Pengembangan *Prototype* Sistem Kriptografi Untuk Enkripsi Dan Dekripsi Data *Office* Menggunakan Metode Blowfish Dengan Bahasa Pemrograman Java

Mohamad Natsir
Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana
Jl. Raya Meruya Selatan, Kembangan, Jakarta, 11650
m.natsir@mercubuana.ac.id

Abstract **-- The development of computer technology that makes computers much more rapidly found from government agencies, housing, schools, internet cafes, household, personal or even called to make a laptop computer as a requirement for all circles. The works were formerly manual now vastly computerized ranging from homework from school to international companies and all the jobs it is an important information. Like the important information, the security system is a crucial part of a computer application and one way to protect our data from the parties is irresponsible to perform cryptographic would like to use that compiler uses blowfish algorithm. Blowfish or OpenPGP.Cipher.4 is included in the class encryption Symmetric Cryptosystem, the encryption method is similar to the Data Encryption Standard (DES - like - Cipher). Created for use on computers that have a large microprossesor (32 - bit or more with large data caches). Blowfish was developed to meet the design criteria allows the implementation where the optimal state can reach 26 clock cycles perbyte compact, which can run on less than 5 KB of memory, the simple algorithm so easy to determine guilt, and the security which the variable key length varies (minimum 32 bits, 448 bits maximum, Multiple 8 bits, 128 bits default).**

Keywords: **cryptography, symmetric key, blowfish algorithm, encryption, decryption**

Abstrak **- Perkembangan teknologi komputer membuat komputer lebih cepat ditemukan. Instansi pemerintah, perumahan, sekolah, kafe internet, rumah tangga, pribadi membuat komputer dan laptop sebagai kebutuhan untuk semua kalangan. Karya-karya yang sebelumnya manual kini jauh terkomputerisasi, mulai dari pekerjaan rumah anak sekolah sampai ke perusahaan-perusahaan internasional. Semua pekerjaan tersebut merupakan informasi penting. Mengingat informasi itu penting, sistem keamanan merupakan bagian penting dari aplikasi komputer. Salah satu cara untuk melindungi data dari pihak yang tidak berkepentingan, adalah dengan melakukan kriptografi menggunakan compiler dengan algoritma Blowfish. Blowfish atau OpenPGP.Chiper.4 termasuk dalam kelas enkripsi Symmetric Cryptosystem di mana metode enkripsinya sama dengan Data Encryption Standard (DES - like - Cipher) dan dibuat untuk digunakan pada komputer yang memiliki microprossesor besar (32 - bit atau lebih dengan cache data yang besar). Blowfish dikembangkan untuk memenuhi kriteria desain, memungkinkan implementasi di mana keadaan optimal dapat mencapai 26 jam siklus perbyte kompak, yang dapat berjalan pada memori kurang dari 5 KB. Algoritma ini sederhana sehingga mudah untuk menentukan kesalahan dan keamanan dengan panjang variabel kunci bervariasi (minimum 32 bit, 448 bit maksimum, Multiple 8 bit, 128bit default).**

Kata Kunci: **Kriptograpi, Kunci Simetris, Algoritma Blowfish, Enkripsi, Dekripsi.**

I. PENDAHULUAN

Perkembangan teknologi komputer yang semakin pesat menjadikan komputer banyak dijumpai mulai berbagai organisasi, perusahaan, ataupun pihak-pihak lain telah memanfaatkan teknologi basis data untuk menyimpan dan mengolah file dokumen organisasi atau perusahaannya. Saat ini, keamanan terhadap file yang tersimpan sudah menjadi persyaratan mutlak. Pengamanan terhadap jaringan komputer yang terhubung dengan tempat penyimpan dokumen sudah tidak lagi menjamin keamanan file dokumen karena kebocoran data dapat disebabkan oleh "orang dalam" atau pihak-pihak yang berlangsung berhubungan dengan file dokumen seperti "administrator". Hal ini menyebabkan pengguna file dokumen harus menemukan cara untuk mengamankan file dokumen tanpa campur tangan administrator.

Pekerjaan-pekerjaan yang dahulunya manualpun kini mulai banyak yang terkomputerisasi mulai dari pekerjaan rumah dari sekolah hingga perusahaan bertaraf internasional dan semua pekerjaan-pekerjaan itu adalah sebuah informasi penting.

Kriptografi adalah salah satu cara untuk mencegah kebocoran data yang bersifat rahasia. Aplikasi kriptografi yang digunakan untuk enkripsi dan dekripsi dengan metode algoritma blowfish merupakan salah satu caranya.

Kriptografi dapat digunakan untuk mengamankan file dokumen. Oleh karena itu, penguna file dokumen membutuhkan bantuan untuk memenuhi kebutuhan keamanan akan file dokumen yang tersimpannya. Penerapan kriptografi pada Tesis ini akan difokuskan bagaimana kriptografi dapat mengamankan file dokumen yang tersimpan menjadi aman sampai dengan file dokumen dibuka oleh pihak yang berhak untuk membukanya. Secara umum ada dua jenis kriptografi, yaitu kriptografi klasik dan kriptografi modern.

Kriptografi klasik (simetrik) adalah suatu algoritma yang menggunakan satu kunci untuk mengamankan data. Dua teknik dasar yang biasa dilakukan adalah substitusi dan transposisi. Sedangkan kriptografi modern (asimetrik) adalah algoritma yang lebih kompleks dari pada algoritma klasik, hal ini disebabkan algoritma ini menggunakan komputer. Algoritma yang akan penulis gunakan adalah algoritma simetrik menggunakan blowfish.

Algoritma kriptografi yang akan digunakan adalah algoritma *Blowfish* alias "OpenPGP.Cipher.4" merupakan enkripsi yang termasuk dalam golongan *Symmetric Cryptosystem,* metode enkripsinya mirip dengan DES (Data Encription System-DES like Cipher). Dibuat untuk digunakan pada komputer yang mempunyai *microprosesor* besar (32-bit keatas dengan cache data yang besar).

Blowfish dikembangkan untuk memenuhi kriteria desain yang cepat dalam implementasinya dimana pada keadaan optimal dapat mencapai 26 *clock cycle perbyte,* kompak dimana dapat berjalan pada memori kurang dari 5 KB, sederhana dalam algoritmanya sehingga mudah diketahui kesalahannya, dan keamanan yang variabel dimana panjang kunci bervariasi (minimum 32 bit, maksimum 448 bit, *Multiple* 8 bit, default 128 bit).

Masalah keamanan merupakan salah satu aspek paling penting dalam dunia teknologi informasi, misalnya: keamanan data/*file* office Sekarang ini, sebagian besar dokumen-dokumen data menggunakan aplikasi Microsoft Office, Adobe Reader karena kemudahan dalam menggunakannya.

Di dalam data Microsoft Office dan Adobe Reader ada beberapa aplikasi yang dapat digunakan, yaitu Microsoft Word, Microsoft Excel, Microsoft Access, Microsoft PowerPoint dan Adobe Reader. Berbagai aplikasi dalam Microsoft Office dapat digunakan untuk mengolah kata dan angka sesuai kebutuhan pengguna. Keamanan data sangat diperlukan, maka setiap orang memerlukan suatu aplikasi yang dapat mengamankan dokumen data rahasia dan penting agar file dokumen tersebut hanya dapat di lihat dan di baca oleh orang tertentu saja.

Beberapa cara telah dikembangkan untuk menangani masalah keamanan ini, salah satu teknik untuk pengamanan data adalah dengan menggunakan algoritma penyandian data. Algoritma penyandian data saat ini telah semakin banyak jumlahnya, sejalan dengan berkembangnya ilmu yang mempelajari penyandian data tersebut. Ilmu ini biasa disebut

Kriptografi. Dalam kriptografi terdapat beberapa metode yang cukup penting dalam pengamanan data, untuk menjaga kerahasiaan data salah satunya adalah enkripsi (*encryption*).

Enkripsi adalah suatu proses yang dilakukan untuk mengubah pesan asli menjadi chipertext. Sedangkan suatu proses yang dilakukan untuk mengubah pesan tersembunyi menjadi pesan asli disebut dekripsi. Pesan biasa atau pesan asli disebut plaintexts edangkan pesan yang telah diubah atau disandikan supaya tidak mudah dibaca disebut dengan *chipertext*.

Untuk mengatasi masalah keamanan dokumen file ini, penulis meneliti melakukan pendekatan teknologi enkripsi data/file Microsoft office dan PDF menggunakan metode algoritma Blowfish. Enkripsi data/file merupakan teknologi untuk memastikan bahwa informasi yang mengalir pada suatu sesi tidak disadap atau diubah orang lain. Blowfish merupakan enkripsi yang termasuk dalam golongan Symmetric *Cryptosystem*. yaitu menggunakan kunci yang sama untuk enkripsi dan dekripsinya.

II. LANDASAN PEMIKIRAN

A. Keamanan Informasi

Keamanan Informasi menurut Sarno dan Iffano adalah suatu upaya untuk mengamankan *asset* informasi terhadap ancaman yang mungkin timbul. Sehingga keamanan informasi secara tidak langsung dapat menjamin kontinuitas bisnis, mengurangi resiko-resiko yang terjadi, mengoptimalkan pengembalian investasi (*return on investment*). Semakin banyak informasi perusahaan yang disimpan, dikelola dan di-sharing-kan maka semakin besar pula resiko terjadi kerusakan, kehilangan atau terekspos nya data ke pihak eksternal yang tidak dinginkan [Prabowo 2013].

B. Aspek-aspek Keamanan Informasi

Dari sekian banyak cara yang tersebar di internet dan dibuku-buku tentang keamanan informasi, sebenarnya ada 8 aspek yang jika semua dapat dipenuhi dan dijalankan dengan baik maka keamanan informasi dapat terjaga dengan baik pula.

