APLIKASI ENKRIPSI EMAIL DENGAN MENGGUNAKAN METODE BLOWFISH BERBASIS J2SE

Yanuar Firmansyah [1)], Sasono Wibowo (Pembimbing TA)[2)]

[1,2)]Sistem Informasi, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro

[1)]nd0el.z0ne@gmail.com

Abstract

Information and communication technology is growing more rapidly as the emergence of new problems in the delivery of information and communication. Many hardware and software with new features and technology that was created to meet the demands it. But the ease of access to all communications media impact for the security of information or messages using the communication media. Information to be very vulnerable to known, taken, and manipulated by parties who are not interested. For this phenomenon is needed a method to maintain the confidentiality of information especially email. The method in question is the way the encryption process. With the application of cryptography (encryption and description) that implement symmetric blowfish algorithms expected contents of the email message or information will be secure and not leaked to eavesdroppers or other irresponsible.

Keyword : Email Aplication, Encryption Email, Cryptography, Blowfish Cryptography, Blowfish Encryption

1. Pendahuluan

Teknologi informasi dan komunikasi berkembang semakin pesat seiring munculnya permasalahan baru dalam bidang penyampaian informasi dan komunikasi. Banyak perangkat keras dan perangkat lunak dengan fitur dan teknologi baru yang diciptakan untuk memenuhi tuntutan dalam bidang tersebut. Karena itulah nilai informasi saat ini sangat tinggi dan penting. Teknologi informasi yang telah terjalin saat ini berdiri diatas media komunikasi sebagai media penyampaian informasi dari satu tempat ketempat lainnya. Informasi-informasi yang ingin disampaikan berjalan melalui media komunikasi tersebut.

Media komunikasi yang banyak digunakan tentu harus merupakan media yang mudah dijangkau oleh semua orang. Contoh media komunikasi yang saat ini sering digunakan adalah telepon, jaringan internet dan *email*. Namun kemudahan pengaksesan media komunikasi oleh semua orang membawa dampak bagi keamanan informasi atau pesan yang menggunakan media komunikasi tersebut. Informasi menjadi sangat rentan untuk diketahui, diambil, dan dimanipulasi oleh pihak-pihak yang tidak berkepentingan.

Bukan hal aneh, jika melakukan pertukaran informasi melalui media jaringan seperti *internet* dan *email*. karena tentunya akan mempercepat dan memudahkan pertukaran informasi terutama untuk jarak yang jauh. Proses pertukaran informasi terutama pengiriman data tanpa melakukan pengamanan terhadap pesan atau informasi yang dikirim, sehingga mudah sekali dilakukan penyadapan pada jalur pengirimannya dan dapat langsung dibaca oleh penyadap. Sebagai contoh pengiriman informasi rahasia suatu perusahaan atau pun sekedar menyimpan data rahasia semisal informasi akun-akun, nomer rekening relasi, dan sebagainya dengan memanfaatkan sebuah akun *email*. bukan hal sulit seseorang yang tidak bertanggung jawab membajak akun *email* tersebut untuk kepentingannya sendiri.

Hal diatas merupakan gambaran bahwa *email* sekarang menjadi jalur pertukaran informasi yang sangat vital untuk berbagai kegiatan sehari-hari. Hal ini tentunya membuat keamanan informasi menjadi sesuatu yang penting. Dari fenomena tersebut maka dibutuhkan metode yang dapat menjaga kerahasiaan informasi khususnya *email*. Metode yang dimaksud adalah kriptografi yaitu sebuah seni dan bidang ke ilmuan dalam penyandian informasi atau pesan dengan tujuan menjaga keamanannya.

Dalam kriptografi terdapat metode yang cukup penting dalam pengamanan informasi atau pesan, salah satunya adalah enkripsi (*encryption*). Enkripsi adalah proses yang dilakukan untuk mengubah pesan asli menjadi pesan yang telah diubah supaya tidak mudah dibaca (*chipertext*). Sedangkan untuk mengubah pesan tersembunyi menjadi pesan biasa (*plaintext*) disebut deskripsi.

