

## BAB 2

### Studi Pustaka

#### 2.1 Jaringan Komputer

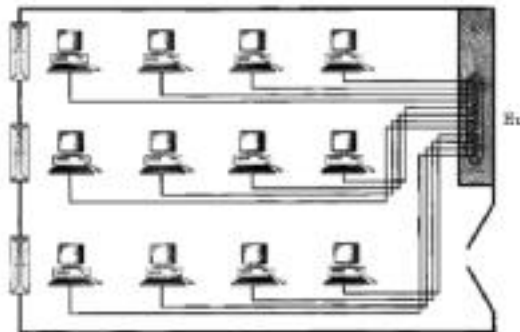
(Tanenbaum, 2007:8) Jaringan Komputer adalah kumpulan komputer otonom yang saling berhubungan dengan satu teknologi. Dua komputer dapat dikatakan berhubungan jika mereka dapat saling bertukar informasi. Media yang menghubungkan komputer tidak hanya melalui kabel tembaga saja, melainkan dapat juga melalui *fiber optic*, pancaran gelombang radio, *infrared*, dan satelit. Kecepatan transfer dari suatu jaringan sering disebut sebagai *bandwidth*, satuan yang dipakai dalam mengukur *bandwidth* ini dapat berupa *bit* per-detik ataupun *byte* per-detik. Satu *byte* terdiri dari 8bit data. Sedangkan 1 *kilobyte* data terdiri dari 1024 *byte* data.

#### 2.2 Bentuk Jaringan

(Forouzan, 2007:13) Jaringan komputer dapat dibagi menjadi 3 (tiga) bagian. Bagian - bagian tersebut adalah sebagai berikut :

##### a. LAN (*Local Area Network*)

Forouzan (2008:14) Local Area Network biasanya digunakan apabila jaringan berskala kecil seperti kampus, kantor dan gedung dan menghubungkan *device* yang ada. Jangkauan yang dapat dicapai LAN masih berjarak beberapa kilometer. LAN berhubungan dengan *area network* yang berukuran relatif kecil. Dapat dikembangkan dengan mudah dan mendukung kecepatan transfer data yang cukup tinggi yaitu 100-1000Mbps.

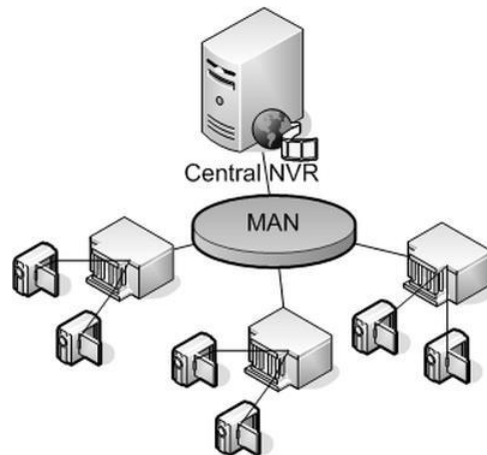


Gambar 2.1 Topologi LAN

Sumber: Forouzan (2008:14)

### b. MAN (*Metropolitan Area Network*)

(Forouzan, 2007: 15) Metropolitan Area Network memiliki jangkauan yang lebih besar daripada LAN, biasanya mencakup satu kota. MAN didesain untuk para pengguna yang menginginkan kecepatan tinggi dan memiliki titik akhir (*endpoint*), di dalam kota. Contoh dari MAN ini adalah *bank* dengan beberapa cabang dalam satu kota, dan *TV cable* dalam satu kota.

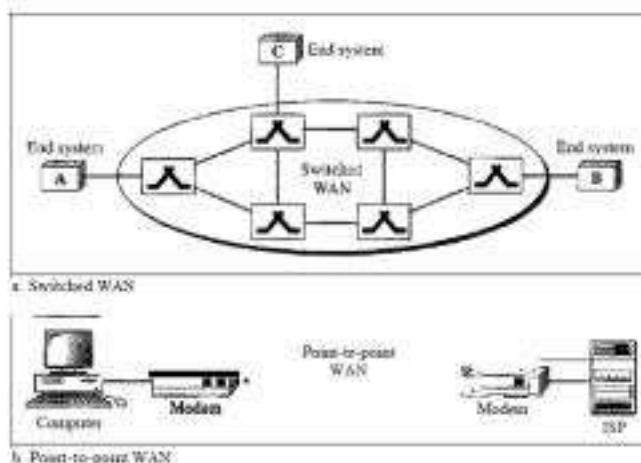


Gambar 2.2 Topologi MAN

Sumber: <http://goo.gl/DXhqKE>

### c. WAN (*Wide Area Network*)

(Forouzan, 2007: 15) *Wide Area Network* dipakai secara umum sebagai alat untuk mengatasi jarak geografis yang luas, menyediakan transmisi informasi data, gambar, suara dan *video*, yang mencakup jangkauan luas seperti negara, benua, maupun penjuru dunia. *Point to point* WAN biasanya disewakan oleh ISP melalui telepon atau *cable tv* untuk menghubungkan komputer rumah atau jaringan LAN kecil ke ISP.



Gambar 2.3 Topologi WAN

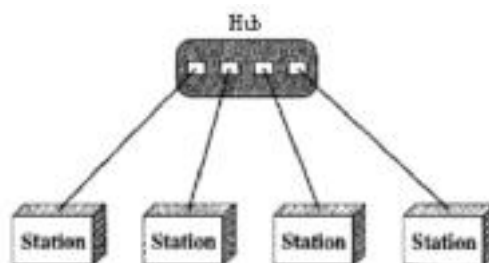
Sumber: Fourzan (2007:15)

### 2.3 Topologi Jaringan

(Stallings, 2007:451) Topologi jaringan adalah peta atau struktur jaringan yang terdiri kumpulan *switch* dan *hub*, yang mampu menghasilkan komunikasi interkoneksi di antara tiap node dari jaringan tersebut. Topologi jaringan dapat dibedakan menjadi dua yaitu topologi fisikal dan topologi logical, diantaranya topologi star, topologi bus, topologi mesh, dan topologi ring. Secara konseptual topologi tersebut dapat dijelaskan sebagai berikut :

#### a. Topologi Star

(Sofana, 2012:114) Dalam topologi *star*, semua kabel dihubungkan dari komputer-komputer ke lokasi pusat (*centrallocation*), titik pusat ini biasanya berupa *hub* atau *switch* sehingga seolah-olah komputer yang terhubung berbentuk seperti bintang. Pada sistem jaringan di DPD menggunakan topologi ini.