Aspek-aspek tersebut adalah:

1. *Authentication*: agar penerima informasi dapat memastikan keaslian pesan tersebut datang dari orang yang dimintai informasi. Artinya informasi yang diterima benar dari orang yang dikehendaki.
2. *Integrity*: keaslian pesan yang dikirim melalui sebuah jaringan dan dapat dipastikan dapat pastikan pesan yang dikirim tidak dapat dimodifikasi oleh orang yang tidak berhak dalam perjalanan pengiriman informasi tersebut.
3. *NonRepudiation*: merupakan hal yang bersangkutan dengan sipengirim informasi. Si pengirim tidak dapat mengelak dialah yang mengirim informasi tersebut,
4. *Authority*: Informasi yang berada pada sistem jaringan tidak dapat dimodifikasi oleh orang yang tidak berhak.
5. *Confidentiality*: merupakan suatu usaha untuk menjaga informasi dari orang yang tidak berhak mengakses. Aspek ini biasanya berhubungan dengan informasi yang diberikan kepada pihak lain.
6. *Privacy*: lebih kearah data-data yang sifatnya private.
7. *Availability*: ketersediaan hubungan dengan ketersediaan informasi ketika dibutuhkan. Sistem informasi yang diserang atau dijebol dapat menghambat atau meniadakan akses ke informasi.
8. *Access Control*: Aspek ini berhubungan dengan cara pengaturan akses kepada informasi. Aspek ini berhubungan dengan aspek *Authentication* dan *Privacy. Access Control* seringkali dilakukan dengan kombinasi user id dan password atau dengan mekanisme lainnya. [Irham 2013]

C. Kriptografi

Kriptografi (Cryptography) berasal dari bahasa yunani, *Cyptos* artinya *secret* atau rahasia sedangkan *graphein* berarti: *writing* atau tulisan. Sehingga kriptografi berarti *secret writing* atau tulisan rahasia. Menurut Bruce Scheiner (1996): Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan sedangkan menurut Menezes (1996): kriptografi adalah limu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, otentikasi dan anti penyangkalan. Kriptografi dapat diartikan sebagai ilmu untuk menjaga kerahasiaan informasi dengan metode dan teknik matematika yang mencakup *confidentiality, integrity, authentication* dan *non-repudiation*.

D. Terminologi dalam Kriptografi

Dalam kriptografi akan sering dijumpai beberapa istilah atau terminology sebagai berikut:

1. Pesan, *plaintext* dan *Ciphertext*.

Pesan (message) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya.

Nama lain untuk pesan adalah *plaintext* atau teks jelas (*cleartext).* Agar pesan tidak dimengerti oleh pihak lain yang tidak berkepentingan, maka pesan perlu disandikan menjadi bentuk lain yang tidak dapat dipahami. Bentuk pesan tersandi disebut *ciphertext* atau *cryptogram.*

Cipherteks harus dapat ditransformasi kembali menjadi plainteks. Sebagai contoh plainteks, uang disimpan di balik buku X, maka cipherteksnya adalah j&kloP#d$gkh*7h^”tn%6^klp..t@.

2. Pengirim dan penerima

Komunikasi data melibatkan pertukaran pesan antara dua entitas. Pengirim (*sender*) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima (*receiver*) adalah entitas yang menerima pesan. Entitas dapat berupa orang, mesin, kartu kredit dan sebagainya.

3. Enkripsi dan dekripsi

Proses menyandikan *plainteks* menjadi chiperteks disebut enkripsi (*encryption*). Sedangkan proses mengembalikan ciphertext disebut enkripsi (encryption), sedangkan proses mengembalikan ciphertext menjadi *plaintext* disebut dekripsi (*decryption).*

4. Algoritma Kriptografi dan Kunci

Algoritma kriptografi disebut juga *chiper* yaitu aturan untuk *enchipering* dan *dechipering*, atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Kriptografi modern mengatasi masalah keamanan algoritma kriptografi dengan penggunaan kunci. Kunci (*key*) adalah parameter yang digunakan untuk transformasi *enchipering* dan *dechipering*. Kunci biasanya berupa *string* atau deretan bilangan.

5. Sistem kriptografi

Kriptografi membentuk sebuah sistem yang dinamakan sistem kriptografi. Menurut Schneier dan Munir, system kriptografi (*cryptosystem*) adalah kumpulan yang terdiri atas algoritma kriptografi, semua plainteks dan chiperteks yang mungkin serta kunci.

6. Kriptanalisis dan kriptografi

Kriptanalisis (*cryptanalysis*) adalah ilmu dan seni untuk memecahkan chiperteks menjadi plainteks tanpa mengetahui kunci yang diberikan. Pelakunya disebut kriptanalis. Kriptologi (*cryptology*) adalah studi mengenai kriptografi dan kriptanalisis.

E. Konsep Matematis Algoritma Kriptografi

Konsep matematis yang mendasari algoritma kriptografi adalah relasi antara dua buah himpunan yaitu himpunan yang berisi elemen-elemen *plainteks* dan himpunan yang berisi *chiperteks*. Enkripsi dan dekripsi merupakan fungsi yang memetakan elemen-elemen antara kedua himpunan tersebut.

Misalkan P menyatakan *plainteks* dan C menyatakan *chiperteks*, maka fungsi enkripsi E memetakan P ke C,

$$E(P) = C$$

Dan fungsi dekripsi D memetakan C ke P,

$$D(C) = P$$

Karena proses enkripsi kemudian dekripsi mengembalikan pesan ke pesan asal maka persamaan berikut harus benar,

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

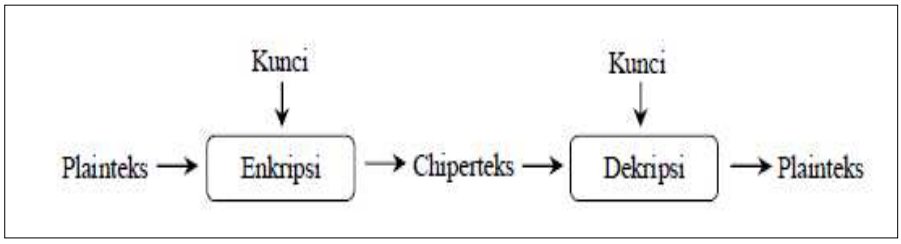
Dengan menggunakan kunci K, maka fungsi enkripsi dan dekripsi menjadi,

$$EK(P) = C$$
$$DK(C) = P$$

dan kedua fungsi ini memenuhi:

$$DK(EK(P)) = P$$

Gambar 1. memperlihatkan skema enkripsi dan dekripsi dengan menggunakan kunci.



Gambar 1. Skema enkripsi dan dekripsi dengan kunci

F. Kriptografi Sistem Simetrik

Sistem simetris adalah system kriptografi yang menggunakan kunci yang sama untuk proses enkripsi dan dekripsi. Sistem ini sering juga disebut dengan algoritma kunci tunggal atau algoritma satu kunci. Bila E adalah fungsi enkripsi (encryption), K adalah kunci rahasia (key), sedangkan M adalah pesan orisinil yang akan dikirimkan (message) dan C adalah pesan sandinya (cipher), maka system simetris dapat diformulasikan sebagai berikut:

$$E_k(M)=C \text{ dan } D_k(C)=M$$

Dalam aplikasinya antara pengirim dan penerima harus ada persetujuan atau sinkronisasi kunci agar saling berkomunikasi. Jadi, keamanan algoritma sistem simetris terletak pada kunci. Siapapun yang memperoleh kunci, akan dapat membuka pesan yang dikomunikasikan. Karena itu selama proses komunikasi bersifat rahasia, maka kunci harus tetap dirahasiakan. Sistem Kripto Simetris yang menyediakan keamanan secara praktis terbagi dua kategori yaitu *Stream Cipher dan Blok Cipher.*

a) Stream Cipher

Bruce Schneier mendefinisikan *stream cipher* sebagai system kriptografi yang mengganti teks terang menjadi teks sandi satu bit per satuan waktu, keamanan sistem ini tergantung pada kondisi internal dari pembangkit kuncinya. Beberapa contoh algoritma penyandian yang menggunakan system Stream cipher antara lain RC4, SEAL, A5, PIKE, Camellia dan lain-lain.

Dari definisi diatas, maka struktur dari *stream cipher* meliputi tiga komponen utama, yaitu :

1. Kunci Input (*Seed*)

Kunci Input (Seed) dapat dibangkitkan secara otomatis atau manual. Seed biasanya merupakan rangkaian bilangan acak dan dibangkitkan oleh suatu *Random Number Generator* (RNG).

2. SKG (*Stream Key Generator)*

SKG merupakan algoritma tertentu yang harus dapat menjamin rangkaian kunci stream yang dibangkitkan merupakan barisan yang pseudorandom. SKG biasanya disebut juga *pseudo Random Number Generator.*

3. Teks Terang

Teks terang akan dienkripsikan menjadi teks sandi. Umumnya output rangkaian kunci dari generator dikombinasikan dengan rangkaian teks terang perbyte per satuan waktu dengan menggunakan operasi *bitwise exclusive-OR (XOR).*

b) Block Cipher

1. Definisi dan gambaran Umum

Block Cipher merupakan salah satu model penyandian dimana teks terang yang akan disandi dibagi kedalam blok-blok yang telah ditentukan sehingga menghasilkan blok-blok teks sandi. Menurut Menezes dan kawan-kawan (1996) *Blok cipher* adalah suatu fungsi yang memetakan n-bit blok teks terang ke n-bit teks sandi, dengan n adalah panjang blok. *Blok cipher* merupakan gambaran dari system sandi substitusi sederhana yang memiliki periode panjang. Berdasarkan definisi tersebut maka *block cipher* memiliki karakteristik sebagai berikut :

a) *Plaintext* dibagi menjadi blok-blok bit dengan panjang sama, misalnya 64 bit.
b) Panjang kunci ekripsi = panjang blok

c) Enkripsi dilakukan terhadap blok bit *plaintext* menggunakan bit-bit kunci.
d) Algoritma enkripsi menghasilkan blok *ciphertext* yang panjangnya = blok plainteks.

