IMPLEMENTASI SISTEM PENGAMANAN DATA BARANG PADA PT. MATAHARI PUTRA PRIMA, TBK

Murni Marbun
Program Studi Teknik Informatika
STMIK Pelita Nusantara Medan, Jl. Iskandar Muda No 1 Medan, Sumatera Utara 20154, Indonesia
dimpleflorence@yahoo.co.id

Abstrak

Masalah keamanan dan kerahasiaan data merupakan salah satu aspek yang sangat penting dalam perkembangan informasi saat ini. Salah satu solusi untuk mengatasi keamanan dan kerahasian data adalah dengan cara melakukan enkripsi terhadap data yang akan diamankan dan waktu yang dipergunakan untuk penyadian data tidak lama dan tergantung berapa besar data yang akan diamankan.

Untuk menyandikan suatu data dan menerjemahkannya kembali digunakan suatu data yang disebut kunci. Algoritma enkripsi/dekripsi yang didasarkan pada kunci secara garis besar dibedakan menjadi dua macam, yaitu algoritma simetrik dan algoritma asimetrik. Algoritma yang digunakan pada tugas akhir ini adalah algoritma simetrik dengan metode enkripsi/dekripsi RC4.

Kata Kunci: Pengamanan Data, PT. Putra Prima, Tbk

I Pendahuluan

Masalah Kerahasiaan dan keamanan saat melakukan pertukaran data adalah hal yang sangat penting dalam komunikasi data, baik untuk tujuan keamanan bersama, maupun untuk privasi individu. Mereka mengginkan agar data nya tidak di ketahui oleh pihak yang tidak berkepentingan. Ketika pesan disadap pada saat pengiriman data melalui email atau melalui jaringan lain nya maka data tersebut tidak akan berguna lagi, sebab data tersebut tidak ada perlindungan yang di terapkan dalam data tersebut.

Keamanan dan kerahasian data pada jaringan komputer saat ini menjadi isu yang sangat penting dan terus berkembang. Beberapa kasus menyangkut kemanan jaringan komputer saat ini menjadi suatu pekerjaan membutuhkan biaya penanganan yang sedemikan besar seperti, sistem perbankan, sistem bandar udara dan sistem yang lain setingkat atau setara dengan yang dibahas, membutuhan tingkat keamanaan yang sedemikan tinggi. Hal ini di sebabkan karena kemajuan bidang jaringan komputer yang terbuka untuk umum sehingga siapapun, kapanpun, dimanapun mempunyai kesempatan untuk mengakses nya melalui internet.

Untuk menjaga keamanan dan kerahasian data yang sangat penting maka di perlukan enkripsi dan dekripsi, guna agar tidak mudah dicuri atau *Hack* oleh pihak yang tidak berkepentingan, kecuali pihak yang berhak yang ingin melihat data tersebut. Maka penulis menyimpulkan mengangkat judul untuk karya ilmiah atau Penelitian adalah tentang:

“ Implementasi Sistem Pengamanan Data Barang Pada PT. Matahari Putra Prima, Tbk “.

Salah satu hal penting dalam komunikasi menggunakan komputer dan dalam jaringan komputer untuk menjamin kerahasian data adalah enkripsi (akan di jelaskan pada bab selanjutnya), enkripsi disini diartikan sebagai kode atau *chiper*. Sebuah sistem pengkodean menggunakan suatu tabel atau kamus yang telah didefenisikan untuk kata dari bagian data atau informasi yang dikirim. Sebuah *chiper* menggunakan suatu algoritma yang dapat mengkodekan semua aliran data *stream* yang mana nantinya data tersebut tidak dapat dibuka atau dibaca oleh pihak yang tidak berkepentingan, karena sistem *chiper* merupakan suatu sistem pengamanan data melalui kode – kode yang di hasilkan, maka teknik ini digunakan dalam sistem keamanan jaringan komputer.

Ada banyak model dan metode enkripsi, salah satu diantaranya adalah enkripsi dengan metode algoritma *Rivest Code 4* (RC4). Model ini merupakan salah satu algoritma kunci simetris yang berbentuk *stream chiper*. Algoritma ini ditemukan pada tahun 1987 oleh *Ronald Rivest* dan terus berkembang sampai tahun 1994 hingga sekarang.

1.1 Perumusan Masalah

Dari latar belakang di atas penulis dapat menyimpulkan rumusan masalah yaitu :

1. Bagaimana cara membuat aplikasi pegamanan data agar tidak mudah terbaca oleh orang lain yang tidak berkepentingan.
2. Bagaimana sewaktu – waktu user lupa dengan password file enkripsi yang telah dibuat nya.

1.2 Batasan Masalah

Adapun batasan masalah yang dibuat oleh penulis untuk membatasi pembahasan tentang pengamanan data yang akan di bahas, karena sistem pengamanan data sudah banyak beredar di internet atau di buku – buku yang sudah banyak

di rilis oleh berbagai penerbit, berikut batasan masalahnya: 1. Algoritma yang digunakan adalah algoritma RC4.
2. Data pengujian dalam program yang digunakan berupa teks dan file teks (*.txt).
3. Tidak membahas mengenai transmisi data.
4. Tidak membahas lebih lanjut tentang jenis – jenis jaringan dan pengertiannya.
5. Database yang digunakan database microsoft access 2007.
6. Bahasa pemograman yang digunakan adalah bahasa pemograman Microsoft Visual Basic 6.0.

1.3 Tujuan Penelitian

Adapun tujuan penelitian yang dibuat oleh penulis adalah semata – mata untuk membuat aplikasi pengamanan data agar tidak sembarangan orang lain melihat atau memanipulasi data yang dimiliki pengguna tersebut, berikut beberapa tujuan yang akan dibuat oleh penulis :
1. Memperkenalkan apa itu metode RC4.
2. Aplikasi yang akan dibuat dengan cara mengenkripsi data agar tidak dapat mudah dibaca.
3. Membuat cara kerja dari suatu pengamanan data dengan teknik enkripsi algoritma RC4.

