BAB II

LANDASAN TEORI

Pada bab ini menjelaskan mengenai tempat dilakukannya penelitian dan menjelaskan mengenai teori yang akan digunakan dalam membuat sistem yang akan dibangun.

2.1 Tinjauan Perusahaan

Tinjauan Perusahaan adalah untuk mengetahui keadaan di perusahaan diantaranya adalah sejarah perusahaan.

2.1.1 Sejarah Perusahaan

Visi *Bookstore* adalah sebuah perusahaan yang bergerak di bidang pelayanan literatur kristiani yang melayani orang dari berbagai denominasi. Visi *Bookstore* secara resmi berdiri pada tanggal 8 Desember 1996, dengan pendirinya Bapak Ir. Ichwan S. Chahyadi, M.Sc., di Jalan Kebonjati No.88 Blok C-18 Bandung. Visi dibentuk dengan beban dan kerinduan untuk menjawab kehendak Tuhan bahwa saat ini semakin banyak orang perlu dan membutuhkan pertolongan untuk bertumbuh di dalam pengetahuan dan pengenalan akan pribadi Yesus Kristus sebagai Tuhan dan Juruselamat.

2.2 Landasan Teori

Landasan teori ini menjelaskan teori-teori yang berhubungan dengan sistem informasi yang akan dibangun.

2.2.1 Pengertian kriptografi

Kriptografi adalah cabang ilmu matematika tentang penyandian, ilmu ini membuat sebuah pesan menjadi aman. Kategori aman ini harus meliputi : *Confidentiality* (pesan tidak diketahui oleh pihak yang tidak berkepentingan), *Authenticity* (penerima pesan bisa mengetahui secara pasti siapa yang mengirim pesan itu), *Integrity* (penerima yakin bahwa pesan yang diterimanya tidak diubah saat dikirim), *Non-repudiation* (pengirim pesan tidak bisa membantah bahwa dia pernah mengirim pesan tersebut).

Pertamakali kriptografi digunakan oleh militer adalah untuk komunikasi rahasia yaitu sebelum perang dunia ke-1. Saat itu orang-orang sudah mengetahui bahwa militer menggunakan pengkodean rahasia dalam berkomunikasi, tapi hanya sedikit yang tahu tentang ilmu kriptografi, dan kemudian *National Security Agency* memperkenalkannya sebagai cabang ilmu matematika.

"Kriptografi adalah suatu ilmu yang menciptakan suatu komunikasi data secara aman yang tidak dimengerti atau diterjemahkan maksudnya oleh setiap orang, kecuali orang tertentu yang dimaksud". (Edgar Allan Poe, *Crytography*, 1985)

Dalam hal ini kriptografi dapat diartikan sebagai ilmu yang mempelajari pengacakan data elektronik dengan bantuan program komputer sehingga tidak ada seorangpun yang dapat mengartikannya.

2.2.2 Mekanisme Kriptografi

Keamanan data telah menjadi aspek yang sangat penting dari suatu sistem informasi. Sebuah informasi umumnya hanya ditujukan bagi segolongan tertentu. Oleh karena itu sangat penting untuk mencegahnya jatuh kepada pihak-pihak lain yang tidak berkepentingan. Untuk melaksanakan tujuan tersebut dirancang suatu sistem keamanan database yang berfungsi melindungi sistem informasi.

Salah satu upaya pengamanan sistem infomasi yang dapat dilakukan adalah kriptografi. Kriptografi sesungguhnya merupakan studi terhadap teknik matematis yang terkait dengan aspek keamanan suatu sistem informasi, antara lain seperti kerahasiaan, integritas data, otentikasi, dan ketiadaan penyangkalan.

Suatu sistem kriptografi (kriptosistem) bekerja dengan cara menyandikan suatu pesan menjadi suatu kode rahasia yang dimengerti oleh pelaku sistem informasi saja. Pada dasarnya mekanisme kerja semacam ini telah dikenal sejak jaman dahulu. Bangsa Mesir Kuno sekitar 4000 tahun yang lalu bahkan sudah mempraktekkannya dengan cara yang sangat primitif.

Dalam era teknologi informasi sekarang ini, mekanisme kerja yang sama masih digunakan tetapi tentunya implementasi sistemnya berbeda. Sebelum membahas lebih jauh mekanisme kerja kriptografi modern, berikut ini diberikan beberapa istilah yang umum digunakan dalam pembahasan kriptografi antara lain:

1. *Plaintext*

   Plaintext (*message*) merupakan pesan asli yang akan dikirimkan dan dijaga keamanannya. Pesan ini tidak lain dari informasi tersebut.