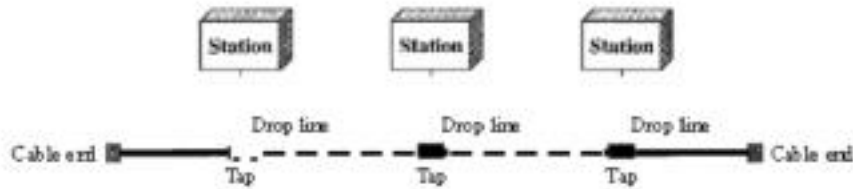


Gambar 2.4 Topologi Star

Sumber: Fourzan (2007:11)

### b. Topologi Bus

Topologi *Bus* bersifat *multipoint*, dengan satu kabel sebagai *backbone* yang menghubungkan semua *device* yang ada dalam jaringan.

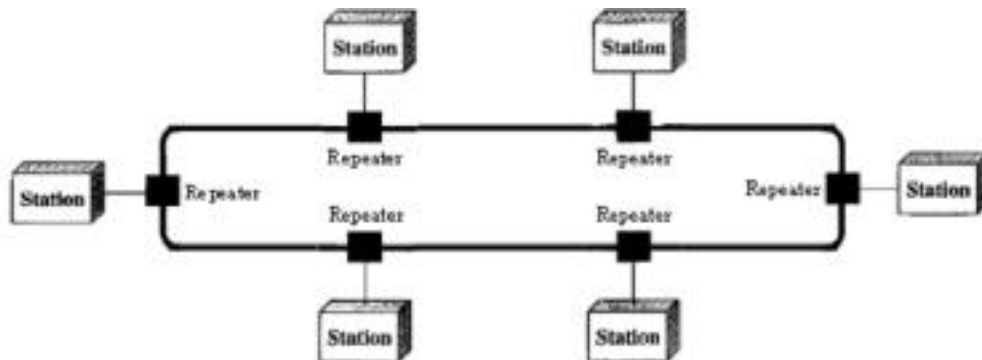


Gambar 2.5 Topologi Bus

Sumber: Fourzan (2007:11)

### c. Topologi Ring

Topologi *ring* menghubungkan *host* dengan *host* lainnya hingga membentuk ring (lingkaran tertutup). Setiap *device* memiliki koneksi *dedicated point-to-point* dengan 2 *device* di setiap sisinya (satu di sisi kiri dan satu di sisi kanan).

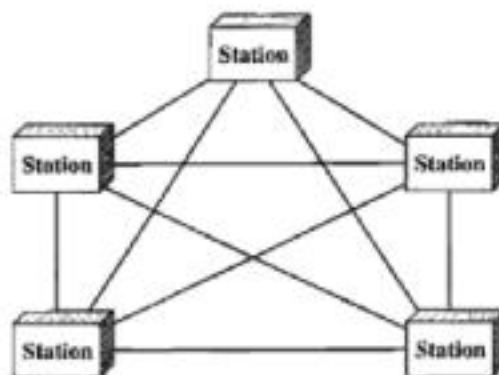


Gambar 2.6 Topologi Ring

Sumber: Fourzan (2007:12)

### d. Topologi Mesh

Dalam topologi *Mesh* setiap *device* memiliki *dedicated point-to-point link* untuk setiap *device* yang ada. Setiap komputer terhubung ke komputer lain melalui kabel, bisa menggunakan kabel *coaxial*, *twisted pair*, bahkan serat *optic*. Topologi *mesh* cocok digunakan pada jaringan yang sangat kritis. Apabila salah satu atau beberapa kabel putus masih tersedia rute alternative melalui kabel yang lain.



Gambar 2.7 Topologi Mesh

Sumber: Fourzan (2007:10)

Dalam penelitian ini penulis menggunakan topologi mesh. Topologi Mesh digunakan karena jaringan yang digunakan di perusahaan PT. CYBER NETWORK INDONESIA saling terhubung.

## 2.4 Perangkat Jaringan

(Sofana, 2012:58) Perangkat keras yang dibutuhkan untuk membangun sebuah jaringan komputer yaitu : *switch*, *router*, *hub*, *bridge* dan lainnya yang dibutuhkan untuk proses transformasi data di dalam jaringan. Beberapa perangkat jaringan komputer di antaranya adalah sebagai berikut :

### a. Switch

*Switch* dapat mempelajari alamat hardware *host* tujuan, sehingga informasi bisa langsung dikirim ke *host* tujuan. *Switch* yang lebih cerdas dapat mengecek *frame* yang *error* dan dapat mem-blok *frame* yang *error* tersebut. *Switch* bekerja pada Layer 2 dan Layer 3. *Switch* bekerja berdasarkan alamat *MAC* pada NIC ( *Network Interface Card* ). Hal ini bertujuan untuk mengetahui kemana paket data itu akan dikirim dan diterima.

### b. Router

*Router* sering digunakan untuk menghubungkan beberapa *network* baik *network* yang sama maupun berbeda dari segi teknologinya. *Router* berfungsi sebagai alat untuk mentransfer paket data dari satu port ke port yang lain. *Router* digunakan dalam jaringan berbasis teknologi protokol *TCP/IP*. Sebuah *router*

memiliki kemampuan *routing*. Artinya *router* secara cerdas dapat mengetahui kemana rute perjalanan informasi yang disebut *packet* kan dilewatkan.

### c. Hub

*Hub* adalah bentuk sederhana dari *Switch*. *Hub* digunakan untuk jaringan sederhana. Cara kerja *Hub* adalah menyalin paket data dari sumber yang terkoneksi pada suatu port dan mentransferkannya ke seluruh port yang tersambung pada *Hub*. Jika jumlah port yang tersedia tidak cukup untuk menghubungkan semua komputer yang akan dihubungkan ke dalam satu jaringan dapat digunakan beberapa *hub* yang dihubungkan secara *up-link*.

### d. Bridge

*Bridge* bekerja pada *physical layer* dan *data link layer*. *Bridge* bekerja pada *physical layer* dengan mengirim kembali sinyal yang diterima, sementara *bridge* bekerja pada *data link layer* dengan memeriksa *MAC address* yang ada di dalam *frame*. *Bridge* memiliki kemampuan *filtering*. Contoh jaringan yang menggunakan *fiber optic* dengan jaringan yang menggunakan *coaxial*.

## 2.5 Model OSI (*Open System Interconnection*)

(Sofana, 2012:91) Model OSI merupakan suatu referensi yang digunakan dalam protokol komunikasi jaringan, menjadi semacam referensi atau acuan bagi siapa saja yang ingin memahami cara kerja jaringan komputer. *OSI Layer* atau dikenal sebagai *Open System Interconnection – seven layer model* sering dipergunakan untuk menjelaskan cara kerja jaringan komputer secara logika. Secara umum *OSI Layer* membagi berbagai fungsi *network* menjadi 7 lapisan sebagai berikut :

### a. Physical Layer

(Sofana, 2012:98) Layer ini menentukan masalah kelistrikan atau gelombang atau medan dan berbagai prosedur atau fungsi yang berkaitan dengan *link* fisik, seperti besar tegangan/ arus listrik, panjang maksimal media transmisi, penggantian jasa, jenis kabel dan konektor. Contoh : RS-232, V.35, V.34, I.430, *hub*, *repeater*, *fiber optics*, 802.11a/b/g/n, RJ45, *Ethernet*, NRZI, NRZ, B8ZS.