Beberapa contoh algoritma *block cipher* yang telah berkembang saat ini adalah DES (Data Encryption Standard), 3DES, FEAL, IDEA, RC-5, AES (*Advanced Encryption Standard), Lucifer, Blowfish,* LOKI dan lain-lain. Untuk penulisan tesis ini Algoritma *blok cipher* yang digunakan untuk enkripsi adalah Algoritma Blowfish 64 bit.

G. Algoritma Blowfish

Blowfish adalah algoritma kriptografi simetris *block cipher* yang dibuat oleh *Bruce Schneier. Blowfish* menyandi teks terang dalam blok-blok berukuran 64-bit menjadi blok-blok teks sandi dengan ukuran sama yaitu 64-bit. Algoritma *Blowfish* terdiri atas dua bagian yaitu pembangkitan sub-kunci (*key-expansion)* dan enkripsi data. Enkripsi data terdiri dari iterasi fungsi sederhana (*feistel Network)* sebanyak 16 kali putaran. Semua operasi adalah penambahan (*addition*) dan XOR pada variable 32-bit. Key expansion mengubah kunci yang dapat mencapai 448 bit menjadi beberapa array subkunci (subkey) dengan total 4168 byte.

Blowfish dikembangkan untuk memenuhi kriteria disain sebagai berikut:

1. Cepat, pada implementasi yang optimal *blowfish* dapat mencapai kecepatan 26 *clock cycle per byte.*
2. Kompak, *blowfish* dapat berjalan pada memori kurang dari 5 KB.
3. Sederhana, *blowfish* hanya menggunakan operasi yang simpel yaitu penambahan (*addition*), XOR, dan penelusuran table (*table lookup)* pada operand 32 bit. Desainnya mudah untuk dianalisis yang membuatnya resisten terhadap kesalahan implementasi.

   Keamanan yang variable, panjang kunci *blowfish* dapat bervariasi dan dapat mencapai 448 bit (56 byte).

H. Enkripsi Data

Terdiri dari iterasi fungsi sederhana (*Feistel Network*) sebanyak 16 kali putaran. Setiap putaran terdiri dari permutasi kunci-dependent dan substitusi- kunci dan data-dependent. Semua operasi adalah penambahan (*addition)* dan XOR pada variabel 32-bit. Semua Operasi adalah tambahan lainnya hanyalah empat penelusuran table (*table lookup)* array berindeks untuk setiap putaran.

Langkah-langkah adalah sebagai berikut:

1. Bagi X menjadi dua bagian yang masing-masing terdiri dari 32-bit: X_L, X_R
2. Lakukan langkah berikut

   For i = 1 to 6

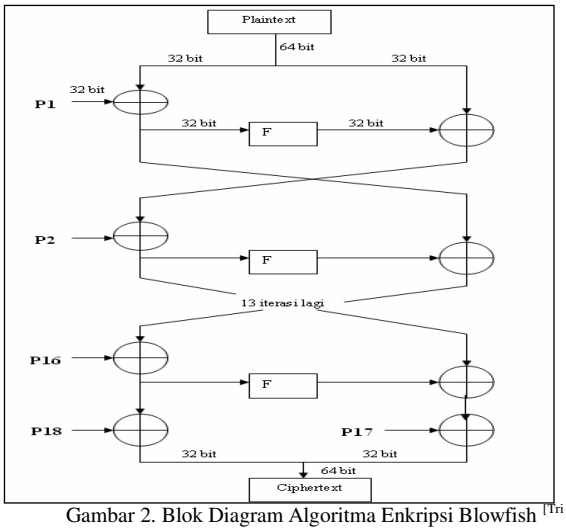
   $X_L = X_L \oplus P_i$

   $X_R = F(X_L) \oplus X_R$

   Tukar X_L dan X_R
3. Setelah iterasi ke=16, tukar X_L dan X_R lagi untuk melakukan membatalkan pertukaran terakhir.
4. Lalu lakukan

   $X_R = X_R \oplus P_{17}$

   $X_L = X_L \oplus P_{18}$
5. Terakhir, gabungkan kembali X_L dan X_R untuk mendapatkan cipherteks.



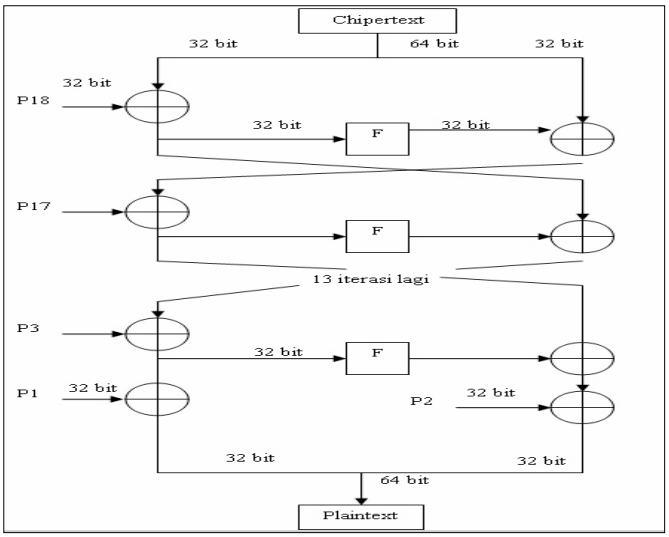
Gambar 2. Blok Diagram Algoritma Enkripsi Blowfish [Tri 2008]

I. Dekripsi Data

Proses dekripsi Algoritma *Blowfish* sama persis dengan proses enkripsi Blowfish. Perbedaan terletak pada urutan penggunaan sub kunci P1, P2,..,P18. Pada proses dekripsi Blowfish urutan penggunaan sub kunci P1, P2,.....,P18 dibalik menjadi P18 P17,.....,P1.

Dekripsi atau *deciphering*. merupakan proses mengembalikan *cipherteks* menjadi *plainteks* semula.

for i = 1 to 16 do
XRi = XLi-1 ⊕ P19-i;
XLi = F[XRi] ⊕ XRi-1;
XL17 = XR16 ⊕ P1;
XR17 = XL16 ⊕ P2;

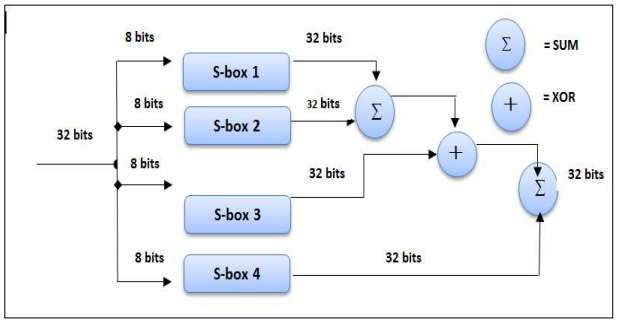


Gambar 3. Blok Diagram Algoritma Dekripsi *Blowfish* [Tri 2008]

J. Pembangkitan Subkunci

Subkunci dihitung menggunakan Algoritma *Blowfiish,* dengan langkah-langkah sebagai berikut:

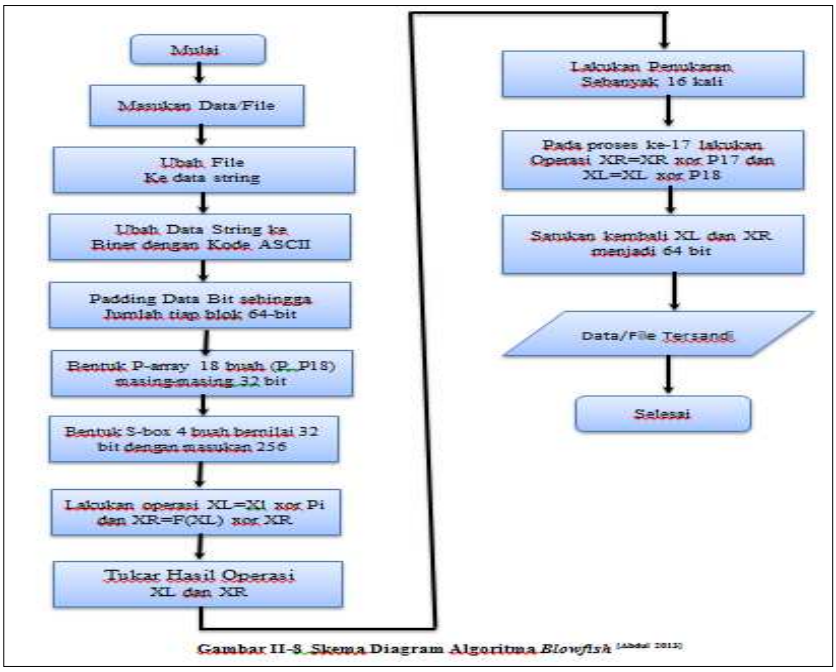
1. Inisialisasi P-array dan kemudian empat S-box secara berurutan dengan string tetap. String ini terdiri dari digit. *Hexadecimal* dari pi. Dimana P-array terdiri dari 18 subkunci dengan ukuran 32 bit:
   P1, P2, …, P18
   P1, P2, …, P18
2. XOR P1 dengan 32 bit pertama kunci, XOR P2 dengan 32 bit kedua dari kunci dan seterusnya untuk setiap bit dari kunci (sampai P18).
   Ulangi terhadap bit kunci sampai seluruh P-array di XOR dengan bit kunci.
3. Enkripsikan semua string nol dengan algoritma Blowfish menggunakan subkunci seperti yang dijelaskan pada langkah 1 dan langkah 2.
4. Gantikan P1 dan P2 dengan keluaran dari langkah 3.
5. Enkripsikan keluaran langkah 3 dengan algoritma Blowfish dengan subkunci yang sudah termodifikasi.
6. Gantikan P3 dan P4 dengan keluaran dari langkah 5.
7. Teruskan proses tersebut, gantikan seluruh elemen dari P-array, dan kemudian seluruh keempat S-Box berurutan, dengan keluaran yang berubah secara kontinyu dari Algoritma *Blowfish*.
   Secara keseluruhan diperlukan 521 iterasi untuk membangkitkan semua subkunci yang dibutuhkan



Gambar 4. Skema Fungsi F Algoritma Blowfish[Tri 2008]

K. Skema Diagram Algoritma Blowfish

Adapun Skema diagram proses enkripsi algoritma *blowfish* disajikan pada Gambar 5.