2. *Chiphertext*

   Chiphertext merupakan pesan yang telah dikodekan (disandikan) sehingga siap untuk dikirimkan.

3. *Chipher*

   Chipher merupakan algoritma matematis yang digunakan untuk proses penyandian plaintext menjadi *ciphertext*.

4. Enkripsi

   Enkripsi (*Encryption*) merupakan proses yang dilakukan untuk menyandikan *plaintext* sehingga menjadi *chiphertext*.

5. Dekripsi

   Dekripsi (*Decryption)* merupakan proses yang dilakukan untuk memperoleh kembali plaintext dari *chiphertext*.

6. Kriptosistem

   Kriptosistem merupakan sistem yang dirancang untuk mengamankan suatu sistem informasi dengan memanfaatkan kriptografi.

2.2.3 Kriptografi dan Sistem Informasi

Keamanan telah menjadi aspek yang sangat penting dari suatu sistem informasi. Sebuah informasi umumnya hanya ditujukan bagi segolongan tertentu. Oleh karena itu sangat penting untuk mencegahnya jatuh kepada pihak-pihak lain yang tidak berkepentingan. Untuk melaksanakan tujuan tersebut dirancang suatu sistem keamanan yang berfungsi melindungi sistem informasi.

Salah satu upaya pengamanan sistem infomasi yang dapat dilakukan adalah kriptografi. Kriptografi sesungguhnya merupakan studi terhadap teknik matematis yang terkait dengan aspek keamanan suatu sistem informasi, antara lain kerahasiaan, integritas data, otentikasi, dan ketiadaan penyangkalan.

Keempat aspek tersebut merupakan tujuan fundamental dari suatu sistem kriptografi antara lain :

1. Kerahasiaan (*Confidentiality*)

   Kerahasiaan adalah layanan yang digunakan untuk menjaga informasi dari setiap pihak yang berwenang untuk mengaksesnya. Dengan demikian informasi hanya akan dapat diakses oleh pihak yang berhak akses saja.

2. Integritas Data (*Data Integrity*)

   Integritas data merupakan layanan yang bertujuan untuk mencegah terjadinya pengubahan informasi oleh pihak-pihak yang tidak berwenang. Untuk menyatakan integritas data ini harus dipastikan agar sistem

informasi mampu mendeteksi terjadinya manipulasi data. Manipulasi data yang dimaksud di sini meliputi penyisipan, penghapusan, maupun penggantian data.

3. Otentikasi (*Authentication*)

   Otentikasi merupakan layanan yang terkait dengan identifikasi terhadap pihak-pihak yang ingin mengakses sistem informasi (*entity Authentication*) maupun keaslian data dari sistem informasi itu sendiri (*data origin authentication*).

4. Ketiadaan Penyangkalan (*Non-repudation*)

   Ketiadaan penyangkalan adalah layanan yang berfungsi untuk mencegah terjadinya penyangkalan suatu aksi yang dilakukan oleh pelaku sistem informasi.

5. Ketersediaan Data (*Availability*)

   Berhubungan dengan ketersediaan informasi ketika dibutuhkan. Sistem yang diserang dapat menghambat atau meniadakan akses terhadap informasi.

6. Pengendalian Pengaksesan Data (*Access Continul Data*)

   Berhubungan dengan pencegahan penggunaan *resources* yang tidak sah dan memberikan akses terhadap layanan-layanan bagi pengguna yang tidak berhak atas layanan tersebut.

2.2.4 Algoritma RSA

Algoritma RSA melakukan pemfaktoran bilangan yang sangat besar, oleh karena alasan tersebut RSA dianggap aman. Untuk membangkitkan kedua kunci, yang dipilih dua bilangan prima acak yang sangat besar. Skema yang dikembangkan oleh Rivest, Shamir dan Adleman yang mengekspresikan bahwa *plaintext*, dienkripsi menjadi *block-block* yang setiap *block* memiliki nilai biner yang diberi simbol "n", *plaintext block* "m" dan *ciphertext block* "c". Untuk melakukan enkripsi pesan "m" pesan dibagi ke dalam *block-block* numerik yang lebih kecil dari "n" (data biner dengan pangkat terbesar), jika bilangan prima yang panjangnya 200 digit dan dapat menambah beberapa bit 0 dikiri bilangan untuk menjaga agar pesan tetap kurang dari nilai "n".