2 Konsep Basisdata

Adapun konsep basis data yang dibuat oleh penulis adalah sebagai berikut:

2.1 Pengamanan Data

Pengamanan data adalah sebuah basis data yang dapat diartikan menurut **Stephens dan Plew (2000),** merupakan mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita lakukan sehari – hari untuk berbagai alasan. Dengan basisdata, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basisdata menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria, data pun harus mudah ditambahkan ke dalam basisdata, dimodifikasi, dan dihapus.

Kemudian, **Silberschatz, dkk., (2002)** mendefenisikan basisdata sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan. Sistem manajemen basisdata (DBMS) adalah kumpulan data yang saling berhubungan dengan kumpulan program untuk mengakses data. Tujuan utama sistem manajemen basisdata adalah menyediakan cara menyimpan dan mengambil informasi basisdata secara mudah dan efisien.

Ramakrishman dan Gehrke (2003) menyaktakan basisdata sebagai kumpulan data, umumnya mendeskripsikan aktivitas satu organisasi atau lebih yang berhubungan. Misalnya, basisdata universitas mungkin berisi informasi mengenai hal berikut:

- Entitas sperti mahasiswa, fakultas, mata kuliah, dan ruang kuliah.
- Hubungan antarentitas seperti registrasi mahasiswa dalam mata kuliah, fakultas yang mengajarkan mata kuliah, dan penggunaan ruang untuk kuliah.

Defenisi basisdata, menurut **McLeord, dkk., (2001)** adalah kumpulan seluruh sumber daya berbasis komputer milik organisasi. Sistem manajemen basisdata adalah aplikasi perangkat lunak yang menyimpan struktur basisdata, hubungan antardata dalam basisdata. Basisdata yang dikendalikan oleh sistem manajemen basisdata adalah satu set catatan data yang berhubungan dan saling menjelaskan.

2.2 Algoritma Kriptografi

Algoritma ditinjau dari asal usul kata, kata algoritma mempunyai sejarah yang menarik, kata ini muncul didalam kamus Webster sampai akhir tahun 1957 hanya menemukan kata algorism yang mempunyai arti proses perhitungan dengan bahasa Arab. Algoritma berasala dari nama penulis buku Arab yang terkenal yaitu *Abu Ja'far Muhammad ibnu Musa Al-Khuwarizmi* (al-Khuwarizmi dibaca oleh orang barat menjadi algorism). Kata algorims lambat laun berubah menjadi algorithm.

Defenisi terminiloginya Algoritma adalah urutan langkah – langkah logis untuk menyelesaikan masalah yang disusun secara sistimatis. Algoritma kriptografi merupakan langkah – langkah logis bagaimana menyembunyikan pesan dari orang – orang yang tidak berhak atas pesan tersebut.

Menurut **Shannon** algoritma tersebut harus memiliki kekuatan untuk melakukan:
1. konfusi/pembingungan (confusion), dari teks terang sehingga sulit untuk direkonstruksikan secara langsung tanpa menggunakan algoritma dekripsinya.
2. difusi/peleburan (difusion), dari teks terang sehingga karakteristik dari teks terang tersebut hilang.

Sehingga dapat digunakan untuk mengamankan informasi. Pada implementasinya sebuah algoritma sandi harus memperhatikan kualitas layanan dari keseluruhan sistem dimana dia diimplementasikan. Algoritma sandi yang handal adalah algoritma sandi yang kekuatannya terletak pada kunci, bukan pada kerahasiaan algoritma itu sendiri. Teknik dan metode untuk menguji kehandalan algoritma sandi adalah kriptanalisa.

Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antara dua himpunan yaitu yang berisi elemen teks terang /

plaintext dan yang berisi elemen teks *sandi/ciphertext*. Enkripsi dan dekripsi merupakan fungsi transformasi antara himpunan-himpunan tersebut. Apabila elemen-elemen teks terang dinotasikan dengan P, elemen-elemen teks sandi dinotasikan dengan C, sedang untuk proses enkripsi dinotasikan dengan E, dekripsi dengan notasi D.

Enkripsi : $E(P) = C$

Dekripsi : $D(C) = P$

atau $D(E(P)) = P$

Secara umum berdasarkan kesamaan kuncinya, algoritma sandi dibedakan menjadi:

1. kunci-simetris/symetric-key, sering disebut juga algoritma sandi konvensional karena umumnya diterapkan pada algoritma sandi klasik.
2. kunci-asimetris/asymetric-key.

Berdasarkan arah implementasi dan pembabakan jamannya dibedakan menjadi 2 bagian yaitu:

1. algoritma sandi klasik classic cryptography.
2. algoritma sandi modern modern cryptography.

Berdasarkan kerahasiaan kuncinya dibedakan menjadi 2 bagian yaitu :

1. algoritma sandi kunci rahasia secret-key.
2 algoritma sandi kunci publik publik-key.

Pada skema kunci-simetris, digunakan sebuah kunci rahasia yang sama untuk melakukan proses enkripsi dan dekripsinya. Sedangkan pada sistem kunci-asimentris digunakan sepasang kunci yang berbeda, umumnya disebut kunci publik(public key) dan kunci pribadi (private key), digunakan untuk proses enkripsi dan proses dekripsinya. Bila elemen teks terang dienkripsi dengan menggunakan kunci pribadi maka elemen teks sandi yang dihasilkannya hanya bisa didekripsikan dengan menggunakan pasangan kunci pribadinya. Begitu juga sebaliknya, jika kunci pribadi digunakan untuk proses enkripsi maka proses dekripsi harus menggunakan kunci publik pasangannya.

3.2 Analisa Kebutuhan

Pada bab pembahasan ini, akan di bahas mengenai bagaimana alur kerja algoritma RC4 dan proses – proses algortima RC4 sehingga dapat membentuk teks yang tidak terbaca oleh orang lain.

