini bekerja pada lapisan ke 2 (*Data Link*) pada OSI *Layer*, atau lapisan *Network Access* pada TCP/IP.

Ada 2 kemungkinan yang akan terjadi pada mekanisme CSMA :

- bila tidak ada aliran data masuk ke NIC, maka selanjutnya NIC tersebut akan bisa mengirim data
- bila ada aliran data masuk ke NIC, maka dia harus menunggu sampai saat aliran data tidak ada agar dapat mengirim data

Dikarenakan tetap terjadi kemacetan arus lalu-lintas pada *shared media* dengan metode CSMA ini, maka selanjutnya dikembangkan kembali 2 *varian* dari CSMA, yaitu :

- CSMA *Collision Avoidance* (CSMA/CA)
- CSMA *Collision Detection* (CSMA/CD)

2.10 Sekilas tentang C++

C++ adalah bahasa pemrograman komputer C++ dikembangkan di *Bell Labs* (Bjarne Stroustrup) pada awal tahun 1970-an, Bahasa itu diturunkan dari bahasa sebelumnya, yaitu BCL, Pada awalnya, bahasa tersebut dirancang sebagai bahasa pemrograman yang dijalankan pada sistem *Unix*, Pada perkembangannya, versi ANSI (*American National Standard Institute*) Bahasa pemrograman C menjadi versi dominan, Meskipun versi tersebut sekarang jarang dipakai dalam pengembangan sistem dan jaringan maupun untuk sistem embedded, Bjarne Stroustrup pada *Bell labs* pertama kali mengembangkan C++ pada awal 1980-an, Untuk mendukung fitur-fitur pada C++, dibangun efisiensi dan sistem *support* untuk pemrograman tingkat rendah (*low level coding*). Pada C++ ditambahkan konsep-konsep baru seperti *class* dengan sifat-sifatnya seperti *inheritance* dan *overloading*. Salah satu perbedaan yang paling mendasar dengan bahasa C adalah dukungan terhadap konsep pemrograman berorientasi objek (*Object Oriented Programming*).

C++ bukanlah versi berikut dari bahasa C, tetapi merupakan bahasa terpisah yang telah dikembangkan dengan menggunakan C sebagai dasar sintaks. C++ dapat digunakan untuk menulis kode prosedural seperti C, tetapi nilainya terletak pada kemampuan untuk menulis program berorientasi objek. (Parsons, 1997, p. 13)

Sedangkan menurut Oualline (2003, p.3), C++ adalah bahasa pemrograman tingkat tinggi yang memungkinkan seorang *software engineer* untuk berkomunikasi secara efisien dengan komputer. C++ adalah bahasa yang sangat fleksibel dan mudah beradaptasi. Sejak diciptakannya pada tahun 1980, C++ telah digunakan untuk berbagai

program termasuk *firmware* untuk mikrokontroler, sistem operasi, aplikasi, dan pemrograman grafik.

2.11 Pengenalan NS-3

Akibat dari penggunaan suatu peralatan jaringan yang semakin luas dan kompleks, kebutuhan akan teknologi simulasi jaringan yang akurat dan *scalable* menjadi sangat mendesak. Disamping pentingnya akan kebutuhan jaringan *testbed* skala besar untuk penelitian, simulasi tetap memegang peranan penting dalam aspek *scalability*(dalam ukuran dan kecepatan eksperimen), *reproducibility*, *rapid prototyping*, dan *education*. Dengan melakukan simulasi sebelumnya, pendekatan dapat dipelajari secara detail dengan berbagai skala, berbagai data aplikasi, berbagai kondisi, dan akan menghasilkan sesuatu yang *reproducible* dan *analyzable*.

Dalam beberapa tahun, keunggulan alat simulasi jaringan ns-2 merupakan standar fakta untuk penelitian akademis dalam protokol jaringan dan metode komunikasi. Banyaknya paper yang telah ditulis untuk melaporkan hasil yang didapat menggunakan ns-2 , dan ratusan model baru telah ditulis dan berkontribusi kepada kode dasar ns-2. Disamping kepopulerannya, dan disamping banyaknya alternatif simulasi jaringan yang didokumentasikan, para peneliti telah membuat sebuah proyek pada tahun 2005 untuk mendesain sebuah simulasi jaringan yang nantinya akan menggantikan ns-2 dalam penelitian jaringan. Keputusan untuk mengembangkan alat baru dimotivasi oleh sebuah fakta tentang bagaimana pemodelan jaringan dengan suatu cara yang terbaik untuk penelitian jaringan. Alat ini didesain untuk menggantikan ns-2, dan nama yang dipakai adalah ns-3.