Dibidang kriptografi RSA adalah sebuah algoritma pada enkripsi *public key*. RSA merupakan algoritma pertama yang cocok untuk *digital signature* seperti halnya enkripsi, dan salah satu yang paling maju dalam bidang kriptografi *public key*. RSA masih digunakan secara luas dalam *protocol electronic commerce,* dan dipercaya dalam mengamankan data dengan menggunakan kunci yang cukup panjang. RSA menyediakan data dan keaslian data (*authentication*) jika Ali mengenkripsi pesan yang dikirim dengan menggunakan *private key*,

seseorang tidak bisa membaca pesan tersebut, serta merubahnya dikarenakan mereka tidak dapat mengdekripsikan pesan tersebut dengan benar.

RSA merupakan sebuah terobosan baru dalam sistem enkripsi data. Sebelum kita mengenal enkripsi data, dimana cara untuk meng-enkripsi dan meng-dekripsi sebuah data, dilakukan dengan kunci yang sama (maksud kunci adalah algoritma yang digunakan). Sebagai contoh enkripsi data sederhana semisal kita punya data abc dan akan dienkripsi menjadi bcd (tiap huruf diganti menjadi hurf yang ada pada urutan berikutnya), maka saat kita akan meng-dekripsi data bcd kembali menjadi abc, cara inilah yang disebut enkripsi simetrik, dimana algoritma dalam meng-enkripsi dan meng-dekripsi mempunyai kesamaan kunci (tinggal membalikkan proses/simetris). Teknik RSA ini disebut juga sistem enkripsi dengan *public key- private key.* Konsep pemikiran dari algoritma RSA adalah sebagai berikut : dianggap semua orang sudah meng-*generate* dua buah kunci (kode untuk enkripsi dan dekripsi, seterusnya akan disebut sebagai sebuah kunci). Kedua kunci ini yang satu dinamakan *private key* dan yang satu lagi dinamakan *public key*. *Private key* sesuai dengan namanya, maka harus disimpan atau dirahasiakan dan hanya diketahui oleh si pemilik kunci, kemudian *public key,* diberikan ke orang lain atau di simpan pada sebuah sumber yang bebas seperti internet.

Seluruh teks (*plaintext*) yang akan di enkripsi harus dikonversikan terlebih dahulu ke dalam bentuk bilangan *sequent*, karena proses enkripsi yang dilakukan adalah proses perhitungan matematis (mod), sehingga teks yang sudah

menjadi bilangan dapat langsung di enkripsi dengan algoritma RSA yang sudah ditetapkan.

Tabel 2.1 Konversi Teks (*Plaintext*) Ke Bilangan *Sequent*

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27

Bilangan ‘0’ dan ‘1’, tidak dijadikan referensi konversi teks (*plaintext*) ke bilangan *sequent,* karena apabila bilangan ‘0’ dan ‘1’, di enkripsi dengan menggunakan kunci *public key*, maka hasilnya tetap, sehingga proses konversi dimulai dari 2.

Contoh proses enkripsi dan dekripsi, diberikan *plaintext* : TAKEHOMETES

Tabel 2.2 Konversi Bilangan Heksadesimal Ke Bilangan Biner

Plaintext	Heksadesimal	Biner
T	54	01010100
A	41	01000001
K	4B	01001011
E	45	01000101
H	48	01001000
O	4F	01001111
M	4D	01001101
E	45	01000101
T	54	01010100
E	45	01000101
S	53	01010011

Plaintext biner :

Tabel 2.3 Tabel *Plaintext* biner

01010100010000010100101101000101010010000100111101001101010001010101000100010101010011

Tabel 2.4 Tabel *Plaintext* Biner Dibagi Dalam 11 Bit Per Blok

Plaintext	Desimal	1024	512	256	128	64	32	16	8	4	2	1
Blok 1	674	0	1	0	1	0	1	0	0	0	1	0
Blok 2	82	0	0	0	0	1	0	1	0	0	1	0

Blok 3	1674	1	1	0	1	0	0	0	1	0	1	0
Blok 4	1156	1	0	0	1	0	0	0	0	1	0	0
Blok 5	1958	1	1	1	1	0	1	0	0	1	1	0
Blok 6	1301	1	0	1	0	0	0	1	0	1	0	1
Blok 7	648	0	1	0	1	0	0	0	1	0	0	0
Blok 8	1363	1	0	1	0	1	0	1	0	0	1	1

2.2.4.1 Enkripsi

Enkripsi (*Encryption*) merupakan proses yang dilakukan untuk menyandikan *plaintext* sehingga menjadi *chiphertext*.