### **b. Data Link Layer**

(Sofana, 2012: 97-98) Menentukan pengalamatan fisik, pendeteksi error, *frame flow control*, dan *network topology*. Ada dua *sublayer* pada data link yaitu : *Logical Link Control* bertugas untuk mengatur komunikasi seperti *error notification* dan *flow control*, *Media Access Control* mengatur pengalamatan fisik yang digunakan dalam proses komunikasi antar adapter. Contoh : *Token ring*, *Frame relay*, SDLC, HDLC , ISL, PPP, IEEE 802.2 / 802.3 (*ethernet*), FDDI, dan ATM.

### **c. Network Layer**

(Sofana, 2012: 96) *Network Layer* dari model OSI ini menentukan rute yang dilalui oleh data. *Layer* ini menyediakan *logical addressing* (pengalamatan logika) dan *path determination* (penentuan rute tujuan). Untuk melengkapi pekerjaan ini, *Network layer* mendefinisikan *logical address* sehingga setiap titik ujung bisa diidentifikasi. *Layer* ini juga mendefinisikan bagaimana *routing* bekerja dan bagaimana rute dipelajari sehingga semua paket bisa dikirim. Contoh : IPX, IP , ICMP, IPsec, ARP, RiP, IGRP, OSPF, dan NBF.

### **d. Transport Layer**

(Sofana, 2012: 96) *Transport Layer* bertanggung jawab membagi data menjadi segmen, mengatur komunikasi *end-to-end*. *Layer* ini bertanggung jawab terhadap keselamatan data dan segmentasi data, seperti mengatur *flow control* (kendali aliran data), *error detection* (deteksi kesalahan), and *correction* (koreksi), *data sequencing* (urutan data), dan *size of the packet* (ukuran paket). Contoh : TCP, SPX, UDP, SCTP, dan IPX.

### **e. Session Layer**

(Sofana, 2012: 95) *Layer* ini mengatur sesi (*session*) yang meliputi *establishing* (memulai sesi), *maintaining* (mempertahankan sesi), dan *terminating* (mengakhiri sesi) antar entitas yang dimiliki oleh *presentation layer*. *Session layer* juga berfungsi untuk memberikan service kepada *presentation layer*. Contoh : SQL, X WINDOW, DNS, NetBIOS, ASP, SCP, OS *scheduling*, RPC

#### **f. Presentation Layer**

(Sofana, 2012: 95) *Layer* ini berfungsi untuk mentranslasikan data yang hendak ditransmisikan oleh aplikasi ke dalam format yang dapat ditransmisikan melalui jaringan. *Layer* ini juga melakukan enkripsi, dekripsi, translasi dan kompresi data untuk mengamankan data. Contoh : TDI, ASCII, EBCDIC, MIDI, MPEG, dan ASCII7.

#### **g. Application Layer**

(Sofana, 2012:94) *Layer* ini berfungsi sebagai perantara antara *software* dan *resources network* yang dipakai untuk mengatur bagaimana *software* dapat berinteraksi dan mengakses jaringan dan menyediakan servis bagi berbagai aplikasi network. Pada *layer* ini terbagi menjadi 2 aplikasi yaitu *client-server* dan *non clientserver*. Contoh: NNTP, HTTP, HTTPS, SMTP, SNMP, Telnet, dan FTP.

### **2.6 TCP/IP**

(Beasley, 2009:156) *Transmission Control Protocol / InternetProtocol* ( TCP / IP ) adalah protokol yang digunakan untuk komunikasi antar *host* dalam jaringan lokal dan di Internet. TCP / IP dapat digunakan untuk mengaktifkan jaringan komunikasi di LAN , jaringan kampus, dan jaringan luas selama host mendukung protokol. TCP / IP secara luas didukung dan termasuk dalam sistem operasi seperti Windows 9x, NT, 2K, XP, Vista, Mac OS, Linux, dan Unix. TCP/IP memiliki 4 buah *layer*, yaitu: *Network interface layer*, *internet layer*, *transport layer* dan *application layer*.

#### **A. Application Layer**

(Beasley, 2009:157) Lapisan ini digunakan untuk memproses permintaan dari *host* dan untuk memastikan sambungan dibuat ke *port* yang sesuai. *Port* pada dasarnya adalah sebuah alamat yang digunakan untuk mengarahkan data ke aplikasi tujuan yang tepat.

#### **B. Transport Layer**

(Beasley, 2009:158) Lapisan ini digunakan dalam membangun jaringan koneksi, mengelola pengiriman data antara penerima dan pengirim, dan mengakhiri sambungan data. Ada dua protocol transport dalam TCP/IP *transport layer* :



### 1. Transmission Control Protocol (TCP)

TCP merupakan *connection-oriented* protokol, yang berarti menetapkan koneksi jaringan, mengelola transfer data, dan berakhir sambungan. Menetapkan protokol TCP seperangkat aturan atau pedoman untuk membangun koneksi.

### 2. User Datagram Protocol (UDP)

UDP merupakan protokol *connectionless*, berarti ini paket UDP diangkut melalui jaringan tanpa sambungan yang didirikan dan tanpa penghargaan bahwa paket data tiba di tujuan. UDP berguna dalam aplikasi seperti konferensi video dan *feed* audio, di mana seperti pengakuan tidak diperlukan.

## C. Internet Layer

(Beasley, 2009:162) Lapisan ini protokol yang digunakan untuk menangani dan routing paket data. Protokol yang merupakan bagian dari TCP / IP lapisan *Internet* termasuk IP, ARP, ICMP, IGMP. Protokol yang merupakan bagian *Borrowed* lapisan internet, yaitu :

### 1. IP (Internet Protocol)

(Beasley, 2009:162) IP merupakan pengalamatan yang digunakan untuk mengidentifikasi sumber dan alamat tujuan paket data yang dikirimkan melalui jaringan IP. Alamat IP adalah alamat logis yang terdiri dari jaringan dan bagian alamat *host*, bagian jaringan yang digunakan untuk mengarahkan data ke jaringan yang tepat. Alamat *host* mengidentifikasi alamat ditugaskan secara lokal ke *host*.

### 2. ARP (Address Resolution Protocol)

(Beasley, 2009:162) ARP digunakan untuk menyelesaikan alamat IP ke alamat *hardware* untuk pengiriman akhir paket data ke tujuan. Masalah ARP *a query* dalam jaringan disebut sebuah permintaan ARP, menanyakan yang antarmuka jaringan memiliki IP ini alamat, protokol yang berisi alamat perangkat keras untuk host tujuan.