NS-3 adalah sebuah simulator jaringan untuk sistem internet yang biasa digunakan untuk penelitian dan pendidikan. NS-3 adalah perangkat lunak yang bersifat *open source*, dilisensikan di bawah lisensi GNU GPLv2, dan secara terbuka tersedia untuk penelitian, pengembangan, dan penggunaan umumnya.

NS-3 ini dimaksudkan sebagai pengganti simulator NS-2. Nama domain akronim "nsnam" berasal dari historis rangkaian ns (*network simulator*) dan Nam (*network animator*). NS-3 ditulis dalam bahasa pemrograman C++ dan Python dan tersedia pula *source code* untuk Linux dan varian Unix, OS X, dan Windows via Cygwin.

Salah satu tujuan dasar dalam mendesain ns-3 adalah untuk meningkatkan *realism* model; sebagai contoh, untuk membuat model implementasi agar lebih mendekati ke implementasi *software* yang sesungguhnya. Alat simulasi yang berbeda menggunakan pendekatan yang berbeda dalam pemodelan, termasuk penggunaan bahasa dan *code generation tool*, serta penggunaan komponen yang berbasiskan *programming paradigms*. Ketika bahasa *high-level modeling* dan simulasi khusus *programming paradigms* memiliki kelebihan, implementasi pemodelan sesungguhnya bukan merupakan satu kekuatannya. Pemisahan *High-level modeling* dapat menyebabkan hasil simulasi terlalu banyak yang berbeda dari hasil percobaan, dan oleh karena itu, perhatian lebih harus diberikan. Sebagai contoh, ns-3 menggunakan C++ sebagai bahasa pemrograman karena ia memiliki fasilitas yang lebih baik dari bahasa lain yang termasuk kedalam bahasa C lainnya. Arsitektur ns-3 juga sama dengan komputer yang menggunakan Linux, dengan *interface internal*(*network* ke *device driver*) dan *interface aplikasi* (*sockets*) yang menggambarkan dengan baik bagaimana sebuah komputer dibangun. Sebagai gambaran nanti, ns-3 juga menegaskan kemampuan lebihnya yang

membolehkan ia untuk digunakan dalam *testbed*, *real device* dan aplikasi, serta tujuannya yang mengurangi kemungkinan terputusnya jaringan saat mengerjakan simulasi ke percobaan.

Manfaat lainnya adalah *reuse*. ns-3 tidak murni sebagai simulator baru tetapi sebuah perpaduan dari beberapa pendahulunya, termasuk ns-2 itu sendiri (*random number generators, selected wireless and error models, routing protocols, the Georgia Tech Network Simulator (GTNetS)[393], and the YANS simulator[271]*). Software yang mengotomatisasi pembangunan jaringan tabel *routing* untuk topologi statis adalah *porting from quagga routing suite*. ns-3 juga memprioritaskan penggunaan *input* standar dan format file *output* sehingga alat-alat eksternal (seperti *packet trace analyzer*) dapat digunakan. Pengguna juga dapat menggunakan *link external* perpustakaan seperti Perpustakaan Ilmiah GNU atau IT++.

Penegasan ketiga terletak pada kemudahan *debugging* dan penyelarasan yang lebih baik dengan bahasa ini. Secara arsitektur, hal ini menyebabkan ns-3 jauh berbeda dari *Tcl object-oriented* dan C++ ns-2, yang sulit untuk didebug dan masih asing untuk sebagian besar siswa. Sebaliknya, desain yang dipilih adalah murni menekankan model C++ berbasis kinerja dan kemudahan *debugging*, dan untuk menyediakan *scripting* Python berbasis API yang memungkinkan ns-3 untuk diintegrasikan dengan lingkungan berbasis Python atau model pemrograman. Pengguna ns-3 bebas untuk menulis simulasi mereka baik dengan C++ *main ()* program atau Python program. API ns-3 tingkat rendah berorientasi pada kemampuan *user* tetapi lebih mudah diakses dengan "helper" API yang *overlay* di atas dari tingkat rendah API.