Adapun langkah-langkah untuk proses enkripsi adalah :

1. Pilih dua bilangan prima, p dan q secara acak.
2. Hitunglah n yang merupakan hasil kali bilangan prima p dan q.
3. Pilih satu bilangan prima k sedemikian rupa sehingga k dan (p-1)*(q-1) memiliki faktor pembagi selain bilangan 1.
4. Pilih satu bilangan prima h sedemikian rupa sehingga h mod (p-1)*(q-1) = 1 dan h = - 1, harus dapat dibagi dengan (p-1) *(q-1)

Ali akam mengirim Abu pesan:

Plaintext : TAKEHOMETES

Ali memilih p = 41, q = 53 dan d = 623, dimana d sebagai *public key* Ali

Rumus Enkripsi RSA : $c = m^k \bmod n$

n = pxq = 41x53 = 2173

k(n)=(p-1)x (q-1) = (41-1) x (53-1) = 2080

gcd(h(n),d) = gcd(2080,623) = 1

Tabel 2.5 Tabel Proses Enkripsi

Blok	Desimal (m)	$C=m^d \bmod n$	Chiphertext biner (11 bit)
1	674	1650	11001110010
2	82	1230	10011001110
3	1674	1655	11001110111
4	1156	143	00010001111
5	1958	353	00101100001
6	1301	1495	10111010111
7	648	349	00101011101
8	1363	590	01001001110

Chiphertext biner :

Tabel 2.6 Tabel *Chiphertext* Biner

11001110010100110011101100111011100010001111001011000011011101011100101011101010010011110

Tabel 2.7 Tabel *Chiphertext* Biner

Chiphertext biner 8 bit	Heksadesimal	Karakter
1100 1110	CE	
0101 0011	53	S
0011 1011	3B	;
0011 1011	3B	;
1000 1000	88	ê
1111 0010	F2	≥
1100 0011	C3	
0111 0101	75	u
1110 1010	EA	Ω
0100 1110	4E	N

Menentukan eksponen dekripsi rahasia (a)

a.d = k mod n + 1

a.623 = k mod 2080 + 1 atau $d^{-1} = 207 \bmod 2080$

gcd(2080,207) = 1

jadi, eksponen dekripsi rahasia Ali = 207

2.2.4.2 Dekripsi

Dekripsi (*Decryption)* merupakan proses yang dilakukan untuk memperoleh kembali *plaintext* dari *chiphertext*.

Rumus Dekripsi RSA : $m = c^h \mod n$

a = 207, dimana a sebagai *private key* Ali

n = pxq = 41 x 53 = 2173

k(n) = (p-1)x (q-1) = (41-1)x (53-1) = 2080

gcd(h(n),a) = gcd(2080,207) = 1

Tabel 2.8 Tabel Proses Dekripsi

Blok	C	$m = c^h \mod n$	Plaintext biner (11 bit)
1	1650	674	01010100010
2	1230	82	00001010010
3	1655	1674	11010001010
4	143	1156	10010000100
5	353	1958	11110100110
6	1495	1301	10100010101
7	349	648	01010001000
8	590	1363	10101010011

Plaintext biner :

Tabel 2.9 Tabel *Plaintext* Biner

0101010001000001010010110100010101001000010011110100110101000101010101000100101010011

Tabel 2.10 Tabel *Plaintext* Biner

Plaintext biner 8 bit	Heksadesimal	Karakter
0101 0100	54	T
0100 0001	41	A
0100 1011	4B	K

0100 0101	45	E
0100 1000	48	H
0100 1111	4F	O
0100 1101	4D	M
0100 0101	45	E
0100 0101	54	T
0100 0101	45	E
0101 0011	53	S

2.2.5 Tools Yang Digunakan Dalam Analisis Sistem

1. *Context Diagram* (Diagram Konteks), merupakan bagian dari DFD yang berada pada level yang paling tinggi yang berfungsi untuk menggambarkan hubungan sistem dengan lingkungan luar. Diagram konteks digunakan untuk merepresentasikan sistem melalui sebuah *buble.* Diagram konteks juga memperlihatkan input maupun output dari sistem lain dan juga penyimpanan yang digunakan bersama dengan sistem lain.