### 3. ICMP (Internet Control Message Protocol)

(Beasley, 2009:163) ICMP digunakan untuk mengontrol aliran data dalam jaringan, kesalahan pelaporan, dan untuk melakukan diagnosa. Perangkat jaringan seperti *router* mengirimkan paket ICMP sumber - *quench* ke host yang meminta perlambatan dalam transfer data.

#### 4. IGMP (Internet Group Message Protocol)

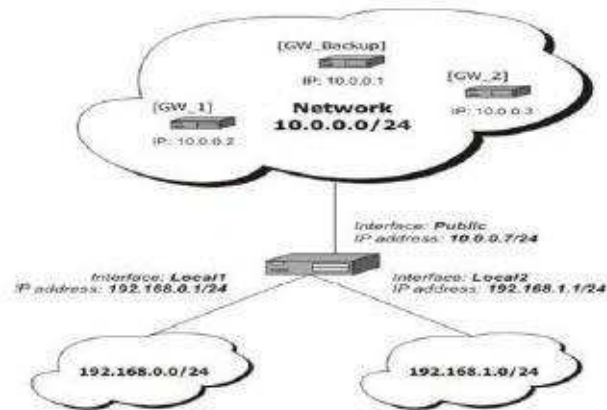
(Beasley, 2009:164) IGMP digunakan ketika satu *host* perlu untuk mengirim data ke banyak tujuan host disebut *multicasting* adalah alamat digunakan untuk mengirim paket data, alamat *multicast* tidak ditugaskan untuk *host* dalam jaringan. Contoh dari aplikasi yang menggunakan paket IGMP adalah ketika *router* menggunakan multicasting untuk berbagi tabel *routing*.

### D. Network Interface Layer

(Beasley, 2009:164) Lapisan ini bagaimana *host* terhubung ke jaringan, bahwa penerima dapat menjadi komputer atau perangkat jaringan seperti *router*. Jenis jaringan yang menghubungkan *host* tidak ditentukan oleh TCP/IP protokol. Penerima bisa terhubung ke jaringan *internet/ethernet* atau token-ring atau *router* yang terhubung ke frame menyampaikan *wide area network*.

## 2.7 Failover

(Megis and Riyadi 2010:30) Definisi *failover* dalam istilah *computer internet working* adalah kemampuan sebuah sistem untuk dapat berpindah secara manual maupun otomatis jika salah satu sistem mengalami kegagalan sehingga menjadi backup untuk sistem yang mengalami kegagalan.



Gambar 2.8 Konsep Failover

Sumber: <https://www.mikrotik.com/testdocs/ros/2.9/img/route-policy.jpg>

Untuk mempermudah dan memperjelas maksud *failover* dapat melihat contoh gambar 2.8. Pada gambar tersebut dapat dilihat sebuah local area network menggunakan lebih dari satu jalur jaringan ISP. Jaringan lokal dengan IP 192.168.0.1/24 menggunakan *gateway* 1, sedangkan IP 192.168.1.1/24 menggunakan *gateway* 2. Jika *gateway* 1 mengalami disconnect (putus) maka *gateway backup* akan menggantikan *gateway* 1. Jika *gateway* 1 sudah kembali normal maka jalur koneksi yang digunakan kembali menjadi *gateway* 1. an begitu juga dengan *gateway* 2 apabila mengalami *disconnect* (putus). Dengan begitu dapat disimpulkan bahwa tujuan dari *failover* adalah digunakan untuk menggantikan atau sistem backup koneksi ISP yang terputus dengan koneksi ISP yang lainnya.

## 2.8 Network Monitoring System

(Putri, 2016) *Network Monitoring System* (NMS) merupakan *tool* untuk melakukan *monitoring* atau pengawasan pada elemen-elemen dalam jaringan komputer. Fungsi dari NMS adalah melakukan pemantauan terhadap kualitas SLA (Service Level Agreement) dari *Bandwidth* yang digunakan. Perangkat lunak NMS digunakan sebagai sistem yang mengelola proses pemantauan terhadap fungsi dan kinerja jaringan yang meliputi kepadatan dan lalu lintas dalam ukuran penggunaan *bandwidth*. Proses *monitoring* ini dapat dikembangkan sampai ke penggunaan sumber daya, seperti sistem *up/down*, utilisasi CPU dan *memory*, serta manajemen port. Hasil pemantauan

tersebut dijadikan bahan dalam pengambilan keputusan oleh pihak manajemen, dan dapat juga digunakan oleh administrator jaringan (*technical person*) untuk menganalisa terjadinya kejanggalan dalam operasional jaringan.

Kegunaan dari NMS tersebut yaitu :

- a. Memberikan informasi tentang operasional dan konektifitas dari peralatan dan sumber daya yang ada dalam jaringan, serta informasi status jaringan secara remote.
- b. Perencanaan peningkatan (*upgrade*) dan perubahan peralatan jaringan.
- c. Mendiagnosa masalah-masalah dalam jaringan.
- d. Bahan untuk keperluan SLA (*Service Level Agreement*).
- e. Memastikan uptime untuk keperluan pengguna yang tergantung dengan ketersediaan jaringan Komputer serta keamanan sistem beroperasi dengan baik.

## 2.9 Gammu

(A, Ramadhika, 2012) Gammu adalah sebuah aplikasi cross-platform yang digunakan untuk menjembatani / mengomunikasikan antara *database SMS Gateway* dengan *sms devices*. Aplikasi Gammu berupa daemon yang berjalan secara *background*. Setiap saat, gammu memonitor *sms devices* dan *database sms gateway*. Saat ada sms masuk ke *sms devices*, maka gammu langsung memindahkannya ke dalam *inbox* dalam *database sms gateway*. Sebaliknya saat aplikasi pengirim SMS memasukkan sms ke dalam *outbox* dalam *database sms gateway*, maka gammu mengirimkannya melalui *sms devices*, dan memindahkan sms ke *sent item* dalam *database*.

## 2.10 My SQL

(Wardani, 2012) MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau yang dikenal dengan DBMS (*Database Management System*), database ini *multithread*, *multi user*. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU *General Public Licence* (GPL), tetapi mereka juga menjual di bawah lisensi komersial untuk kasus-kasus yang bersifat khusus.

Kekuatan MySQL tidak ditopang oleh sebuah komunitas, seperti Apache, yang dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh

pemilik masing-masing, tetapi MySQL didukung penuh oleh sebuah perusahaan profesional dan komersial, yakni MySQL AB dari Swedia.

MySQL adalah *Relational Database Mangement System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (General Public License). Di mana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *closed Source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau mseleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

### 2.11 Jaringan Client-Server

(Komputer, 2010:5) *Client-server* adalah salah satu model komunikasi 2 komputer atau lebih yang berfungsi melakukan pembagian tugas. Client bertugas untuk melakukan *input*, *update*, penghapusan, dan menampilkan data sebuah *database*. Sementara *server* bertugas menyediakan pelayanan untuk melakukan manajemen, yaitu menyimpan dan mengolah *database*.