3.2.1 RC4

Dalam proses enkripsi dan dekripsi, algoritma RC4 menggunakan dua buah *Substitution Box* (S-Box) yaitu *array* sepanjang 256 yang berisi permutasi dari bilangan 0 sampai 255, dan S-Box kedua, yang berisi permutasi merupakan fungsi dari kunci dengan panjang variabel.

Cara kerja algoritma RC4 yaitu inisialisasi S-Box pertama, S[0], S[1], ..., S[255], dengan bilangan 0 sampai 255. Pertama isi secara berurutan S[0]=0, S[1]=1, ..., S[255]=255. Kemudian inisialisasi *arrray* lain (S-Box), misal *array* K dengan panjang 256. Isi *array* K dengan kunci diulangi sampai seluruh *array* K[0], K[1],...., K[255] terisi seluruhnya. Proses inisialisasi S-Box (*Array S)* dapat dilihat pada tabel 3.1 dibawah ini :

Tabel 3.1. Proses Inisialisasi S-Box (*Array* S)

Proses Inisialisasi S-Box (*Array* S)
For i = 0 to 255 S[i] = i

Proses inisialisasi S-Box (*Array* K) dapat dilihat pada tabel 3.2 berikut:

Tabel 3.2. Proses Inisialisasi S-Box (*Array* K)

Proses Inisialisasi S-Box (*Array* K)
For i = 0 to 255 K[i] = kunci [i mod lenght]

Kemudian lakukan langkah pengacakan S-Box yang dapat dilihat pada tabel 3.3 sebagai berikut :

Tabel 3.3. Proses Pengacakan S-Box

Proses Pengacakan S-Box
i = 0; j = 0; for i = 0 to 255 j = (j + S[i] + K[i] mod 255 swap S[i] dan S[j]

Byte K di XOR kan dengan *Plaintext* untuk menghasilkan *ciphertext* atau di XOR kan dengan *ciphertext* untuk menghasilkan *Plaintext*.

3.2.2 Penerapan Algoritma RC4

Untuk membuktikan bagaimana cara kerja algoritma RC4 itu bekerja berikut contoh penerapan enkripsi dan deksripsi algoritma RC4.

3.2.2.1 Contoh Enkripsi Algoritma RC4

Contoh penerapan algoritma RC4 dengan state-*array* 4 *byte*, kenapa state-array 4 *byte*, jika kita menyelesaikan secara manual dengan state-*array* 256 *byte* ini akan bayak memakan waktu lama dalam menyelesaikan satu persatu iniliaisasi array S-Box. Dalam penerapan ini kita akan mengenkripsi kata **sony** dengan kunci **1717**.

Pertama inisialisasi S-Box dengan panjang kunci 4 *byte*, dengan S[0]=0, S[1]=1, S[2]=2, dan S[3]=3 dengan panjang kunci 4 *byte*

juga, sehingga terbentuk la *Array* S *dan Array* K dalam tabe 3.4.

Tabel 3.4. State-*Array* S dan *Array* K

Array S	**0**	**1**	**2**	**3**
Array K	**1**	**7**	**1**	**7**

Setelah membuat Tabel state-*array*, Berikutnya mencampur operasi dimana kita akan menggunakan variabel i dan j ke indeks *array* S[i] dan k[j]. Pertama kita beri nilai inisial untuk i dan j dengan 0. Operasi pencampuran adalah pengulangan rumusan (j + S[i] + K[i] mod 4 yang diikuti dengan penukaran S[i] dengan S[j]. Karena menggunakan *array* dengan panjang 4 *byte* maka algoritma menjadi :

For i = 0 to 4
J = (j + S[i] + K[i] mod 4
Swap S[i] dan S[J]

Dengan algoritma seperti diatas maka nilai awal i=0 sampai i=3 akan menghasilkan *array* S seperti berikut:

Iterasi pertama:

i = 0, maka
j = (j + S[i] + K [i]) mod 4
=(j + S[0] + K[0]) mod 4
= (0 + 0 + 1) mod 4
= 1

Swap S[0] dan S[1] sehingga menghasilkan:

Array S	**1**	**0**	**2**	**3**

Iterasi kedua :

i = 1, maka
j = (j + S[i] + K [i]) mod 4
=(j + S[1] + K[1]) mod 4
= (1 + 0 + 7) mod 4 = 0

Swap S[1] dan S[0] sehingga menghasilkan:

Array S	**0**	**1**	**2**	**3**

Iterasi ketiga:

i = 2, maka
j = (j + S[i] + K [i]) mod 4
=(j + S[2] + K[2]) mod 4
= (0 + 2 + 1) mod 4
= 3

Swap S[2] dan S[3] sehingga menghasilkan:

Array S	**0**	**1**	**3**	**2**

Iterasi keempat:

i = 3, maka
j = (j + S[i] + K [i]) mod 4
=(j + S[3] + K[3]) mod 4
= (3 + 2 + 7) mod 4
= 0

Swap S[3] dan S[0] sehingga menghasilkan:

Array S	**2**	**1**	**3**	**0**

Setelah mendapatkan hasil *array* S dari iterasi keempat, maka proses selanjutnya yaitu meng-XOR kan kata **sony** sebanyak 4 kali dikarenakan *plaintext* yang akan dienkripsi berjumlah 4 karakter sebagai contoh. Hal ini disebabkan dibutuhkan nya 1 kunci dan 1 kali pengoperasian XOR untuk tiap – tiap karakter pada *plainteks. Array* yang digunakan untuk meng-XOR kan adalah *array* dari hasil pencarian nilai *array* terakhir.