Gambar II-8 Skema Diagram Algoritma *Blowfish* [Abdul 2012]

Gambar 5. Skema Diagram Algoritma Blowfish

L. Object Oriented Programming (OOP) Pemrograman JAVA

JAVA adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam (*Cellular*). Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di *Sun Microsystems* saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal.

Aplikasi-aplikasi berbasis *java* umumnya dikompilasi ke dalam p-code (bytecode) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum /non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa *platform* system operasi yang berbeda, *java* dikenal pula dengan slogannya, "*Tulis sekali, jalankan di mana pun*". Saat ini java merupakan bahasa pemrograman yang paling popular digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web [Hendra 2011].

M. Pemrograman Java Netbeans.

Netbeans merupakan sebuah aplikasi Integrated Development Environment (IDE) yang berbasiskan Java dari Sun Microsystems yang berjalan di atas swing. Swing merupakan sebuah teknologi Java untuk pengembangan aplikasi desktop yang dapat berjalan pada berbagai macam platform seperti windows, linux, Mac OS X dan Solaris. Sebuah IDE merupakan lingkup pemrograman yang di integrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan Graphic User Interface (GUI), suatu kode editor atau text, suatu kompiler dan suatu *debugger*.

Netbeans juga digunakan oleh sang programmer untuk menulis, meng-*compile*, mencari kesalahan dan menyebarkan program netbeans yang ditulis dalam bahasa pemrograman java namun selain itu dapat juga mendukung bahasa pemrograman lainnya dan program ini pun bebas untuk digunakan dan untuk membuat professional desktop, enterprise, web, and mobile applications dengan Java language, C/C++, dan bahkan dynamic languages seperti PHP, JavaScript, Groovy, dan Ruby.

NetBeans merupakan sebuah proyek kode terbuka yang sukses dengan pengguna yang sangat luas, komunitas yang terus tumbuh, dan memiliki hampir 100 mitra (dan terus bertambah!). Sun Microsystems mendirikan proyek kode terbuka NetBeans pada bulan Juni 2000 dan terus menjadi sponsor utama. Dan saat ini pun netbeans memiliki 2 produk yaitu Platform Netbeans dan Netbeans IDE. Platform Netbeans merupakan framework yang dapat digunakan kembali (reusable) untuk menyederhanakan pengembangan aplikasi desktop dan Platform NetBeans juga menawarkan layanan-layanan yang umum bagi aplikasi desktop, mengijinkan pengembang untuk fokus ke logika yang spesifik terhadap aplikasi.

N. Fitur-fitur dari Platform Netbeans.

Fitur-fitur dari Platform Netbeans antara lain:

- Manajemen antarmuka (misal: menu & toolbar)

- Manajemen pengaturan pengguna
- Manajemen penyimpanan (menyimpan dan membuka berbagai macam data)
- Manajemen jendela
- Wizard framework (mendukung dialog langkah demi langkah)

Netbeans IDE merupakan sebuah IDE open source yang ditulis sepenuhnya dengan bahasa pemrograman java menggunakan platform netbeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, web, EJB, dan aplikasi mobile). Fitur lainnya adalah sistem proyek berbasis Ant, kontrol versi, dan refactoring.

Versi terbaru saat ini adalah NetBeans IDE 5.5.1 yang dirilis Mei 2007 mengembangkan fitur-fitur Java EE yang sudah ada (termasuk Java Persistence support, EJB-3 dan JAX-WS). Sementara paket tambahannya, NetBeans Enterprise Pack mendukung pengembangan aplikasi perusahaan Java EE 5, meliputi alat desain visual SOA, skema XML, web service dan pemodelan UML.

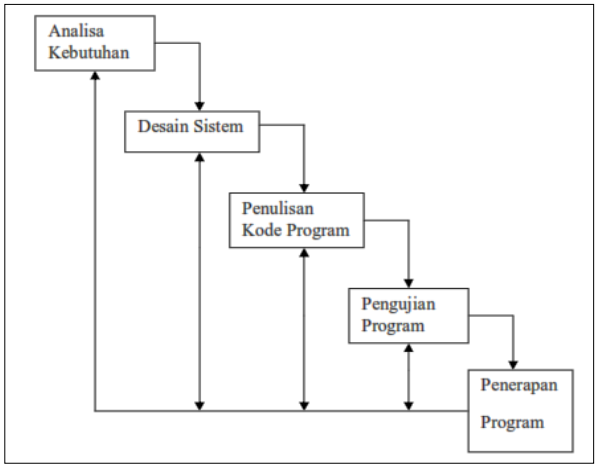
NetBeans C/C++ Pack mendukung proyek C/C++.Modularitas: Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk bahasa pemrograman Java, editing, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan Java dalam sekali download, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul-modul juga mengijinkan NetBeans untuk bisa dikembangkan. Fitur-fitur baru, seperti dukungan untuk bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan. Sebagai contoh, Sun Studio, Sun Java Studio Enterprise, dan Sun Java Studio Creator dari Sun Microsystem semuanya berbasis NetBeans IDE.

O. Fitur fitur yang terdapat dalam netbeans antara lain:

1. Smart Code Completion: untuk mengusulkan nama variabel dari suatu tipe, melengkapi keyword dan mengusulkan tipe parameter dari sebuah method.
2. Bookmarking: fitur yang digunakan untuk menandai baris yang suatu saat hendak kita modifikasi.
3. Go to commands: fitur yang digunakan untuk jump ke deklarasi variabel, source code atau file yang ada pada project yang sama.
4. Code generator: jika kita menggunakan fitur ini kita dapat meng-generate constructor, setter and getter method dan yang lainnya.
5. Error stripe: fitur yang akan menandai baris yang eror dengan memberi highlight merah.[Hendra 2011]

P. Model Pengembangan Sistem dengan Metode Waterfall

Metode *Waterfall* adalah sebuah metode pengembangan software yang bersifat sekuensial dan terdiri dari 5 tahap yang saling terkait dan mempengaruhi seperti terlihat pada Gambar 6.



Gambar 6. Tahapan Pengembangan Sistem dengan Metode Waterfall[Pressman 1997]

Keterkaitan dan pengaruh antar tahap ini ada karena *output* sebuah tahap dalam *Waterfall Model* merupakan input bagi tahap berikutnya, dengan demikian ketidaksempurnaan hasil pelaksanaan tahap sebelumnya adalah awal ketidaksempurnaan tahap berikutnya. Memperhatikan karakteristik ini, sangat penting bagi tim pengembang dan perusahaan untuk secara bersama-sama melakukan analisa kebutuhan dan desain sistem sesempurna mungkin sebelum masuk ke dalam tahap penulisan kode program.

Q. Flow chart

Flow chart atau diagram alir merupakan sebuah diagram dengan symbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing langkah tersebut menggunakan tanda panah. Diagram ini bisa memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut.

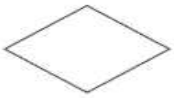
Flow chart digunakan dalam merancang dan mendokumentasikan proses yang kompleks atau program. Seperti jenis lain diagram, mereka membantu memvisualisasikan apa yang terjadi dan dengan demikian membantu pengunjung untuk memahami proses, dan mungkin juga menemukan kelemahan, kemacetan, dan ketidakjelasan lain di dalamnya. Ada berbagai jenis diagram alur yang masing-masing memiliki *repertoire* kotak sendiri dan ketentuan notasinya. Dua jenis yang paling umum dari kotak di *flow chart* adalah :

1. Langkah pengolahan, biasanya disebut aktivitas, dan dilambangkan sebagai persegi panjang.
2. Keputusan, biasanya dilambangkan sebagai belah ketupat.

Sebuah *flow chart* digambarkan sebagai lintas-fungsional saat halaman dibagi menjadi *swimlanes* yang berbeda yang menggambarkan control dari unit organisasi yang berbeda. Sebuah simbol muncul dalam jalur khusus berada dalam kontrol ari unit organisasi. Teknik ini memungkinkan penulis untuk mencari tanggung jawab untuk melakukan tindakan atau membuat keputusan dengan benar, menunjukkan tanggungjawab masing-masing unit organisasi untuk bagian yang berbeda dari sebuah proses tunggal.

Flow chart menggambarkan aspek-aspek tertentu dari proses dan mereka biasanya dilengkapi dengan jenis lain dari diagram. Misalnya, Kaoru Ishikawa mendefinisikan *flow chart* sebagai salah satu dari tujuh alat dasar kontrol kualitas, setelah histogram, *pareto chart, check sheet, control chart, cause-and-effect diagram, and the scatter diagram*.

TABEL 1.
SIMBOL FLOW CHART [TOSIN 1994]

GAMBAR	FUNGSI	KETERANGAN
	Proses atau Langkah	Menyatakan kegiatan yang akan ditampilkan dalam diagram alir
	Titik Keputusan	Proses/langkah dimana adanya keputusan atau adanya kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
	Masukan atau Keluaran Data	Sigunakan untuk mewakili data masuk, atau data keluar.
	Terminasi	Menunjukan awal atau akhir sebuah proses
	Garis Alir	Menunjukan arah aliran proses atau algoritma
	Kontrol atau Inspeksi	Menunjukkan proses / langkah dimana ada inspeksi atau pengontrolan

R. TINJAUAN STUDI

Dalam penerapannya sering kali metode algortima Blowfish ini menjadi tidak optimal. Karena strategi implementasi yang tidak tepat. Algoritma Blowfish akan lebih optimal jika digunakan untuk aplikasi yang tidak sering berganti kunci, seperti jaringan komunikasi atau enkripsi file otomatis. Selain itu, karena algoritma ini membutuhkan memori yang besar, maka algoritma ini tidak dapat diterapkan untuk aplikasi yang memiliki memori kecil seperti smart card. Panjang kunci yang digunakan, juga mempengaruhi keamanan penerapan algoritma ini.