Aplikasi *client-server* merupakan jawaban atas berkembangnya teknologi informasi, di mana sebuah perusahaan memiliki banyak departemen dan harus terhubung satu sama lain dalam melakukan akses data.

### 2.12 Tasksel

(Ubuntu, 2012) Tasksel adalah *tool* Debian / Ubuntu yang menginstal beberapa paket terkait sebagai "*task*" terkoordinasi ke sistem. Fungsi ini mirip dengan *meta-paket*, dan sebenarnya, sebagian besar tugas yang tersedia dari tasksel juga tersedia sebagai *meta-packages* dari Ubuntu package managers (seperti *Synaptic Package Manager* atau *KPackageKit*).

### 2.13 Kalkun

(Kalkun, 2011) Kalkun adalah manajemen SMS (*Short Message Service*) berbasis *open-source*, menggunakan gammu-smsd sebagai mesin SMS *gateway* untuk mengirim

dan menerima pesan dari telepon / modem anda. Kalkun dan gammu adalah dua hal yang berbeda. Anda harus meng-*install* dan mengkonfigurasi gammu ter-sendiri. Kalkun hanya mengelola database yang diberikan oleh gammu. Arsitekturnya sebagai berikut;

Kalkun ↔ Mesin DB ↔ Gammu-SMSD ↔ Telepon / Modem

Karena berbasis web, anda hanya perlu meletakkan perangkat lunak ini di satu komputer, misalkan saja sebagai *server* anda. Dan komputer lain di jaringan anda (atau bahkan di *internet*, jika memang dibutuhkan), komputer tersebut hanya butuh browser untuk menggunakannya. Kelebihan lainnya adalah anda tidak perlu khawatir dengan kapasitas memori ponsel, karena pesan anda tersimpan di database.

Berikut adalah fitur-fitur pada Kalkun:

- a. *Multiple database backend* (Saat ini support MySQL, PostgreSQL dan SQLite3)
- b. Multiuser (setiap pengguna memiliki *Inbox*, *Outbox*, folder, dan *phonebook* miliknya sendiri)
- c. Percakapan SMS dikelompokkan berdasarkan nomor telepon
- d. Mengelola SMS (*Inbox*, *Outbox*, *Sent Items*)
- e. Mengirim SMS, secara pribadi atau kelompok
- f. SMS terjadwal, dengan *datetime* atau *delay*
- g. Kelola Buku Telepon, secara pribadi atau kelompok
- h. Buat *signature* pada pesan Anda
- i. Buat dan kelola folder pribadi
- j. Balasan cepat, balas, pesan ke depan
- k. *Flash / Long SMS*
- l. *Simple Auto Reply*
- m. Dukungan Unicode
- n. Lokalisasi (Saat ini tersedia dalam Bahasa Inggris, Bahasa, Ceko, Portugis, Italia, dan Bahasa Rusia)

(Jbubik, 2014) Berikut adalah plugin yang tersedia pada Kalkun:

- a. *Server Alert*: Monitor *server / service* (layanan)

- b. *Blacklist Number*: Blokir pesan dari nomor tertentu
- c. *Simple Autoreply*: Menjawab SMS secara otomatis
- d. *SMS to Email*: Meneruskan (mem-forward) dari SMS ke *e-mail*
- e. *External Script*: Membuat program tanpa Kalkun *code*

Lisensi Kalkun dirilis di bawah GNU *General Public License* . dan bebas untuk dimodifikasi dan / atau didistribusikan ulang *source code* nya.

## 2.14 Domain Name System (DNS)

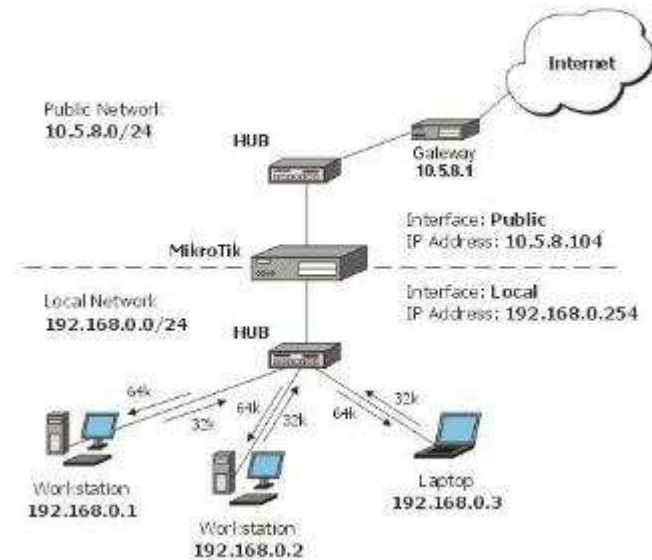
(Syamsudin, 2010) Domain Name System (DNS) adalah suatu sistem yang memungkinkan nama suatu host pada jaringan komputer atau internet ditranslasikan menjadi IP address. Dalam pemberian nama, DNS menggunakan arsitektur hierarki :

- a. Root-level domain: merupakan tingkat teratas yang ditampilkan sebagai tanda titik (.).
- b. Top level domain: kode kategori organisasi atau negara misalnya: .com untuk dipakai oleh perusahaan; .edu untuk dipakai oleh perguruan tinggi; .gov untuk dipakai oleh badan pemerintahan. Selain itu untuk membedakan pemakaian nama oleh suatu negara dengan negara lain digunakan tanda misalnya .id untuk Indonesia atau au untuk australia.
- c. Second level domain: merupakan nama untuk organisasi atau perusahaan, misalnya: microsoft.com; yahoo.com, dan lain-lain.

## 2.15 Mikrotik

(Puspitasari, 2007) Mikrotik adalah salah satu vendor baik *hardware* dan *software* yang menyediakan fasilitas untuk membuat *router*. Salah satunya adalah Mikrotik *Router OS*, ini adalah *Operating system* yang khusus digunakan untuk membuat sebuah *router* dengan cara menginstallnya ke komputer. Fasilitas atau *tools* yang disediakan dalam Mikrotik *Router Os* sangat lengkap untuk membangun sebuah *router* yang handal dan stabil.

Adapun cara kerja mikrotik di asumsikan pada gambar dibawah ini:



Gambar 2.9. Cara kerja Mikrotik

Sumber: Puspitasari, Nila Feby. "Implementasi Mikrotik sebagai solusi router murah dan mudah." (2007).