Array S	**2**	**1**	**3**	**0**

Inisialisasi

i = 0
j = 0

iterasi pertama

i = (0 + 1) mod 4
= 1
j = (0 + S[1]) mod 4
= (0 + 1) mod 4
= 1

Swap (S[1], S[1]

2	**1**	**3**	**0**

K1 = S[(S[1] + S[1]) mod 4] = S[2 mod 4] = 2
K1 = 00000010

Iterasi kedua

i = (1 + 1) mod 4 = 2
j = (1 + S[2] mod 4 = (1 + 3) mod 4 = 0
swap (S[2], S[0]

3	**1**	**2**	**0**

K2 = S[(S[2] + S[0]) mod 4] = S[5 mod 4] = 2
K2 = 00000010

Iterasi ketiga

i = (2 + 1) mod 4 = 3
j = (0 + S[3] mod 4 = (3+ 0) mod 4 = 3
swap (S[3], S[3]

3	**1**	**2**	**0**

K3 = S[(S[3] + S[3]) mod 4] = S[6 mod 4] = 2
K3 = 00000010

Iterasi keempat

i = (3 + 1) mod 4 = 0
j = (3 + S[0] mod 4
= (3+ 3) mod 4 = 3

swap (S[0], S[3]

0	**1**	**2**	**3**

K2 = S[(S[0] + S[3]) mod 4] = S[3 mod 4] = 3

K2 = 00000011

Setelah menemukan kunci untuk setiap karakter, maka dilakukan operasi XOR antar karakter pada plaintext dengan kunci yang dihasilkan. Berikut adalah tabel 3.5 ASCII untuk tiap karakter pada plaintext yang digunakan.

Tabel 3.5 kode ASCII untuk setiap karakter plaintext yang digunakan

HURUF	KODE ASCII (8 Bit)
s	**01110011**
o	**01101111**
n	**01101110**
y	**01111001**

Proses XOR dari kunci bisa dilihat pada tabel 3.6:

Tabel 3.6 proses XOR kunci enkripsi dengan plainteks pada enkripsi

	s	**o**	**n**	**y**
Plaintext	01110011	01101111	01101110	01111001
Key	00000010	00000010	00000010	00000011
Chipertext	01110001 **(s)**	01101101 **(/)**	01101100 **(÷)**	01111010 **(Ü)**

Maka dapat disimpulkan untuk mencari hasil enkripsi Chipertext diperlukan rumus dibawah ini :

Berikut contoh penjumlahan bilangan biner:

Plaintext + key (Kunci) = Chipertext

0 + 0 = 0

0 + 1 = 1

1 + 1 = 0

3.2.2.2 Contoh Dekripsi Algoritma RC4

Setelah hasil enkripsi telah didapatkan maka proses selanjutnya adalah proses pendekripsian, dilakukan dengan proses XOR antara kunci dekripsi yang sama dengan kunci dekripsi dengan ciphertext yang dapat dilihat pada tabel 3.7 dibawah ini :

Tabel 3.7 Proses XOR kunci dekripsi dengan chipertext pada Dekripsi

Chipertext	01110001 **(s)**	01101101 **(/)**	01101100 **(÷)**	01111010 **(Ü)**
Key	00000010	00000010	00000010	00000011
Plaintext	01110011	01101111	01101110	01111001
	s	**o**	**n**	**y**

3.2.3 Prangkat Pendukung

Adapun perangkat pendukung yang dibuat oleh penulis adalah sebagai berikut:

1. Komputer atau laptop
2. Operasi sistem (windows 7 atau windows xp)
3. Dan pendukung perangkat lunak lain nya

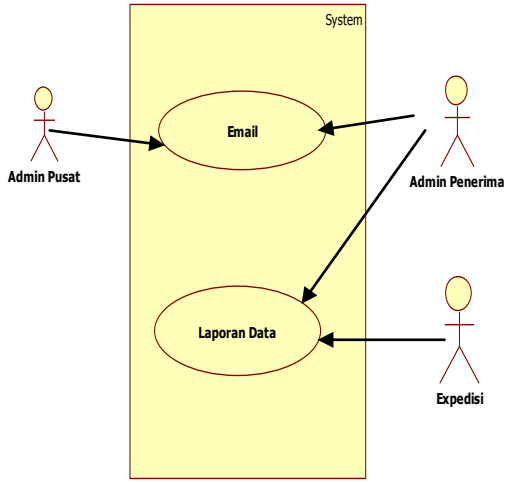
3.2.4 Sistem Yang sedang Berjalan

Adapun sistem yang sedang berjalan dipersahaan dimana penulis meneliti adalah sebagai berikut:

1. Pengiriman data barang dikatagorikan 2 User yang disebut *admin* pusat dan *admin* penerima.
2. Pengiriman data barang yang dilakukan *admin* melalui jaringan internet.
3. *Admin* mengirimkan data barang memanfaatkan fasilitas email (gmail ataupun y.mail.com).
4. Data barang dikirim melalui email pada tanggal yang sudah ditentukan, *Admin* penerima memberitahu kepada expedisi bahwa tanggal yang sudah ditentukan akan masuk barang dan memberikan data barang singkat dari *admin* pusat.
5. Setelah barang masuk, maka pihak *expedisi* meng-check barang beserta data barang yang sudah diterima dari supir pengangkutan dan memberikan laporan barang masuk ke *admin* Penerima.
6. *Admin* penerima akan meng-check ulang data barang yang dikirim oleh admin pusat sesudah dicheck oleh pihak expedisi.
7. Data barang yang sudah selesai di check dan data barang klop atau sesuai dengan data yang dikirim, *admin* penerima akan meng-kirim kembali data tersebut ke *admin* pusat.

Setelah membuat penjelasan kalimat diatas, maka penulis akan membuat gambaran sistem yang sedang berjalan, yang disebut dengan useCase dimana penulis melakukan penelitian.

Perusahaan penulis maksud didalam penjelasan bab ini adalah **PT. MATAHARI *DEPARTMENT STORE* MEDAN FAIR.** Berikut gambar 3.3 sistem yang sedang berjalan di perusahaan tersebut.