Kesimpulannya, ns-3 bukan merupakan alat komersial, dan ada sumber daya yang terbatas untuk melakukan perawatan jangka panjang dari sebuah kode dasar yang terus bertambah. Oleh karena itu, perawatan perangkat lunak adalah masalah utama. Dua masalah pada ns-2 yang menuntun tim ns-3, setelah pertimbangan hati-hati, untuk meninggalkan tujuan dari kesesuaian balik dengan atau tambahan ns-2. Pertama, ns-2 itu tidak menaikkan standar *coding*, dan menerima model dengan pengujian software yang tidak konsisten dan verifikasi model, serta kurangnya pertimbangan desain sistem secara keseluruhan. Kebijakan ini memungkinkan alat ini untuk tumbuh jauh dari waktu ke waktu tapi akhirnya akan menyebabkan pengguna kehilangan kepercayaan pada hasilnya, membuat perangkat lunak kurang fleksibel untuk konfigurasi ulang. NS-3 mengekspor *bindings* untuk *Python* dan berpotensi sebagai bahasa *scripting* lain di masa depan. Sebuah standar pengkodean yang lebih ketat, memeriksa kode proses, dan infrastruktur tes telah dimasukkan ke tempatnya. Maka mungkin saja untuk membangun ns-3 dengan kompatibilitas penuh pada tingkat *scripting*, tetapi proyek ns-3 tidak memiliki sumber daya untuk mempertahankan seperti sebuah *backward-compatibility layer*. Oleh karena itu, keputusan dibuat untuk menciptakan simulator baru oleh *port ns-2* yang bisa digunakan tanpa mencurigai *maintainability* jangka panjang ns-3.

Dengan dukungan yang signifikan dari *US National Science Foundation*, INRIA dan pemerintah Prancis, *Georgia Institute of Technology*, *University of Washington*, dan *Google Summer of Code program*, ns-3 juga telah dioperasikan sebagai *free software*, proyek perangkat lunak *open source* dari awal.

2.12 Parameter yang Diukur

2.12.1 Packet Delivery Ratio (PDR)

Paket delivery ratio adalah rasio jumlah paket data yang dikirimkan ke tujuan node dibagi dengan jumlah paket data yang dikirimkan oleh sumber node. Atau dapat dikatakan pula bahwa PDR adalah perbandingan paket yang berhasil diterima dan dikirim dalam jaringan. PDR dapat dihitung sebagai berikut :

$$\eta_{PDR} = -10 \log \left(\frac{1}{r} \sum_{i=1}^r \frac{1}{y_i^2} \right),$$

Dimana r adalah jumlah pengulangan simulasi di bawah angka yang eksperimental yang sama dan y adalah kinerja respon.

2.12.2 Throughput

Throughput adalah ukuran kecepatan data dalam melewati sebuah entitas (contohnya sebuah port atau jaringan). (Forouzan, 2003)

Throughput adalah bandwidth aktual yang terukur pada suatu ukuran waktu tertentu dalam suatu hari menggunakan rute internet yang spesifik ketika sedang mendownload suatu file. Throughput lebih pada menggambarkan bandwidth yang sebenarnya (aktual) pada suatu waktu tertentu dan pada kondisi dan jaringan internet tertentu yang digunakan untuk mendownload suatu file dengan ukuran tertentu.

2.12.3 Delay

Delay adalah jumlah waktu yang dibutuhkan oleh suatu jaringan untuk mengirim paket dari sumber ke tujuannya. (Norton, 1999)

Delay yang biasa juga disebut dengan latency. Delay mengacu pada waktu yang dibutuhkan untuk paket yang akan dikirim melintasi jaringan dari sumber ke tujuan. Dalam jaringan komputer , delay propagasi adalah jumlah waktu yang diperlukan untuk kepala sinyal untuk perjalanan dari pengirim ke penerima melalui sebuah perantara. Hal ini dapat dihitung sebagai rasio antara panjang link dan kecepatan propagasi melalui media tertentu.