## 2.16 Routing

(Zamzami, 2013) Sebuah mekanisme yang digunakan untuk mengarahkan dan menentukan jalur yang akan dilewati paket dari satu *device* ke *device* yang berada di jaringan lain. Ada beberapa hal yang mesti diketahui, sebelum kita membuat *routing* diantaranya:

### a. *Routing Host*

Proses *routing* yang dilakukan oleh *host* cukup sederhana. Jika *host* tujuan terletak di jaringan yang sama atau terhubung langsung, *IP* datagram dikirim langsung ke tujuan. Apabila *routing host* menuju jaringan yang berbeda (*internet*), *IP* datagram dikirim ke *default* router. Router ini yang akan mengatur pengiriman *IP* selanjutnya, hingga sampai ke tujuannya. Dalam suatu *table routing* terdapat: *IP address* tujuan, *IP address next hop* router (*gateway*), *Flag* yang menyatakan jenis *routing*. Dalam proses meneruskan paket ke tujuan, *IP* router akan melakukan hal-hal berikut:



1. Mencari di *table routing*, *entry* yang cocok dengan *IP address* tujuan. Jika ditemukan, paket akan dikirim ke *next hop* router atau *interface* yang terhubung langsung dengannya.
2. Mencari di *table routing*, *entry* yang cocok dengan alamat jaringan dari tujuan jaringan. Jika ditemukan, paket dikirim ke *next hop* router tersebut.
3. Mencari di *table routing*, *entry* data yang bertanda *default* , jika ditemukan paket dikirim ke router tersebut.

Masalah *routing* merupakan konsekuensi dari *internet working* . Semakin kompleks topologi dan konfigurasi dari *network* , semakin penting teknik dan sistem *routing* yang handal untuk di-implementasikan dalam *network* . Agar dapat meneruskan paket ke tujuan yang benar, setiap sistem TCP / IP melakukan *routing*. Namun demikian, tidak seluruh sistem TCP / IP menjalankan *routing protocol* . Dengan demikian harus dibedakan antara *routing* dan *routing protocol* .

#### b. *Routing protocol*

Adalah suatu program yang mempertukarkan informasi yang digunakan untuk membentuk *routing table*. Sedangkan *routing* itu sendiri adalah aksi pengiriman datagram-datagram berdasarkan informasi yang diambil dari *routing table*. Ada tiga konfigurasi *routing* yang umum digunakan pada *TCP / IP network* , yaitu:

##### 1. Minimal *routing*

Suatu *network* yang terisolasi dari *network* lain hanya memerlukan minimal *routing*, yakni informasi nomor *network* yang langsung terhubung dengannya. Tabel *routing* minimal ini dibentuk pada saat *interface* dikonfigurasi.

##### 2. *Routing statik*

Suatu *network* yang hanya memiliki sejumlah *gateway* yang terbatas ke *network* lainnya biasanya dikonfigurasi dengan *routing statik*.

##### 3. *Routing dinamis*

Suatu *network* yang memiliki lebih dari satu rute untuk mencapai tujuan yang sama sebaiknya menggunakan *routing dinamis*. *Routing table* dinamis dibentuk berdasarkan informasi yang dipertukarkan oleh *routing*

*protocol*. *Routing protocol* didesain untuk secara dinamis menentukan *routing* berdasarkan kondisi terakhir. *Routing table* senantiasa di-*update* berdasarkan informasi dari setiap *gateway*, sehingga bila suatu rute putus akibat *gateway* yang bersangkutan tidak bekerja, rute ke *network* tujuan melalui *gateway* tersebut akan dipindahkan melalui *gateway* lain. *Routing dinamis* umumnya digunakan untuk jaringan komputer yang besar dan kompleks.

### 2.17 Service Level Agreement (SLA)

(Muhammad, 2013) SLA merupakan kesepakatan formal yang dapat dinegosiasikan guna mengidentifikasi harapan, tanggung jawab, dan memfasilitasi komunikasi antara penyedia produk/layanan (*supplier*) dengan pelanggannya (*costumers*) yang diukur dengan jangka waktu tertentu.

Dari definisi SLA di atas, terdapat dua pihak yang berkepentingan, yaitu pihak penyedia (*supplier*) dan pihak pelanggan (*costumer*). Tentunya keduanya memiliki harapan masing-masing yang bisa saja berbeda. Harapan pelanggan menginginkan produk/layanan tersedia dengan cepat, namun dari pihak penyedia memerlukan waktu proses untuk menyediakan produk/layanan yang dibutuhkan tersebut. Perbedaan harapan inilah yang perlu dikomunikasikan agar tidak terjadi konflik.

Di sinilah diperlukan SLA untuk menjembatani perbedaan harapan, mendefinisikan kewenangan dan tanggung jawab masing-masing pihak sekaligus menjadi alat ukur efektifitas penyediaan produk/layanan oleh *supplier*.

### 2.18 Fiber Driver (Fiber Optik Media Converter)

(Putuasduki, 2015) *Fiber Driver* atau *Fiber Optik Media Converter* adalah perangkat untuk merubah sinyal dari digital ke cahaya dan sebaliknya dari sinyal cahaya menjadi si nyal *digital*, *converter FO* ada dua jenis *single mode* dan *multi mode*. Untuk menghubungkan dari kabel FO ke *converter* di butuhkan menggunakan konektor yang memiliki *standard* tipe yang berbeda beda. Pada kabel serat optik, sambungan ujung terminal atau disebut juga konektor,.

### 2.19 Multi Router Traffic Grapher (MRTG)

(Santoso, 2010) *Multi Router Traffic Grapher* (MRTG) adalah aplikasi yang digunakan untuk memantau beban trafik pada link jaringan. MRTG akan membuat report dalam bentuk harian, mingguan, bulanan dan tahunan berdasarkan *interface router* yang ada. MRTG akan membentuk dokumen dalam bentuk HTML, MRTG itu sendiri terdiri dari *script perl* yang menggunakan *Simple Network Management Protocol* (SNMP) untuk memantau trafik pada *router*.

MRTG merupakan software yang berlisensi GNU *General Public License*. MRTG tidak hanya untuk melihat trafik jaringan, tetapi dapat juga digunakan untuk melihat trafik CPU *load*, *memory load*. Ada banyak *tool* seperti MRTG, seperti *cacti*, *ntop*, *nagios*, *iptraff*. Agar MRTG dapat berjalan baik dibutuhkan *software* penunjang lainnya antara lain SNMP, *Web Server* (apache 2.0).

### 2.20 The Dude

(Mikrotik, 2017) Sebagai salah satu alternatif yang mudah untuk melakukan *monitoring*, Mikrotik membuat sebuah aplikasi yang dinamakan *The Dude*. Dengan aplikasi ini kita bisa melakukan management jaringan network kita. *The Dude* akan secara otomatis membaca atau mendeteksi setiap perangkat yang terhubung ke jaringan yang satu segment. Selain itu dapat juga menyusun dari rancangan topologi jaringan kita, serta dapat melakukan monitoring dan memberikan informasi jika terdapat masalah pada perangkat-perangkat yang terhubung ke jaringan kita.