Gambar 3.3 Sistem yang sedang berjalan

3.2.5 Sistem Yang Diusulkan

Pengiriman data barang yang dilakukan oleh perusahaan dimana tempat penulis melakukan penelitian adalah secara online, dimana perusahaan tersebut mengkirimkan data mereka antar email, yang mana nanti nya data tersebut dapat dibuka oleh pihak penerima ataupun pihak pengirim. Pengiriman data secara online tidaklah aman untuk digunakan, kenapa demikian karena jaringan internet sangat la terbuka untuk umum, dan siapa saja, dimana saja semua orang dapat mengakses jaringan internet.

Maka dari itu penulis membuat sistem yang diusulkan untuk menjaga tetap rahasia document tersebut jika melakukan pengiriman data secara online. Jenis pengamanan data yang dibuat oleh penulis adalah enkripsi dan dekripsi data dengan metode RC4, dimana nanti nya pengirim dapat mengenkripsi data yang akan dikirim ke penerima data dan data tersebut dapat di dekripsi untuk membaca kembali file data tersebut dengan menggunakan kunci yang sudah di tentukan.

3.3 Teknik Pengumpulan Data

Pengumpulan data adalah prosedur yang sistimatis dan standar untuk memproleh data yang diperlukan. Selalu ada hubungan antara metode mengumpulkan data dengan masalah penelitian yang ingin dipecahkan. Secara umum metode pengumpulan data dapat dibagi atas beberapa kelompok:

1. Studi Pustaka
   Mempelajari buku – buku referensi sebagai pendukung tugas akhir ini dan hasil penelitian sebelumnya yang pernah dilakukan oleh orang lain. Tujuannya ialah untuk mendapatkan landasan teori mengenai masalah yang akan diteliti.
2. Observasi/Pengamatan
   Pengumpulan data dengan obsevasi langsung atau dengan pengamatan langsung adalah cara pengambilan data dengan menggunakan mata tanpa ada pertolongan alat standar lain untuk keperluan tersebut. Dengan cara pengamatan, data yang langsung mengenai perilaku yang tipikal dari objek dapat dicatat segera, tanpa menggantungkan data dari ingatan orang lain.
3. Wawancara
   Wawancara adalah proses memperoleh keterangan untuk memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab, sambil bertatap muka antara si penanya atau pewawancara dengan si penjawab atau responden dengan menggunakan alat yang dinamakan interview guide (panduan wawancara).

3.4 Perancangan Sistem

Adapun perancangan sistem yang dibuat penulis adalah sebagai berikut:

3.4.1 Reqrutment Analisis

Aplikasi sistem pengamanan data dirancang dengan menggunakan bahasa pemograman *Microsoft Visual Basic 6.0* dengan beberapa komponen standard seperti:

1. *Command Button*, sebagai tombol.
2. *Text Box*, sebagai tempat teks.
3. *Label*, untuk menampilkan tulisan.
4. *CommonDialog*, Sebagai alat penyambung *open file* folder.
5. *ProgressBar*, untuk memuat loading .
6. *Timer*, untuk menjalankan waktu atau time .
7. *DataGrid*, tabel penampilan data .
8. *Image*, untuk menampilkan gambar.
9. *Shape.*

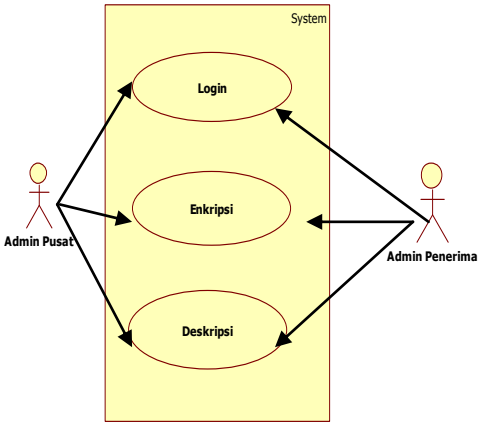
Aplikasi ini memiliki beberapa buah *form* sebagai berikut:

1. *Form* Login.
2. *Form* Menu.
3. *Form* Kunci Data.
4. *Form* Tabel Data.

3.4.2 Design Sistem

Adapun desain perancangan sistem yang diusulkan dalam pengamanan data adalah sebagai berikut:

1. Use Case Diagram yang Diusulkan
   Use Case Diagram menggambarkan secara grafis prilaku aplikasi. Adapun *Use Case* gambar 3.4 dibawah ini adalah *system* pengamanan data yang diusulkan.



Gambar 3.4 *Use Case* Sistem yang diusulkan

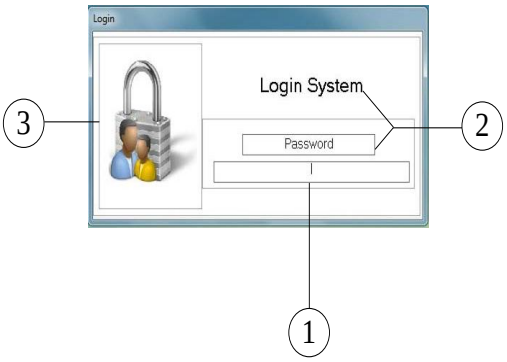
IV Hasil Penelitian

4.1.1 Tampilan Sistem

Pada bab ini, penulis akan membuat tampilan sistem yang dirancang dan sedikit penjelasan tentang masing – masing sistem. Berikut tampilan dan penjelasan nya:

1. Form Login

Form login, berisi beberapa komponen standart seperti : *textbox* sebagai tempat string password login untuk masuk ke tampilan menu, *label* menampilkan teks yang tidak dapat diperbaiki oleh pemakai, *Image* sebagai gambar simbol login pada form login. Berikut tampilan *form* login., dapat dilihat pada gambar 4.1 berikut:



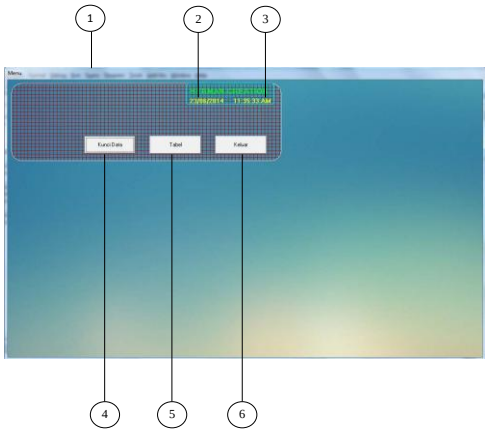
Gambar 4.1 *Form* Login

Keterangan:

1) : *textbox* login.
2) : *label* login system, password.
3) : *image* login.