2. *Data Flow Diagram* (DFD), merupakan *tool* yang digunakan untuk memodelkan sistem aliran informasi yang terdapat dalam sistem. Didalam DFD dapat dilihat proses-proses yang dilakukan sistem. Interaksi antar proses dan interaksi proses-proses tersebut dalam *entity* luar. DFD juga menunjukkan data yang masuk ke dalam sebuah proses, data-data lain yang diperlukan untuk pengerjaan proses tersebut dan data hasil proses tersebut.

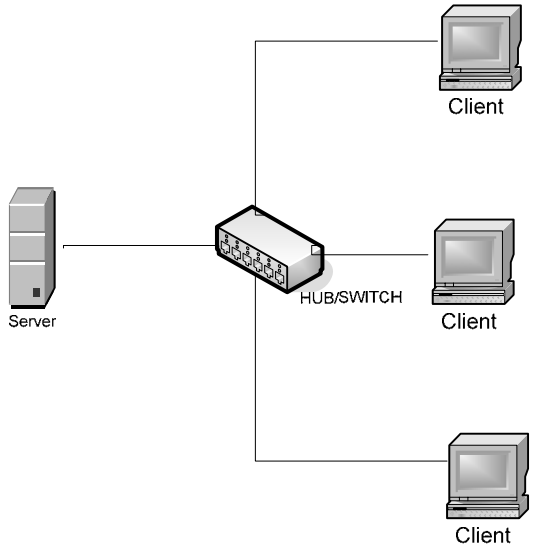
2.2.6 Pengertian Kamus Data

Kamus data adalah deskripsi formal mengenai seluruh elemen yang tercakup dalam diagram alir data atau DFD, elemen-elemen pada kamus data akan menjadi bahan untuk menyusun basis data.

2.2.7 Pengertian Struktur Tabel

Struktur tabel adalah suatu notasi yang digunakan untuk menggambarkan hubungan antara satu tabel dengan tabel yang lain dalam sebuah sistem.

2.2.8 Model *Client – Server*



Gambar 2.1 Model *Client- server*

Konsep jaringan *client-server* ini membedakan dengan jelas kedudukan suatu komputer, mana yang dapat memberikan layanan jaringan (*server*) dan mana yang hanya meminta layanan (*client*). Jika suatu komputer diinstal sebagai *server* yang memberikan segala sumber daya (*resource*) contohnya : printer, modem kepada *client*, maka digunakan suatu aplikasi jaringan yang dinamakan *client- server* program, dimana *server* menggunakan aplikasi jaringan yang disebut *server* program, sementara client menggunakan *client* program untuk dapat berkomunikasi dengan *server* program pada *server*. Beberapa sifat dari *client – server* diantaranya sebagai berikut :

1. *Server* dan *Client* berada pada posisi serta proses yang berbeda.

2. *Server* dan *Client* dapat berjalan pada mesin yang sama atau berbeda.

3. Setiap proses dapat menyembunyikan informasi tersendiri.

4. Komunikasi dapat dilakukan bisa bersifat protokol *peer to peer.*

Koneksitas *Client* ke *Server* :

a. *Serve*r harus terlebih dahulu dijalankan (*running*).

b. *Server* harus menentukan port untuk koneksi yang masuk.

c. *Client* harus mengetahui nama *host* dari *server* beserta *port* yang digunakan untuk koneksi.