### 2.21 Linux

(Linux, 2007) Linux adalah nama yang diberikan kepada sistem operasi komputer bertipe Unix. Linux merupakan salah satu contoh hasil pengembangan perangkat lunak bebas dan sumber terbuka utama. Seperti perangkat lunak bebas dan sumber terbuka lainnya pada umumnya, kode sumber Linux dapat dimodifikasi, digunakan dan didistribusikan kembali secara bebas oleh siapa saja.

(Weeks, 2007) Nama "Linux" berasal dari nama pembuatnya, yang diperkenalkan tahun 1991 oleh Linus Torvalds. Sistemnya, peralatan sistem dan pustakanya umumnya

berasal dari sistem operasi GNU, yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi GNU adalah dasar dari munculnya nama alternatif GNU/Linux.

(Lyons, 2007) Linux telah lama dikenal untuk penggunaannya di *server*, dan didukung oleh perusahaan-perusahaan komputer ternama seperti Intel, Dell, Hewlett-Packard, IBM, Novell, Oracle Corporation, Red Hat, dan Sun Microsystems. Linux digunakan sebagai sistem operasi di berbagai macam jenis perangkat keras komputer, termasuk komputer desktop, super k`omputer, dan sistem benam seperti pembaca buku elektronik, sistem permainan video (*PlayStation 2*, *PlayStation 3* dan *XBox*), telepon genggam dan *router*.

## 2.22 phpMyAdmin

(phpMyAdmin, 2010) phpMyAdmin adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi MySQL melalui Jejaring Jagat Jembar (*World Wide Web*). phpMyAdmin mendukung berbagai operasi MySQL, diantaranya (mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (*users*), perizinan (*permissions*), dan lain-lain).

## 2.23 Apache

(Cfadheli, 2014) Apache adalah sebuah nama *web server* yang bertanggung jawab pada *request-response* HTTP dan *logging* informasi secara detail. Selain itu, Apache juga diartikan sebagai suatu *web server* yang kompak, modular, mengikuti standar protokol HTTP, dan tentu saja sangat digemari. Kesimpulan ini bisa didapatkan dari jumlah pengguna yang jauh melebihi para pesaingnya. Sesuai hasil survei yang dilakukan oleh Netcraft, bulan Januari 2005 saja jumlahnya tidak kurang dari 68% pangsa *web server* yang berjalan di Internet. Ini berarti jika semua web server selain Apache digabung, masih belum bisa mengalahkan jumlah Apache.

Saat ini ada dua versi Apache yang bisa dipakai untuk *server* produksi, yaitu versi mayor 2.0 dan versi mayor 1.3. Apache merupakan *web server* yang paling banyak digunakan saat ini. Hal ini disebabkan oleh beberapa sebab, di antaranya adalah karena sifatnya yang *opensource* dan mudahnya mengkostumisasikannya. diantaranya dengan

menambahkan *support secure protocol* melalui *ssl* dan konektifitasnya dengan *database server* melalui bahasa *scripting PHP*.

## 2.24 Winbox

(Dwiyatno, 2014) WinBox adalah sebuah *software* yang digunakan untuk memudahkan *user* masuk dan melakukan konfigurasi pada alat MikroTik baik dengan *mode Command Line Interface (CLI)* maupun *mode Graphical User Interface (GUI)*. Lewat WinBox kita bisa mengkoneksikan diri ke MikroTik *router* dengan alamat IP ataupun MAC address dari MikroTik tersebut.

## 2.25 Network Address Translation (NAT)

(Riadi, 2011) *Network Address Translation (NAT)* adalah suatu metode untuk menghubungkan lebih dari satu komputer ke jaringan *internet* menggunakan satu alamat IP. NAT merupakan teknologi yang memungkinkan IP *Private* dapat membagi koneksi akses *internet* jaringan yang didesain untuk menyederhanakan IP *address*, dan berperan juga untuk melindungi jaringan dan kemudahan serta fleksibilitas dalam administrasi jaringan. Banyaknya penggunaan metode ini disebabkan karena ketersediaan alamat IP *address* yang terbatas. NAT berlaku sebagai penerjemah antara dua jaringan.

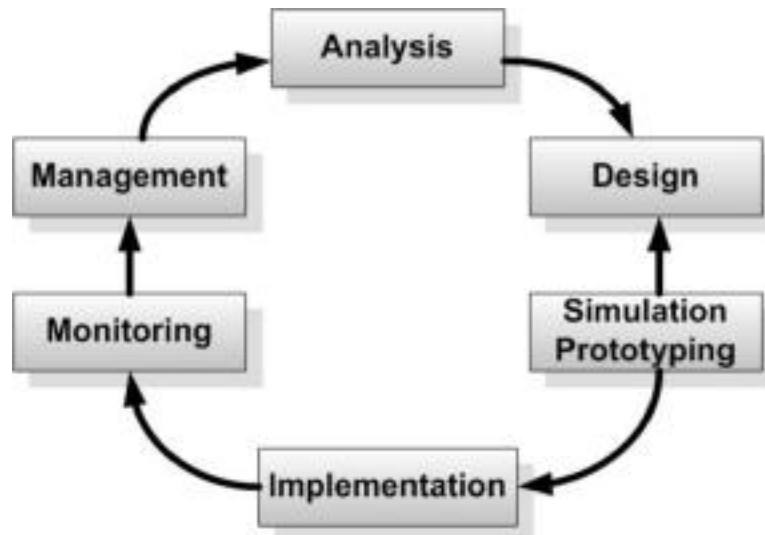
IP *address* sebagai sarana pengalamatan di *internet* semakin menjadi barang mewah dan eksklusif. Tidak sembarangan orang sekarang ini bisa mendapatkan IP *address* yang valid dengan mudah. Oleh karena itulah dibutuhkan suatu mekanisme yang dapat menghemat IP *address*. Logika sederhana untuk penghematan IP *address* adalah dengan membagi suatu nomor IP *address* valid ke beberapa client IP *address* lainnya. Atau dengan kata lain, beberapa komputer bisa mengakses internet walau hanya memiliki satu IP *address* yang valid. Salah satu mekanisme itu disediakan oleh NAT. NAT bekerja dengan jalan mengkonversikan IP *address* ke satu atau lebih IP *address* lain. IP *address* dikonversi adalah IP *address* yang diberikan untuk tiap mesin dalam jaringan internal. IP *address* yang menjadi hasil konversi terletak di luar jaringan internal tersebut dan merupakan IP *address* legal yang valid.

### 2.26 Masquerade

(Nuhamara, 2016) *Masquerade* adalah salah satu fasilitas di mikrotik yang memungkinkan komputer pengguna terhubung ke jaringan internet dengan alamat IP private melalui mikrotik sebagai penerjemah alamat jaringan.