2. Form Menu Utama

Form utama, menampilkan *command button* yang berfungsi sebagai link atau penghubung untuk membuka *form* kunci data, *form* tabel data dan tombol untuk menutup *form* menu. Berikut tampilan *form* menu utama, dapat dilihat pada gambar 4.2 dibawah:



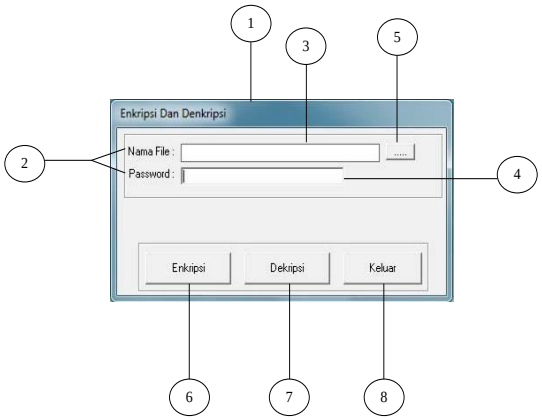
Gambar 4.2 *Form* Menu Utama

Keterangan :

1 : title bar.
2 : *label*, untuk menampilkan date/month/year.
3 : *label*, untuk menampilkan time.
4 : *command button*, untuk membuka *form* kunci data.
5 : *command button*, untuk membuka *form* tabel data.
6 : *command button*, untuk menutup *form* menu.

3. Form Kunci Data

Form kunci data, berisikan komponen standart seperti: *textbox* sebagai tempat string password enkripsi/dekripsi dan sebagai tempat nama data file yang telah dicari, *label* menampilkan teks yang tidak dapat diperbaiki oleh pemakai, *command button* enkripsi sebagai tombol enkripsi file, *command button* dekripsi sebagai tombol dekripsi file, *command button* sebagai tombol untuk menutup *form* kunci data. Berikut tampilan *form* kunci data dapat dilihat pada gambar 4.3 dibawah:



Gambar 4.3 *Form* Kunci Data

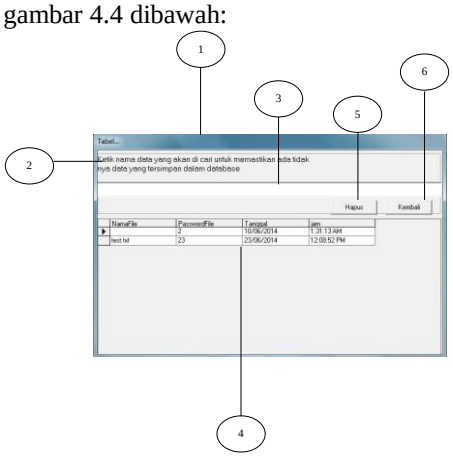
Keterangan:

1 : title bar.
2 : *label*, menampilkan string.
3 : *textbox*, sebagai tempat nama string yang telah dicari.

4 : *textbox,* sebagai tempat password string enkripsi/dekripsi.
5 : *command button*, sebagai tombol cari file.
6 : *command button*, sebagai tombol enkripsi.
7 : *command button*, sebagai tombol dekripsi.
8 : *command button*, sebagai tombol menutup *form* kunci data.

4. Form Tabel Data

Form tabel data, berisikan beberapa komponen standart seperti: *textbox* sebagai tempat string pencarian nama data, *label* menampilkan teks yang tidak dapat diperbaiki oleh pemakai, *datagrid* berfungsi sebagai tabel data dari database yang sudah dikoneksikan, *command button* hapus berfungsi sebagai menghapus data dari database, *command button* kembali untuk menutup *form* tabel data. Berikut tampilan *form* tabel data, dapat dilihat pada gambar 4.4 dibawah:



Gambar 4.4 *Form* Tabel Data

Keterangan:
1 : title bar.
2 : *label*, menampilkan string.
3 : *textbox*, tempat pencarian data.
4 : *datagrid*, tampilan data dari database.
5 : *command button*, tombol untuk menghapus data
6 : *command button*, tombol untuk menutup *form* tabel data

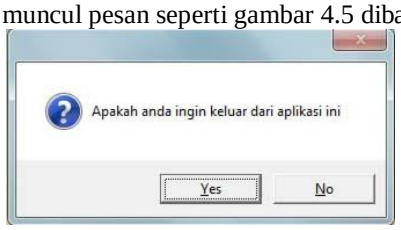
4.1.1.1Langkah – Langkah Menjalankan Sistem

Pada bab ini, penulis akan menjelaskan bagaimana menjalankan sistem yang dirancang. Berikut penjelasan nya:

1. Form Login

Saat sistem dibuka yang pertama kali muncul adalah *form* login, *form* login ini berfungsi sebagai pengamanan pertama untuk membuka sistem menu. Sistem menu sangatlah penting dalam rancangan ini, kenapa demikian, didalam *form* menu ini lah rahasia pengamanan data dan menyimpan password file yang di enkripsi. Kunci pengamanan data tersebut tersimpan di *form* tabel data yang terkoneksi langsung kedalam database. Berikut langkah kerja *form* login:

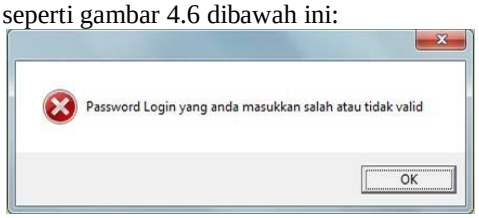
1. Masukkan password yang telah ditentukan, didalam *texbox.*
2. Setelah user mengketik password login dalam *textbox* tekan *enter* untuk masuk.
3. Jika ingin keluar dari *form* login tekan *esc* untuk keluar, setelah menekan *esc* maka akan muncul pesan seperti gambar 4.5 dibawah ini:



Gambar 4.5 Pesan Keluar Dari *Form* Login

Jika memilih *Yes* maka aplikasi akan secara otomatis keluar, jika memilih *No* akan kembali ke *form* login.