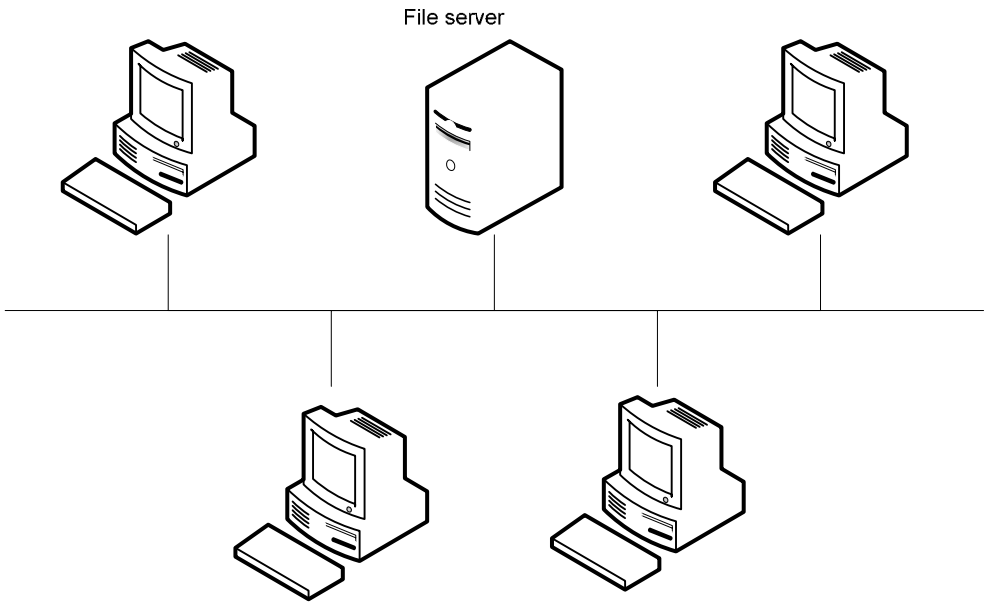
d. *Client* mengirim pesan ke *server* untuk melakukan suatu pekerjaan (*task*), *server* dan *client* bersifat asimetrik.

2.2.9 Topologi Jaringan

Topologi jaringan merupakan suatu struktur atau bagaimana menghubungkan antara simpul dan pusat dalam suatu jaringan. Tiap struktur mempunyai keunikan dan kerugian masing-masing. Topologi yang umum adalah sebagai berikut :

2.2.9.1 Topologi Linear Bus / Garis Lurus

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

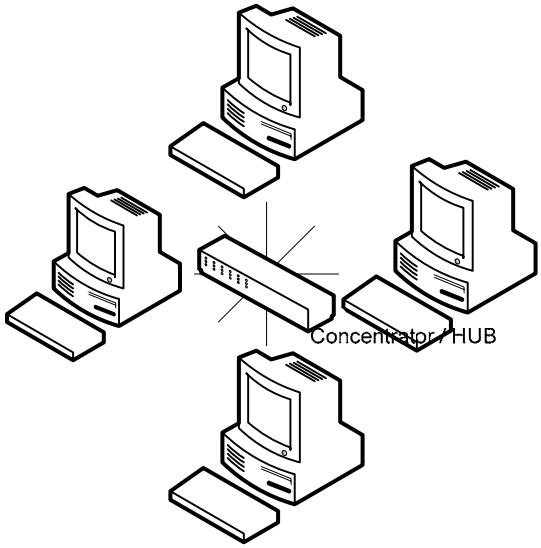


Gambar 2.2 Topologi Linear Bus

2.2.9.2 Topologi Star / Bintang

Topologi model ini didesain dimana setiap *node* (*file server*, *workstation,* dan perangkat lainnya) terkoneksi ke jaringan melewati sebuah *concentrator* baik berupa *HUB* ataupun *SWITCH.* Data yang terkirim ke jaringan akan melewati *concentrator* sebelum melanjutkan ke tempat tujuannya. *Concentrator* akan mengatur dan mengontrol keseluruhan fungsi jaringan. Dia akan bertindak

sebagai *repeater* dalam skala kecil dari aliran data. Konfigurasi pada jaringan model ini menggunakan kabel *Twisted Pair*.



Gambar 2.3 Topologi Bintang

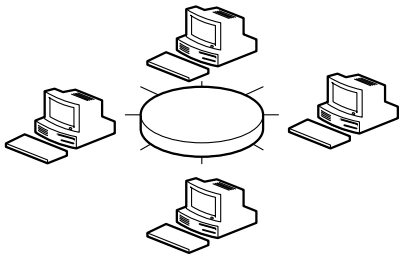
2.2.9.3 Topologi Tree / Pohon

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

2.2.9.4 Topologi Ring / Cincin

Dalam topologi Ring atau Cincin semua *workstation* dan *server* dihubungkan sehingga terbentuk suatu pola lingkaran atau cincin. Tiap *workstation* ataupun *server* akan menerima dan melewatkan informasi dari satu

komputer ke komputer lainnya, bila alamat-alamat yang dimaksud sesuai maka informasi diterima dan bila tidak sesuai informasi akan dilewatkan.