Abdul Aziz melakukan penelitian dengan menerapkan Steganografi Teknik LSB (Least Significant Bit) dengan Algoritma Kriptografi Blowfish dan fungsi Hash SHA-1 dengan tujuan menghasilkan rancangan aplikasi pengamanan data dengan menggunakan gabungan teknik tersebut yang berbasis web, dimana pesan rahasia dienkripsi terlebih dahulu menggunakan algoritma Blowfish, lalu setelah itu disisipkan pada media dengan menggunakan metode LSB. Sedangkan untuk mengamankan password menggunakan fungsi hash SHA-1. Output yang dihasilkan oleh aplikasi adalah sebuah file gambar yang di dalamnya telah mengandung pesan/data rahasia atau disebut dengan *stegoimage*. [Aziz 2013]

Tri Andriyanto melakukan penelitian dengan tujuan menerapkan Algoritma kriptografi simetris dengan kategori *block cipher* adalah Algoritma *International Data Encryption Algorithm* (IDEA) dan Algoritma *Blowfish*. Kedua algoritma ini beroperasi dalam bentuk blok bit, dengan ukuran blok sebesar 64 bit. Kedua algoritma ini juga dikenal sangat tangguh dalam mengamankan informasi. Studi dan perbandingan antara algoritma IDEA dan algoritma *Blowfish* dilakukan untuk membandingkan kinerja algoritma IDEA dan *Blowfish* dalam hal kecepatan proses dan penggunaan memori pada saat proses enkripsi dan dekripsi suatu file.[Andriyanto 2008]

Ana Wahyuni melakukan penelitian dengan menerapkan Aplikasi Kriptografi untuk mengamankan E-Dokumen dengan Metode Hybrid: Biometrik Tandatangan dan DSA (Digital Signature Algorithm) dengan tujuan menghasilkan rancangan aplikasi kriptografi dengan metode *hybrid :* Biometrik tandatangan dan DSA (*Digital Signature Algorithm*) sebagai salah satu solusi pada masalah manajemen kunci dan memenuhi kebutuhan ketidaktunggalan *signer*. Biometrik yang digunakan adalah tandatangan *offline* satu atau lebih pengguna

menghasilkan satu atau lebih tandatangan *digital* untuk satu e-dokumen. Selanjutnya e-dokumen, tandatangan *digital* dan kunci publik ditransmisikan lewat internet via *e-mail* pada pihak *verifier*. Kemudian pihak *verifier* memverifikasi apakah hasilnya valid artinya e-dokumen tersebut masih otentik/ utuh dan pengirim adalah *signer* sebenarnya dari e-dokumen tersebut. Sebaliknya jika hasilnya tidak valid artinya e-dokumen tersebut sudah tidak otentik/ utuh dan atau pengirim bukanlah *signer* sebenarnya dari e-dokumen tersebut.[Wahyuni 2011]

Penelitian mengenai sistem pengamanan Short Message Service (SMS) telah dilakukan oleh Ashish Ranjan, et.al[Ranjan 2012]. Sistem yang dikembangkan berupa pengamanan SMS untuk kebutuhan M-Commercemenggunakan program J2ME. Sistem pengamanan SMS tersebut melibatkan server yaitu bank sebagai pihak yang memvalidasi pembayaran oleh pembeli menggunakan ponselnya pada mekanisme M-Commerce. Proses validasi yang dilakukan server bank adalah proses enkripsi, dekripsi dan verifikasi pesan SMS yang dikirim oleh pembeli. Algoritma kriptografi yang digunakan dalam sistem tersebut yaitu algoritma kriptografi simetrik TEA dan fungsi hash MD5. Hasil penelitian tersebut menyebutkan bahwa proses enkripsi, dekripsi dan verifikasi e-mail berhasil diterapkan. Namun berkaitan dengan pemenuhan aspek keamanan informasi, dalam penelitian tersebut hanya memenuhi aspek confidentiality dan data integrity.

III. DESAIN PENELITIAN

A. Jenis Penelitian

Penelitian tesis ini merancang aplikasi kriptografi untuk enkripsi dan dekripsi data atau file *office* menggunakan metode *blowfish* dengan Bahasa Pemrograman JAVA yang dilakukan merupakan jenis penelitian terapan. Hasil penelitian bertujuan untuk memberikan solusi atas permasalahan secara praktis dan manfaatnya dapat dirasakan secara langsung.

B. Metode Pengumpulan Data

Metode-metode pengumpulan data yang digunakan dalam penelitian ini adalah:

1. Metode Observasi. Yaitu melalui pengamatan langsung terhadap obyek penelitian, dimana proses observasi dilakukan dengan mempelajari software-software pengamanan data yang menggunakan Algoritma Steganografi Teknik LSB (Least Significant Bit) dengan Algoritma Kriptografi Blowfish dan fungsi Hash SHA-1 denga menggunakan pemograman PHP.
2. Metode Studi Pustaka, dimana data yang diperoleh dengan mempelajari, meneliti, dan membaca buku, jurnal, tesis dan artikel-artikel sebagai acuan pembahasan dalam membangun sistem yang dikembangkan berhubungan dengan Algoritma *Blowfish*.

C. Teknik Perancangan, Implementasi, dan Pengujian

Kriptografi simetrik dapat dikatakan yang tidak dipatenkan dan cukup kuat saat ini dalam pengamanan data. Dimana kriptografi simetrik ini dapat digunakan dengan tujuan otentikasi yang berarti agar penerima data/ file informasi dapat memastikan keaslian data tersebut datang dari orang yang dimintai data / file . Artinya data informasi yang diterima benar dari orang yang dikehendaki. Ketika seseorang mengirimkan sebuah data / file yang terenkripsi dengan memanfatkan kunci public dimana kita yakin dengan kunci pribadi yang kita miliki saja maka data tersebut dapat didekripsi. Dapat dikatakan bahwa telah terjadi proses otentikasi kepada pengirim yaitu kunci rahasia yang dikirimkan oleh si pengirim menjadi identitas dari pengirim pribadi.

Pada proses perancangan, teknik perancangan yang dilakukan adalah :

1. Perencanaan Sistem, pada tahapan ini penulis membuat perencanaan yang berhubungan dengan pembahasan sistem ini. Karena sistem merupakan bagian dari sebuah sistem yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa *subset* dari kebutuhan ke *software* tersebut.
2. Desain system secara umum, pada tahapan ini penulis membuat desain *workflow* terhadap sistem dan pemrograman yang diperlukan untuk pengembangan sistem informasi. Proses desain menterjemahkan syarat/kebutuhan ke dalam sebuah representasi software yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.

D. Implementasi

Algoritma *Blowfish* digunakan sebagai salah satu metode kriptografi simetrik, yang meliputi mekanisme enkripsi data / file menggunakan kunci pribadi untuk enkripsi dan dekripsi data / *file*. Dengan pemanfaatan bahasa pemograman JAVA untuk pembuatan aplikasinya.

E. Pengujian Sistem

Pengujian sistem secara terinci (*testing*), pada tahapan ini penulis melakukan proses pengujian terhadap sistem aplikasi yang telah dibuat. Proses pengujian dilakukan pada :

1. Logika internal *software*, dan memastikan seluruh pernyataan-pernyataan dalam kode program sudah di uji, dimana tidak ada lagi kesalahan dan memastikan output akan memberikan hasil yang sesuai dengan yang dibutuhkan.

2. Pengujian terhadap beberapa tipe data / *file* yang akan dienkripsi dan didekripsi seperti *doc, docx, xls, xlsx, Powerpoint dan pdf.*
3. Pengujian terhadap keefektifan dari *file-file* yang diuji.
4. Pengujian dengan menggunakan kunci yang sama saat mengenkripsi.

F. Perancangan Aplikasi

Aplikasi pengamanan pada tahap perancangan sistem menggunakan Flowchart Diagram Model. Diagram Flowchart adalah sebuah model *process planning* dan structure prediction grafik/gambar untuk memvisualisasi, menspesifikasikan dan membangun sebuah sistem perangkat lunak berorientasi objek. Implementasi sistem dilakukan dengan menggunakan perangkat keras dan perangkat lunak dengan spesifikasi yang terdapat dalam obyek penelitian.

Perancangan dan Desain Sistem secara umum membuat desain *workflow* terhadap system dan pemrograman yang diperlukan untuk mengembangkan sistem Informasi. Proses desain menterjemahkan syarat/kebutuhan ke dalam sebuah *representasi software* yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode.

G. Pengujian Sistem

Pengujian sistem secara terinci (*testing)*, pada tahapan ini penulis melakukan proses pengujian terhadap sistem aplikasi yang telah dibuat. Proses pengujian dilakukan pada :

1. Logika internal *software*, dan memastikan seluruh pernyataan-pernyataan dalam kode program sudah di uji, dimana tidak ada lagi kesalahan dan memastikan output akan memberikan hasil yang sesuai dengan yang dibutuhkan.
2. Pengujian terhadap beberapa tipe file yang akan dienkripsi dan dekripsi seperti doc,docx, xls, xlsx, Powerpoint dan pdf.

H. Pengujian dan Analisis

Proses pengujian dan analisis dilakukan untuk mengidentifikasi apakah sistem yang dikembangkan sesuai dengan analisis sistem yang telah dibuat. Pengujian sistem yang dilakukan terdiri dari pengujian *black box*, pengujian performa dan pengujian keamanan. Pengujian *black box* dilakukan dengan menjalankan atau mengeksekusi unit atau modul, kemudian diamati apakah hasil dari unit tersebut sesuai dengan proses yang dikehendaki atau tidak. Pengujian performa dilakukan untuk mengukur dan mengetahui waktu proses enkripsi pada saat akan mengirimkan *e-mail* terenkripsi. Pada pengujian performa jumlah sampel data yang diambil berdasarkan *random sampling* yaitu sebanyak 50 sampel *e-mail* yang dikirim dengan kapasitas

3. yang sama untuk mengetahui waktu rata-rata proses enkripsi *e-mail*. Pengujian keamanan dilakukan untuk menguji apakah sistem yang dikembangkan memenuhi aspek keamanan informasi yang terdiri dari *confidentiality*, *data integrity*, *authentication* dan *non-repudiation.* Dari hasil seluruh pengujian tersebut selanjutnya akan dianalisis berdasarkan hipotesis yang telah diformulasikan untuk ditarik kesimpulan.