4. Jika password benar maka *form* menu akan terbuka, jika salah maka akan muncul pesan seperti gambar 4.6 dibawah ini:



Gambar 4.6 Pesan Tidak *Valid* Login

2. Form Menu

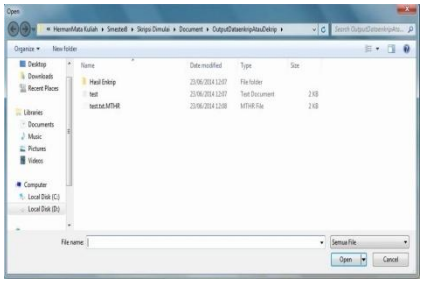
Form menu adalah sebuah sistem dimana sistem itu sebagai penghubung untuk membuka setiap sistem yang dirancang didalam nya. Berikut cara mengunakan nya:

1. Dalam *form* menu terdapat 3 *commond button:* kunci data, tabel dan keluar
   serta tanggal dan jam sesuai format operasi sistem yang digunakan.
2. Jika anda memilih *command button* kunci data, maka akan terbuka *form* enkripsi/dekripsi.
3. Jika anda memilih *command button* tabel, maka akan muncul *form* tabel data.
4. Jika anda memilih *command button* keluar, maka *form* menu akan ditutup.

3. Form Kunci Data

Form kunci data adalah sebagai tempat dimana terjadi nya proses enkripsi dan dekripsi. Berikut cara mengunakan nya:

1. Jika user ingin mengenkripsi data hal pertama yang dilakukan user adalah mencari data yang ingin di enkripsi, dalam pembahasan ini dicontoh kan nama sebuah file dengan nama *test*.txt. langkah selanjutnya cari file dengan cara menekan tombol cari (...). berikut contoh gambar 4.7 *open* file:

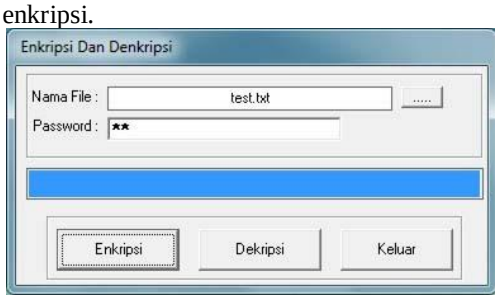


Gambar 4.7 *Open* File

2. Setelah file sudah dicari, masukkan password kedalam *texboxt* untuk mengkripsi data, jika anda tidak memasukkan password ketika anda menekan tombol enkripsi, maka akan muncul pesan seperti gambar 4.8 dibawah ini:

Gambar 4.8 Pesan Password Enkripsi

3. Langkah selanjutnya adalah ketika user menekan tombol enkripsi maka akan terjadi proses enkripsi, pada saat proses enkripsi sedang berjalan sistem akan secara otomatis merekam *nama file, password file, jam dan tanggal* yang akan di simpan kedalam database. Berikut gambar 4.9 proses enkripsi.



Gambar 4.9 Proses Enkripsi

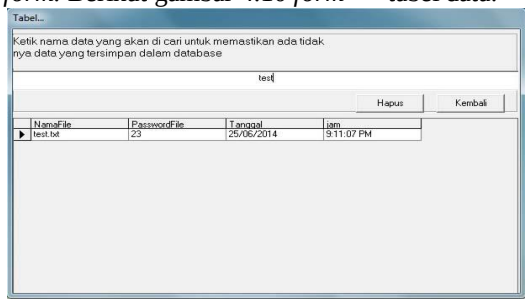
4. Setelah proses selesai maka sistem akan membentuk file dengan nama *test*.txt menjadi *text*.txt.MTHR. Untuk peletakan file yang sudah selesai di enkripsi, maka tempat file yang sudah selesai di enkripsi akan di letakkan pada folder dimana user tersebut mencari file dan disitulah file tersebut akan diletakkan.

Untuk proses dekripsi sama hal nya dengan proses enkripsi yang sudah dibahas diatas. Kegunaan proses dekripsi itu sendiri adalah untuk mengembalikan file yang di enkripsi menjadi file semula.

5. Form Tabel Data

Form tabel data, fungsi dari *form* ini sendiri adalah untuk menampilkan data dari database yang sudah direkam sewaktu melakukan proses enkripsi dilakukan. Didalam *form* ini terdapat beberapa komponen standart yang digunakan sebagai berikut:

1. *Textbox* yang berfungsi untuk mencari data yang di inginkan oleh user, hanya memasukkan kata kunci atau kalimat yang ingin dicari maka secara otomatis *datagrid* akan mengarahkan tanda panah ke arah data yang di inginkan.
2. Tombol hapus berfungsi untuk menghapus data yang ada dalam database.
3. Tombol kembali berfungsi untuk menutup *form.* Berikut gambar 4.10 *form* tabel data:



Gambar 4.10 *Form* Tabel

4.2 Pembahasan

Pada bab ini, penulis akan membahas kelemahan dan kelebihan sistem yang sudah dirancang. Adapun kelemahan dan kelelebihan nya adalah sebagai berikut:

4.2.1 Kelebihan Sistem

Adapun kelebihan sistem yang dirancang oleh penulis adalah sebagai berikut:

1. Mampu melakukan pengacakan teks dengan kombinasi kunci yang telah dibuat dan tidak dapat dibaca oleh orang lain yang ingin mengetahui nya.
2. Sangat mudah untuk digunakan dan memori sistem yang dirancang sangatlah kecil.
3. Tidak memberatkan kinerja sistem operasi yang digunakan
4. Untuk mendapatkan kunci enkripsi yang dirancang membutuhkan waktu yang sangat lama atau membutuhkan waktu 1000 kali keturunan untuk dapat memecahkan pencarian kunci dan teks asli yang sudah dienkripsi.
5. Menyimpan secara otomatis nama file, password file, jam dan tanggal ke dalam database.
6. Memiliki *form* tabel data apabila user lupa password file enkripsi yang ditampilkan di *datagrid*.