Gambar 2.4 Topologi Ring

2.2.10 Pengertian Database

Database diterjemahkan dalam bahasa Indonesia mempunyai arti sebagai basis data. Basis data adalah sejumlah kumpulan data atau sebuah *file* yang terdiri atas sejumlah *record* (tabel) masing-masing terbentuk dari *field-field* (kolom) dari tipe tertentu, dan kumpulan operasi yang memudahkan pencarian, penyimpanan, kombinasi ulang dan aktivitas sejenisnya.

Dalam mengelola sebuah database pengguna tidak bisa langsung berinteraksi dengan database tersebut, sehingga diperlukan adanya sebuah antarmuka ynag menghubungkan antara pengguna dan database, yang saat ini dikenal dengan nama *Database Management System* (DBMS). Dengan menggunakan DBMS pengguna tidak perlu melacak rincian fisik lokasi *file* dan format, mengindex skema, dan lain-lain. DBMS mengatur seluruh permintaan untuk tindakan database (misal query atau pembaharuan).

Sedangkan dalam beberapa sumber lainnya, database didefinisikan sebagai sekumpulan tabel, *view*, *index*, *trigger*, prosedur dan objek-objek lain, dan tabel adalah inti dari semua database. Tabel menyimpan data yang dikelompokkan

kedalam bentuk baris dan kolom seperti layaknya lembar kerja. Setiap baris mewakili *record* dan setiap kolom adalah atribut atau *field*, dan setiap *field* mengandung satu jenis informasi.[7]

2.2.11 Sekilas Tentang Delphi

Borland Delphi atau yang biasa disebut dengan delphi saja merupakan sarana pemrograman aplikasi visual. Bahasa pemrograman yang digunakan adalah bahasa pemrograman pascal. Delphi merupakan generasi penerus dari turbo pascal. Turbo pascal yang diluncurkan pada tahun 1983 dirancang untuk dijalankan pada sistem operasi DOS, sedangkan delphi yang diluncurkan pertama kali tahun 1995 dirancang untuk beroperasi dibawah sistem operasi *windows*. Borland delphi merupakan salah satu bahasa pemrograman yang semenjak diluncurkan pertama kali langsung dilirik dan diminati oleh para programer komputer. Hal ini disebabkan karena delphi memberikan berbagai fasilitas pembuatan aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan Perangkat Lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrograman yang terstruktur. Keunggulan lain dari delphi adalah dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis *windows*.

Khusus untuk program database borland delphi menyediakan fasilitas objek yang kuat dan lengkap yang memudahkan programer dalam membuat

program. Format database yang dimiliki delphi adalah format database paradox, dBase, MSAcess, ODBC, Sybase, oracle, dll.

2.2.12 MySQL

MySQL adalah *Relational Database Management System* (RDMS) yang didistribusikan secara gratis disebuah lisensi GPL (*General Public License*). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat *close source* atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan (seleksi) dan pemasukan data yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dengan cara kerja optimizernya dalam melakukan proses perintah-perintah SQL yang dibuat oleh *user* maupun program-program aplikasinya sebagai database *server* lainnya dalam *query* data. MySQL adalah satu dari sekian banyak sistem database merupakan terobosan solusi yang tepat dalam aplikasi database.

Menurut ANSI (*American National Standards Institute*), bahasa ini merupakan standar untuk *relational database menagement sistems* (RDBMS).

Pernyataan-pernyataan SQL digunakan untuk melakukan beberapa tugas seperti : *update* data pada database, atau menampilkan data dari database. Beberapa software RDBMS dan dapat menggunakan SQL, seperti : Oracle, Sybase, Microsoft Access, Ingres, dsb. Setiap *software* database mempunyai

bahasa perintah / sintaks yang berbeda, namun pada prinsipnya mempunyai arti dan fungsi yang sama.

Perintah-perintah tersebut antara lain : *"Select", "Insert", "Update", "Delete", "Create", dan "Drop"*, yang dapat digunakan untuk mengerjakan hampir semua kebutuhan untuk memanipulasi sebuah database.

2.2.13 Pengertian Program Aplikasi

Program aplikasi adalah suatu perangkat lunak yang dibangun untuk membantu *user* dalam menyelesaikan suatu masalah tertentu. Dalam hal ini sistem informasi Bookishstore adalah suatu program aplikasi yang di bangun untuk mengelola database penjualan, sehingga seorang *user* dipastikan akan mengalami kesulitan untuk mengelola suatu database bila tanpa perantara dan dibantu oleh suatu program aplikasi.