IV. PEMBAHASAN HASIL PENELITIAN

A. Analisa Masalah

Perkembangan teknologi informasi khususnya menggunakan teknologi komputer mengalami perkembangan pesat dan membawa dampak terhadap perubahan berbagai aspek kehidupan dengan semakin pesatnya perkembangan teknologi informasi menggunakan computer, semakin bertambah pula masalah yang mengiringi perkembangan teknologi tersebut. Salah satunya adalah masalah keamanan data.

Masalah keamanan data merupakan faktor yang harus diperhatikan dalam perkembangan teknologi informasi selanjutnya. Dimana jika membicarakan tentang informasi tentunya berkaitan pula dengan isi yang terkandung dalam informasi tersebut. Informasi yang akan disampaikan/dikirimkannya, sehingga pengiriman informasi dapat dipertanggungjawabkan dari segi kerahasiaan file / informasi, keaslian data file / informasinya, keutuhan dari data/informasi yang dikirimkan, dan juga pengirim data / informasi tidak dapat mengelak bahwa dialah yang mengirim informasi tersebut.

Pengamanan data/informasi dengan menggunakan metode enkripsi merupakan cara yang biasa digunakan dalam hal pengamanan data/informasi yang akan dikirimkan, agar data/informasi yang dikirimkan dapat terjaga kerahasiaannya. Idenya adalah merubah data/informasi yang dikirimkan menjadi data/informasi yang akan dikirimkan, agar data/informasi yang dikirimkan menjadi data/informasi yang tidak dapat dimengerti dimana untuk mengubah kebentuk yang dapat dimengerti menggunakan algoritma yang dapat mengkodekan data yang diinginkan.

Dalam aspek-aspek pengamanan data/informasi tidak hanya dibutuhkan kerahasiaan dalam pengiriman data/informasi. Namun juga harus mengutamakan komponen-komponen sebagai berikut :

1. *Authentication*. Penerima data/informasi ataupun tugas dapat memastikan keaslian pengirimannya. Sehingga penyerang tidak dapat berpura-pura sebagai orang lain.
2. *Integrity.* Penerima data/informasi ataupun tugas harus dapat memeriksa apakah data/informasi telah di modifikasi di tengah jalan atau tidak seorang penyusup seharusnya tidak dapat memasukkan tambahan, pengurangan ataupun merubah ke dalam suatu data/informasi selama perjalanan.

3. *Nonrepudiation*. Pengirim seharusnya tidak dapat mengelak bahwa dialah pengirim data/informasi yang sesungguhnya.
4. *Authority.* Data/informasi yang berada pada sistem jaringan seharusnya hanya dapat di modifikasi oleh pihak yang berwenang

B. Temuan dan Interpretasi

Temuan yang didapat untuk proses keamanan file /informasi dengan menggunakan metode algoritma blowfish agar diproses sebuah data yang sederhana.

Misalkan plaintext yang penulis gunakan adalah "kamu dengan kata kuncinya "kalian".

C. Perubahan Plaintext dan key menjadi Biner

Sistem bilangan biner atau sistem bilangan basis dua adalah sebuah penulisan angka dengan menggunakan dua symbol yaitu 0 dan 1 kemudian cara merubah alphabet menjadi biner adalah dengan table ASCII [Roubaix 2013] .

Dari Tabel ASCII didapat, penulis akan merubah *plaintext* yang digunakan yaitu "kamu"

k = 01101011
a = 01100001
m = 01101101
u = 01110101

Sehingga plaintext "kamu" apabila dirubah menjadi biner akan menjadi "01101011011000010110110101110101". Plaintext tersebut bernilai 32 bit karena dihitung berdasarkan jumlah digit yang dihasilkan.

Kemudian dari table yang sama, penulis akan merubah key yang akan digunakan yaitu "kalian".

k = 01101011
a = 01100001
l = 01101100
i = 01101001
a = 01100001
n = 01101110

Sehingga key "kalian" apabila dirubah menjadi biner akan menjadi "011010110110000101101100011010010110000101101110 . Key tersebut bernilai 48 bit karena dihitung berdasarkan jumlah bit yang dihasilkan.

D. Enkripsi Blowfish

Pada pembahasan sebelumnya bahwa metode blowfish mengambil 64 bit data pada prosesnya sehingga plaintext mendapat tambahan bit tanpa merubah nilainya menjadi :

Plaintext=0000000000000000000000000000000001101011011000010110110101110101

Pada key proses pengambialnnya harus memiliki 32 bit atau kelipatannya sehingga pada key yang digunakan penulis mendapat tambahan bit tanpa merubah nilainya menjadi :

Key=00000000000000000110101101100001011011000110100101100000101101110

Plaintext yang sudah menjadi 64 bit kemudian dibagi 2, 32 bit pertama disebut XL, 32 bit yang kedua disebut XR sehingga menjadi:

XL = 00000000000000000000000000000000
XR = 01101011011000010110110101110101

Karena key merupakan kelipatan 32 bit maka, pembagian key menjadi : P1, P3, P5, P7, P9, P11, P13, P15, P17 = 00000000000000000110101101100001

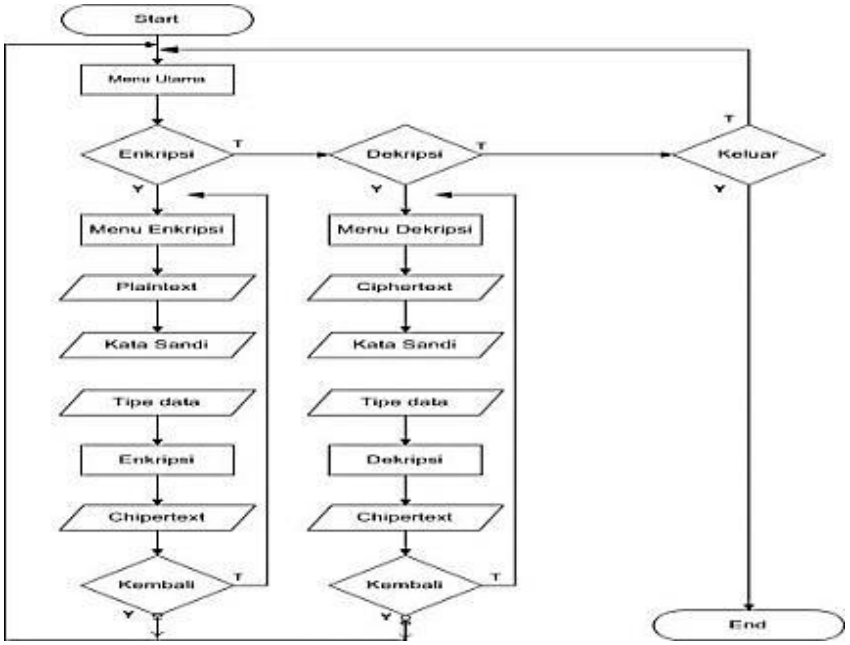
P2, P4, P6, P8, P10, P12, P14, P16, P18 = 01101100011010010110000101101110

Langkah selanjutnya adalah mengetahui fungsi "XOR", XOR berasal dari kata *Exclusive* OR. Fungsi ini akan memberikan keluaran 1 jika masukan-masukannya mempunyai keadaan yang berbeda. Fungsi XOR bertujuan untuk menghasilkan nilai logika TRUE (benar) jika salah satu dari pernyataan benar, dan menghasilkan nilai logika FALSE (salah) jika kedua pernyataan salah atau kedua pernyataan tersebut benar [Wen-dejavu 2013].

Masukan1	Masukan2	Keluaran
0	0	0
0	1	1
1	0	1
1	1	0

E. Perancangan Sistem

Pada bagian ini akan dilakukan peran cangan sistem aplikasi pengamanan data / file office sebagai lanjutan dari proses analisis sistem yang telah dilakukan sebelumnya. Perancangan sistem ini bertujuan untuk memberikan gambaran dan rancang bangun mengenai sistem yang akan dikembangkan. Perancangan menggunakan model *flowchart* susunan program. Adapun *flowchart* susunan program yaitu : menu utama, menu enkripsi, menu dekripsi :



Gambar 7. Flowchart Sususan Program

F. Perancangan Layar Aplikasi

1. Layar Menu Utama

Dalam merancang sebuah sistem aplikasi, salah satu hal yang perlu diperhatikan ada layar aplikasi atau *graphical user interface* (GUI). Berikut ini adalah GUI yang dirancang untuk aplikasi sistem pengamanan data / *file office* untuk enkripsi dan dekripsi pada kriptosistem simetris menggunakan pemograman Java.

Menu utama adalah sebagai menu awal ketika aplikasi dimulai, dimenu utama terdapat 3 tombol sebagai berikut :

1. Tombol ***Encrypt***, menekan tombol ini akan mengarahkan anda ke menu enkripsi jika anda ingin melakukan enkripsi.
2. Tombol ***Decrypt***, menekan tombol ini akan mengarahkan anda ke menu dekripsi jika anda ingin melakukan dekripsi dari file yang telah dienkripsi namun anda harus ingat kata kunci untuk dapat melakukan dekripsi dengan benar dan mendapatkan *plaintext* yang sebenarnya.
3. Tombol ***Close***, menekan tombol ini akan menutup dan keluar dari aplikasi.