### 2.27 Network Development Life Cycle (NDLC)

(Stiawan, 2009) Metode NDLC mempunyai elemen-elemen yang mendefinisikan fase, tahapan langkah-langkah atau mekanisme proses yang menggambarkan secara keseluruhan proses dan tahapan pengembangan sistem jaringan yang berkesinambungan. NDLC terdiri dari 6 tahapan yaitu analisis, design, simulation prototyping, implementation, Monitoring dan Management.



**Gambar 2.10 Network Development Life Cycle** Stiawan. D. (2009:2) Menjelaskan beberapa tahapan dalam *Network Development Life Cycle*, berikut tahapannya:

#### 1. Analysis

Tahap awal ini dilakukan analisa kebutuhan, analisa permasalahan yang muncul, analisa keinginan user, dan analisa keamanan jaringan yang sudah ada saat ini. Metode yang biasa digunakan pada tahap ini diantaranya;

- a. Wawancara, dilakukan dengan pihak terkait melibatkan dari struktur manajemen atas sampai ke level bawah / operator agar mendapatkan data yang konkrit dan lengkap. Pada kasus di Computer Engineering biasanya juga

melakukan *brainstorming* juga dari pihak vendor untuk solusi yang ditawarkan dari vendor tersebut karena setiap mempunyai karakteristik yang berbeda

- b. Survei langsung lapangan, pada tahap analisis juga biasanya dilakukan *survei* langsung lapangan untuk mendapatkan hasil sesungguhnya dan gambaran seutuhnya sebelum masuk ke tahap design, *survei* biasa dilengkapi dengan alat ukur seperti GPS dan alat lain sesuai kebutuhan untuk mengetahui detail yang dilakukan.
- c. Membaca manual atau blueprint dokumentasi, pada *analysis* awal ini juga dilakukan dengan mencari informasi dari manual-manual atau blueprint dokumentasi yang mungkin pernah dibuat sebelumnya. Sudah menjadi keharusan dalam setiap pengembangan suatu sistem dokumentasi menjadi pendukung akhir dari pengembangan tersebut, begitu juga pada project network, dokumentasi menjadi syarat mutlak setelah sistem selesai dibangun.
- d. Menelaah setiap data yang didapat dari data-data sebelumnya, maka perludilakukan analisa data tersebut untuk masuk ke tahap berikutnya. Adapun yang bisa menjadi pedoman dalam mencari data pada tahap analysis ini adalah:
  1. *User / people*: jumlah user, kegiatan yang sering dilakukan, peta politik yang ada, level teknis user
  2. *Media H/W & S/W*: peralatan yang ada, status jaringan, ketersediaan data yang dapat diakses dari peralatan, aplikasi s/w yang digunakan
  3. *Data*: jumlah pelanggan, jumlah inventaris sistem, sistem keamanan yang sudah ada dalam mengamankan data.
  4. *Network*: konfigurasi jaringan, *volume* trafik jaringan, *protocol*, *monitoring* network yang ada saat ini, harapan dan rencana pengembangan kedepan
  5. *Perencanaan fisik*: masalah listrik, tata letak, ruang khusus, sistem keamanan yang ada, dan kemungkinan akan pengembangan kedepan.

## 2. Design

Dari data-data yang didapatkan sebelumnya, tahap *Design* ini akan membuat gambar design topology jaringan interkoneksi yang akan dibangun,

diharapkan dengan gambar ini akan memberikan gambaran seutuhnya dari kebutuhan yang ada. Design bisa berupa *design* struktur *topology*, *design* akses data, *design* tata layout perkabelan, dan sebagainya yang akan memberikan gambaran jelas tentang project yang akan dibangun. Biasanya hasil dari design berupa;

- a. Gambar-gambar *topology* (*server farm*, *firewall*, *datacenter*, *storages*, *lastmiles*, perkabelan, titik akses dan sebagainya)
- b. Gambar-gambar *detailed estimasi* kebutuhan yang ada

### 3. Simulation Prototype

Beberapa pengguna akan membuat dalam bentuk simulasi dengan bantuan Tools khusus di bidang network seperti *BOSON*, *PACKET TRACERT*, *NETSIM*, dan sebagainya, hal ini dimaksudkan untuk melihat kinerja awal dari network yang akan dibangun dan sebagai bahan presentasi dan *sharing* dengan *team work* lainnya. Namun karena keterbatasan perangkat lunak simulasi ini, banyak para networker's yang hanya menggunakan alat bantu tools *VISIO* untuk membangun *topology* yang akan didesign.

### 4. Implementation

Di tahapan ini akan memakan waktu lebih lama dari tahapan sebelumnya. Dalam implementasi pengguna akan menerapkan semua yang telah direncanakan dan di design sebelumnya. Implementasi merupakan tahapan yang sangat menentukan dari berhasil / gagalnya *project* yang akan dibangun dan ditahap inilah *Team Work* akan diuji dilapangan untuk menyelesaikan masalah teknis dan non teknis. Ada beberapa Masalah-masalah yang sering muncul pada tahapan ini, diantaranya;

- a. Jadwal yang tidak tepat karena faktor-faktor penghambat,
- b. Masalah dana / anggaran dan perubahan kebijakan
- c. Team work yang tidak solid
- d. Peralatan pendukung dari vendor

Oleh karena itu dibutuhkan manajemen project dan manajemen resiko untuk meminimalkan sekecil mungkin hambatan-hambatan yang ada.



## 5. *Monitoring*

Setelah implementasi tahapan *monitoring* merupakan tahapan yang penting, agar jaringan komputer dan komunikasi dapat berjalan sesuai dengan keinginan dan tujuan awal dari user pada tahap awal analisis, maka perlu dilakukan kegiatan *monitoring*. *Monitoring* bisa berupa melakukan pengamatan pada;

- a. Infrastruktur *hardware*: dengan mengamati kondisi *reliability* / kehandalan sistem yang telah dibangun (*reliability* = *performance* + *availability* + *security*),
- b. Memperhatikan jalannya packet data di jaringan (*pewaktuan*, *latency*, *peektime*, *troughput*)
- c. Metode yang digunakan untuk mengamati “kesehatan” jaringan dan komunikasi secara umum secara terpusat atau tersebar

Pendekatan yang paling sering dilakukan adalah pendekatan *Network Management*, dengan pendekatan ini banyak perangkat baik yang lokal dan tersebar dapat di *monitor* secara utuh.

## 6. *Management*

Di manajemen atau pengaturan, salah satu yang menjadi perhatian khusus adalah masalah *Policy*, kebijakan perlu dibuat untuk membuat / mengatur agar sistem yang telah dibangun dan berjalan dengan baik dapat berlangsung lama dan unsur *Reliability* terjaga. *Policy* akan sangat tergantung dengan kebijakan level management dan strategi bisnis perusahaan tersebut. *IT* sebisa mungkin harus dapat mendukung atau *alignment* dengan strategi bisnis perusahaan.