4.2.2 Kelemahan Sistem

Adapun kelemahan sistem yang dirancang oleh penulis adalah sebagai berikut:

1. Sistem ini tidak dapat mengganti password login secara langsung.

2. Nama file, password file, jam dan tanggal tidak dienkripsi sewaktu proses penyimpanan string tersebut kedalam database.
3. User tidak dapat membuka sistem menu apabila user lupa password login.
4. Password login yang digunakan adalah password permanen dari penulis, yang disimpan di database.
5. Jika database dihapus maka aplikasi ini tidak dapat dijalankan, dikarenakan aplikasi dan database terhubung baik dari segi login maupun menu dan *form* lain nya.

5. Kesimpulan

Berdasarkan hasil penelitian yang telah dibuat, maka penulis dapat menyimpulkan penelitian yang sudah dibahas pada bab – bab sebelum nya dan memberikan saran bagi pembaca untuk kemajuan penelitian dan program yang sudah dibuat oleh penulis.

Berikut ini beberapa kesimpulan dari hasil penelitian yang telah dibuat oleh penulis:

1. Berdasarkan hasil penelitian yang dilakukan oleh penulis dari sebuah perusahaan instansi dimana penulis melakukan penelitan dapat disimpulkan untuk pengamanan data agar tidak terbaca oleh orang lain, maka penulis membuat suatu sistem pengamanan data yang disebut enkripsi dan dekripsi. Dimana sistem ini digunakan untuk mengamankan data-data yang sangat penting dan tidak dapat dibaca oleh orang lain, kecuali orang yang berhak yang ingin melihat data tersebut. Algoritma yang digunakan oleh penulis adalah algoritma RC4 untuk pengamanan data yang telah dibuat.
2. Berdasarkan hasil penelitian pada suatu sistem enkripsi dan dekripsi yang telah dibuat muncul sebuah pertanyaan, seandainya user lupa password enkripsi data yang telah diproses bagaimana cara menemukan nya kembali. Password file atau data yang sudah dienkripsi tidak dapat ditemukan kembali, kenapa demikian password dan setiap kata yang ada dalam file sudah di XOR kan atau diacak kata dari setiap kata yang telah diproses dalam sistem dengan kombinasi password yang diketik, maka dari itu penulis membuat suatu *form* tabel yang berfungsi dimana *form* ini akan merekam nama file, password file, jam dan tanggal enkripsi yang telah diproses yang disimpan didalam database.

Referensi:

Dony Ariyus. *Kriptografi Keamanan Data Dan Komunikasi*, Yogyakarta: Graha Ilmu, *2006.*

Rifki Sadikin. *Kriptografi Untuk Keamanan Jaringan dan Implementasinya Dalam Bahasa Java*, Yogyakarta: ANDI OFFSET, *2012.*

Theresia Ari Prabawati (Editor), Sri Sulistiyani (Setting), Leo Agung (Desain Cover), Dee Setiawan (Korektor). *Microsoft Visual Basic 6.0 untuk Pemula*, Yogyakarta:ANDI; Madiun: MADCOMS, *2008.*

Janner Simarmata, Iman Paryudi. *Basis Data*, Yogyakarta: ANDI OFFSET, *2006, 2010.*

Hari Darmawan. *Matahari Department Store Peraturan Perusahaan*, Jakarta Selatan, *2014 – 2016.*

Marlinda, Linda. *Sistem Basis Data*. Edisi Pertama. Yogyakarta: Andi, *2004.* Sutabri, Edy. *Basis Data dalam Tinjauan Konseptual*, Edisi Pertama.Yogyakarta: Andi, *2011.*

MADCOMS. *Seri Panduan Pemograman Microsoft Visual Basic 6.0.*Edisi Kedua. Yogyakarta: ANDI OFFSET, *2004.*

Bagus Karuniawan, *Sistem Informasi Manajemen Dengan Visual Basic 6*, Edisi Pertama. Yogtakarta: ANDI OFFSET, *2002*.

Wahana Komputer, *The Best Encryption Tools*, Jakarta: PT Elex Media Komputerindo.

Anhar, *Cara Mudah Mengamankan Data Kompuer & Laptop*. Edisi Pertama Jakarta Selatan: MEDIA KITA, *2010*.

Goutam Paul Subhamoy Maitra, KENNETH H. ROSEN, *RC4 Stream Cipher And Its Variants*, Taylor & Francis Group: CRC Press, *2012.*

Ir. Pandapotan Sianipar, *Cara Mudah Menguasai Word 2007*. Jakarta: PT Elex Media Komputindo, *2008.*

Deris Stiawan, *Sistem Keamanan Komputer*. Jakarta: PT Elex Media Komputindo. *2005*.

MADCOMS, *Mahir Dalam 7 Hari: Microsoft Visual Basic 6.0 + Crytal Report 2008*, Yogyakarta: ANDI OFFSET, *2010.*

Julius Hermawan, *Analisa Desain & Pemograman Berorientasi Obyek Dengan UML*, Yogyakarta: ANDI OFFSET, *2010*.

Drs. Zulkifli Amsyah, MLS, *Manajemen Sistem Informasi*, Jakarta: PT Gramedia Pustaka Utama, *1977/2005*.

Matahari Department Store. *Visi Dan Misi, 2012*