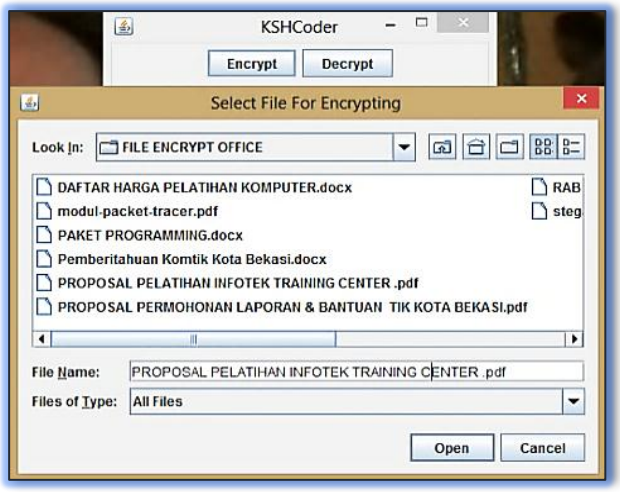
Gambar 8. Tampilan Layar Menu

2. Layar Menu Enkripsi

Menu Enkripsi pada aplikasi kriptografi file adalah menu tempat proses enkripsi terjadi. Adapun proses kerja dari menu *Encrypt* sebagai berikut :

1. Pilihlah Encrypt/Decrypt, dengan menekan kotak checkbox.
2. Tombol Encrypt, menekan tombol ini akan mengarah dan membuka Lokasi File yang akan anda minta untuk proses *Encrypt*.

3. *Look In : Select File for Encrypting* , Pilih folder yang aktif file yang telah disimpan untuk di seleksi pilihan agar di *Encrypt file office.*



Gambar 9. Tampilan Layar Menu *Enkripsi*

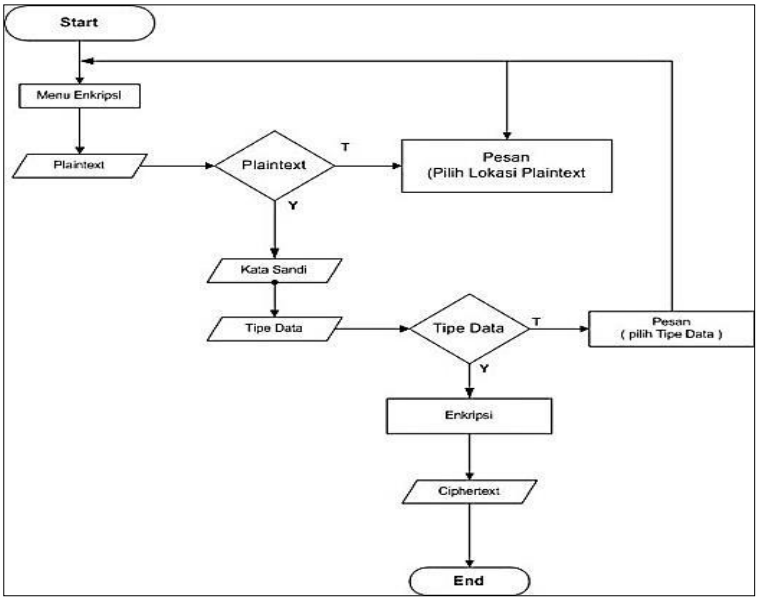
4. *Tombol Open,* menekan tombol ini akan membuka layar *Enter Key* sebagai masukkan kata kunci agar *file office* telah di kunci kerahasiaannya.

Gambar 10. Tampilan Layar Menu *Enter Key*

5.Tombol Proses *encrypt* dilakukan, pada saat tombol encrypt ditekan program akan mengambil *file* dari lokasi yang telah ditentukan oleh user kemudian menekan tombol *Open* agar program aplikasi akan mengeluarkan pesan enter key yang user masukkan secara manual menggunakan *keyboard* kata kunci mengunci / mengamankan file office maka akan muncul pesan proses Ready di view status.

Gambar 11. Tampilan Layar Menu *Encrypting*

Adapun model diagram flowchart proses Encrypt disajikan pada Gambar 12.



Gambar 12. Flowchart Proses *Encrypt*

3. Layar Menu Decrypt

Menu *Decrypt* pada aplikasi kriptografi file adalah menu tempat proses enkripsi terjadi. Adapun proses dari menu *Decrypt* adalah sebagai berikut :

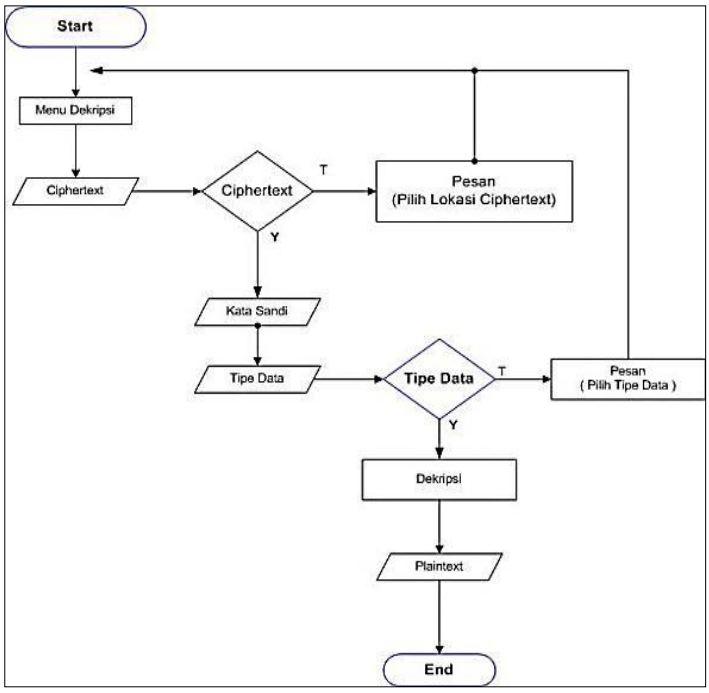
1. Pilihlah Encrypt/Decrypt, dan juga pilih Delete plain after Encryption dengan menekan kotak checkbox.
2. Tombol *Decrypt*, menekan tombol ini akan mengarah dan membuka Lokasi File yang akan di *Decrypt*.
3. *Select File for Encrypting* , Pilih folder yang aktif file yang telah di encrypt untuk di kembalikan data semula.
4. *Enter Key* , Masukkan kata kunci yang telah di encrypt sebelumnya dan juga akan di hapus file yang berubah menjadi *extention* .**enc**

Gambar 13. Tampilan Layar Menu *Enter Key*

5. *Tombol OK* , Masukkan kata kunci yang telah di encrypt sebelumnya dan juga akan di hapus file yang berubah menjadi *extention* .**enc** di folder aktif.

Gambar 14. Tampilan Layar Menu Proses *Decrypting*

Adapun model diagram flowchart proses *Decrypt* disajikan pada Gambar 15.



Gambar 15. Flowchart Proses *Decrypt*

4. Pengujian Sistem

Pengujian aplikasi untuk mengenkripsi *file office* dan setelah proses enkripsi selesai dilakukan akan dilihat hasilnya dilakukan pengujian dan waktu durasi dalam kecepatan (*speed*) dan ukuran data / *file* , maka dapat terlihat waktu enkripsi tidak kurang dari 5 KB dan apabila ukuran file asli besar maka akan cepat waktu enkripsi dan apabila ukuran filenya kecil maka waktu enkripsi semakin lama proses, ini dapat dilihat pada Tabel 2 dibawah ini :

TABEL 2.
HASIL PENGUJIAN SISTEM UNTUK FILE OFFICE

Nama *File*	Ukuran File Asli (KB)	Ukuran File Enkripsi (KB)	Waktu Enkripsi (s) Metode Blowfish	Waktu Dekripsi (s) Metode Blowfish
Daflatih Komputer.docx	43	43	4	4
RAB KOMTIK 2013.xlsx	18	18	4	4
SmartCar Android.pptx	97	97	3	3
Proposal Infotek.pdf	997	998	4	4
lagu relegi.mp3	5,078	5,078	1	1
video Corning glass.flv	15,011	15,011	4	4

5. Pengujian Black Box

Pada pengujian *black box* terfokus pada apakah implementasi program memenuhi kebutuhan dari analisis sistem yang telah ditentukan. Pengujian dilakukan dengan menjalankan atau mengeksekusi unit atau modul, kemudian diamati apakah hasil dari unit tersebut sesuai dengan proses yang dikehendaki atau tidak. Proses yang dijadikan objek pada pengujian *black box* ini terdiri dari modul *Encrypt*, modul *Enter Key*, modul *Decrypt*.

TABEL 3.
HASIL PENGUJIAN *BLACKBOX* PROSES MENU UTAMA

Menu Utama			
Input	**Yang Diharapkan**	**Hasil Pengamatan**	**Kesimpulan**
Menekan Tombol *Encrypt*	Proses *Select File for Encrypting* berhasil dan masuk ke file folder aktif	Proses *Select File for Encrypting* Berhasil di terima	Diterima
Modul *Enter Key*	Proses *enter key* input kode password dan confirm key input kode password yang sama berhasil di file di *Encrypt*	Proses *Enter key* dan *confirm key* berhasil *file office* di encrypt	Diterima
Menekan Tombol *Decrypt* dan option Delete file plain	Proses Decrypt dan option Delete plain melakukan *Select File for Encrypting* untuk *file* asli berhasil di plaintext.	Proses Decrypt data / file berhasil kembali menjadi plaintext	Diterima
Menekan Tombol *Close*	Proses Close dilakukan menutup Aplikasi menu utama berhasil masuk	Proses Close modul keluar dari Aplikasi menu utama berhasil	Diterima

G. Implikasi Penelitian

Berdasarka hasil dalam penelitian ini, implikasi penelitian yang ditinjau dari aspek sistem, manajerial, dan aspek penelitian lanjut dapat disusun.

Implikasi dari aspek sistem terkait dengan konsep strategic, taktis, sampai dengan teknik operasional, hardware, software, dan infrastruktur yang diperlukan. Implikasi dari aspek manajerial berkaitan dengan metode pengiriman data khususnya yang berkaitan dengan organisasi yang menekankan pada masalah keamanan data yang akan dikirimkan. Sedangkan dari aspek penelitian lanjut berkaitan dengan meningkatkan kualitas penelitian sebelumnya, seperti kekuatan kunci metode algoritma *Blowfish.*

1 Aspek Sistem

Dalam implementasi Sistem Keamanan Data kunci simetrik Algoritma *Blowfish* ini, aplikasi digunakan pada user di *computer client* untuk data / file yang akan dienkripsi baik yang terhubung dengan intranet maupun internet jika digunakan dalam lingkungan organisasi sendiri.

Dalam pengoptimalan penggunaan sistem yang dirancang berbasis computer, maka pengguna sebaiknya memahami penggunaan komputer secara umum, sehingga akan lebih mudah dalam penggunaan aplikasinya.

Sistem operasi yang diperlukan untuk implemetasi yaitu multi platform yang memudahkan dalam pengaplikasiannya serta aplikasi Java Netbeans Desktop. Selain itu metode Algoritma Blowfish dianggap sebagai algoritma kunci modern simetris berbentuk cipher block cukup terkuat strategi implementasi yang tepat akan lebih optimal dapat berjalan pada memori 5 KB dan kesederhanaan pada algoritmanya, sistem aplikasi kriptografi ini terletak pada kerahasiaan kunci dan bukan pada kerahasiaan algoritma digunakan. Saat ini dimana banyak perusahaan-perusahaan dibidang keamanan data, perbankan dan *broadcasting*. Hal ini tentu saja memancing orang yang tidak bertanggung jawab untuk membongkar sistem keamanan Algoritma ini. Sehingga penggunaan kekuatan kunci harus selalu dipertimbangkan.

2 Aspek Penelitian Lanjut

Dengan adanya penelitian ini maka kalangan akademis bisa menggunakan hasil penelitian sebagai referensi untuk penelitian yang sejenis dan bisa lebih mengembangkan lagi dalam penelitian berikutnya.

Upaya untuk meningkatkan penelitian berkaitan dengan sistem pengamanan Data menggunakan metode Algoritma *Blowfish* ini dapat dilakukan dengan memperluas ruang lingkup penelitian. Pada aplikasi yang dikembangkan dalam penelitian ini, penulis hanya membuat *prototype* sistem pengaman data secara umum saja. Dalam penerapannya bisa saja digunakan pada metode pembelajaran jarak jauh (e-learning), metode pengiriman Arsip Data Komputer (ADK) yang digunakan pada lembaga-lembaga negara maupun swasta dimana memang dibutuhlan *Confidentiality, Integrity, Availability, Non repudiation* dalam proses pengiriman datanya.

V. KESIMPULAN DAN SARAN

1. Kesimpulan

Berdasarkan permasalahan, studi pustaka, tinjauan penelitian, tinjauan obyek penelitian, metodologi penelitian dan pembahasan hasil dalam penelitian Prototipe Sistem Kriptografi kunci simetris metode *Blowfish* untuk enkkripsi dan dekripsi Data Office dengan Bahasa Pemrograman Java, maka dapat disimpulkan sebagai berikut :

1. Keamanan metode *Blowfish* merupakan salah satu algoritma yang tidak dipatenkan dan cukup kuat karena memiliki ruang kunci yang besar dan panjangnya bisa beragam, sehingga tidak mudah diserang pada bagian kuncinya. Suatu sistem kriptografi yang baik terletak pada kerahasiaan kunci dan bukan pada kerahasiaan algoritma yang digunakan. *Blowfish* pada strategi implementasi yang tepat akan lebih optimal, dapat berjalan pada memori kurang dari 5 KB dan kesederhanaan pada algoritmanya.
2. Penerapan kriptografi khususnya algoritma *Blowfish* dalam sistem pengamanan data office menggunakan enkripsi dan dekripsi dapat memberikan *otentikasi, integritas* dan *non repudiation* sebagai persyaratan penting bagi interaksi antara penerima dan pengirim data untuk identitas diri dari pemilik data.
3. Dengan adanya konsep kriptografi enkripsi dan dekripsi dengan metode Blowfish merupakan salah satu cara yang paling efektif untuk menghilangkan segala kekhawatiran akan perubahan data dan penggandaan dokumen file yang dapat melakukan oleh pihak-pihak yang tidak berhak.
4. Semakin besar data / ukuran file dokumen yang akan di enkripsi dan dekripsi maka semakin cepat waktu kecepatan dalam proses enkripsi dan dekripsinya.

2. Saran

Penggunaan kekuatan kunci dalam melakukan enkripsi dan dekripsi harus diperhatikan betul, karena tidak menutup kemungkinan kekuatan kunci terkuat saat ini dari pada metode algoritma yang digunakan dapat dipecahkan.

DAFTAR PUSTAKA

[1] Andriyansyah. Perangkat Lunak Keamanan Data menggunakan metode Blowfish dan metode RC4, Jakarta,2005.

[2] Ariyus, Dony. Computer Security, Yogyakarta: 2006.

[3] Ariyus, Dony. Pengantar Ilmu Kriptografi: Teori, Analisis, dan Implementasi. Andi, Yogyakarta, 2008.

[4] Asep, Herman, Suyanto. Pemograman Java: Konsep Pemrograman Berorientasi Objek< http://www.bambutechno.com> (Diakses 22 Januari 2014).

[5] Abdul Aziz. Implementasi Setganografi Teknik LSB (Least Significant Bit) dengan Algoritma Kriptografi Blowfish dan Fungsi Hash SHA-1: 2013.

[6] Burnett, Steve and Raine, Stephen. RSA Security's Official Guide to Cryptography. Mc Graw Hill Osborn, California: 2004.

[7] Christian, W. Dawson, Projects in Computing and Information Systems. < www.pearsoned.co.uk> England.First published: 2005 (Diakses 13 Maret 2013).

[8] Firmansyah. Kriptografi Sederhana http://politekniktelkom.ac.id/30110176/program-kriptografi-sederhana. [Diakses 12 Oktober 2013].

[9] Hariyanto, Bambang. Esensi-Esensi Bahasa Pemrograman Java. Informatika, Bandung: 2003.

[10] Hendra, kurniawan, Eri, Mardiani, Nur, Rahmansyah. Program Java Netbeans, XAMPP dan iReport. Elex Media Media Komputindo, Jakarta:2011.

[11] Irham. Delapan Aspek Keamanan Informasi. jurnal Nasional 2013-aspek-keamanan informasi.html. [Diakses 10 Oktober 2013].

[12] Kadir, Abdul. Dasar Pemrograman Java™ 2. Andi, Yogyakarta: 2005.

[13] Mary, Shaw. What Makes Good Research in Software Engineering School of Computer Science, Carnegie Mellon University, Pittsburgh PA 15213 USA <http://www.cs.cmu.edu/~shaw/> (Diakses 15 Maret 2013).

[14] Miftakhul, Huda Bunafit. Membuat Aplikasi mini/Supermarket dengan JAVA. Elex Media Komputindo:2011.

[15] Moedjiono. Pedoman Penelitian, Penyusunan dan Penilaian Tesis (V.5). Universitas Budi Luhur, Jakarta 2012 http://pascasarjana.budiluhur.ac.id/wpcontent/upload/2012/10/Pedoman-Tesis-PPS-UBL-V5-010112.pdf [Diakses 10 Oktober 2013].

[16] Moedjiono. Metodologi Penelitian dan Pedoman penyusunan Laporan Tugas Akhir Tesis & Paper. Universitas Budi Luhur, Jakarta,2013.

[17] Munir, Rinaldi. *Pengantar Kriptografi*. Bandung, Indonesia,2006.

[18] Prabowo, Setyo Budi. Definisi Keamanan Informasi. Informatika:2013

[19] Pressman, S.R. *Software Engineering a Practitioner's Approach,* USA Mc Grawhill. Inc:1997.

[20] Publikasi. Implementasi Algoritma Blowfish untuk aplikasi enkripsi dan Dekripsi File http://repository.amikom.ac.id/files/Publikasi_06.11.1189/.[Diakses 10 Oktober 2013].

[21] Saputra, Nugrah Dwi. Keamanan Informasi. SemnasIF 2010 [Diakses 10 Oktober 2013].
[22] Schneier.Bruce. Description of a new Variable-Length Key, 64-Bit Block Cipher (Blowfish), SpringerVerlag:1994.
[23] Schneier, Bruce. Applied Cryptography, Second Edition, John Wiley & Son, New York:1996.
[24] SeminasIf. Seminar Nasional Informatika http://repository.upnyk.ac.id/Aplikasi_Kriptografi_File_Menggunakan_Algoritma_Blowfish/. [Diakses 10 Oktober 2013].
[25] Seth, Shasi M. and Mishra, Rajan., "Comparative Analysis of Encryption Algorithms for Data Communication", *International Journal of Computer Science and Technology (IJCST)*, vol. 2, (Juni, 2011): 292-294.
[26] Stalling. Cryptography and Network Security Principles and Practices, Fourth Edition, 2005.
[27] Sumarkidjo dkk. Jelajah Kriptografi, Lembaga Sandi Negara, Jakarta, 2007.
[28] Tosin, Rijanto. Flowchart untuk Siswa dan Mahasiswa. Dinastindo. Jakarta, Indonesia, 1994.
[29] Tri, Andriyanto, Pardede. Studi Perbandingan IDEA dan Algoritma Blowfish. Seminar Nasional FTi Universitas Gunadarma Jakarta :2008.
[30] Wahana. Memahami Model Enkripsi dan Security Data, Andi Offset, Yogyakarta:2003.
[31] Whitten, L.J, Bentley, D.L & Dittman, C.K. *System Analysis and Design Methods,* USA Mc Grawhill, Inc:2004.
[32] Encryption Algorithm for JavaBased Devices", *International Seminar on Scientific Issues and Trends*:2011.