Berdasarkan paparan serta analisa masalah tersebut, maka penulis merasa perlu untuk membangun aplikasi penyandian informasi atau pesan yang akan dikirimkan atau

disimpan memanfaatkan media *email* yang ditujukan untuk membantu mengatasi masalah keamanan data sehingga orang lain tidak dapat mengetahui isi dari pada informasi atau pesan tersebut. Atas dasar hal-hal tersebut diatas, penulis mengambil judul "Aplikasi Enkripsi *Email* Dengan Menggunakan Metode *Blowfish* Berbasis J2SE".

a. Rumusan Masalah

Bagaimana melakukan pengamanan terhadap informasi atau pesan pada *email* dengan menggunakan algoritma *Blowfish* sehingga *email* tersebut terjaga keamanannya?

b. Batasan Masalah

1. permasalahan enkripsi dan deskripsi terhadap sebuah dokumen elektronik (pada kasus ini adalah *email*) berupa *text*.
2. Menggunakan algoritma *blowfish* dan pengaksesan ke server email dengan source kode yang telah tersedia pada media *library* java.

c. Tujuan Penelitian

membangun aplikasi untuk pengamanan dokumen elektronik (*email*) dengan kriptografi (enkripsi dan deskripsi), sehingga ketika dikirimkan melalui jaringan tidak aman isi dari *email* tersebut tidak bisa dimengerti oleh orang lain kecuali orang yang mempunyai aplikasi ini dan mengetahui kunci yang digunakan untuk mengenkripsi *email* tersebut.

2. Landasan Teori

a. Email

Surat elektronik atau pos elektronik (bahasa Inggris: *email*) adalah sarana mengirim surat melalui jalur jaringan komputer (misalnya internet).

1. SMTP

Simple Main Transfer Protocol (SMTP) merupakan salah satu protokol yang umum digunakan untuk pengiriman surat elektronik di internet. Protokol ini dipergunakan untuk mengirimkan data dari komputer pengirim ke server surat elektronik penerima.

2. POP3

Post Office Protocol version 3 (POP3) adalah protokol yang digunakan untuk mengambil surat elektronik (*email*) dari server *email*.

b. Kriptografi Blowfish

Blowfish atau disebut juga *OpenPGP.Chiper.4* adalah enkripsi yang termasuk dalam golongan *Symmetric Cryptosystem*. Algoritma kunci simetrik cipher blok yang dirancang pada tahun 1993 oleh Bruce Schneider untuk menggantikan DES (*Data Encryption Standard*).

Blowfish dirancang dan diharapkan mempunyai kriteria perancangan yang diiginkan sebagai berikut :

1. Cepat, *Blowfish* melakukan enkripsi data pada microprocessor 32-bit dengan rate 26 clock cycles per byte.
2. Compact, *Blowfish* dapat dijalankan pada memory kurang dari 5K.
3. Sederhana, *Blowfish* hanya menggunakan operasi – operasi sederhana, *Blowfish* hanya menggunakan operasi – operasi sederhana, seperti penambahan, XOR, dan lookup tabel pada operan 32-bit.

Memiliki tingkat keamanan yang bervariasi, panjang kunci yang digunakan oleh *Blowfish* dapat bervariasi dan bisa sampai sepanjang minimal 32-bit, maksimal 448 -bit, Multiple 8 bit, default 128 bit.

Dalam penerapannya sering kali algortima 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.

Algoritma Blowfish terdiri atas dua bagian, yaitu ekspansi kunci dan enkripsi data (Schneier, 1996).

a. Ekspansi Kunci (Key-Expansion)

Key expansion berfungsi untuk mengkonversikan sebuah kunci sampai 56 byte (448 bit)menjadi beberapa array subkey dengan total 4168 byte.

b. Enkripsi Data

1. Bentuk inisial P-array sebanyak 18 buah (P1, P2, ..., P18) masing-masing bernilai 32-bit. Array P terdiri dari delapan belas kunci 32-bit subkunci P1,P2, ..., P18
2. Bentuk S-box sebanyak 4 buah masing-masing bernilai 32-bit yang memiliki masukan 256. Empat 32-bit S-box masing-masing mempunyai 256 entri :

   $S_{1,0}, S_{1,1}, \ldots\ldots\ldots\ldots, S_{1,255}$

   $S_{2,0}, S_{2,1}, \ldots\ldots\ldots\ldots, S_{2,255}$

   $S_{3,0}, S_{3,1}, \ldots\ldots\ldots\ldots, S_{3,255}$

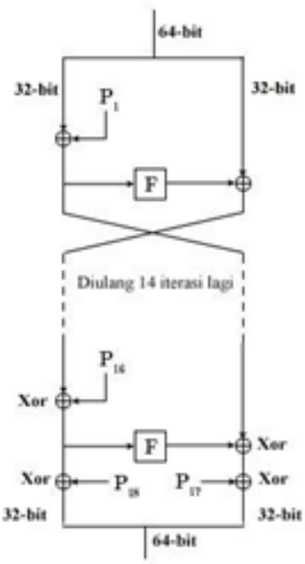
   $S_{4,0}, S_{4,1}, \ldots\ldots\ldots\ldots, S_{4,255}$

3. Plaintext yang akan dienkripsi diasumsikan sebagai masukan, Plaintext tersebut diambil sebanyak 64-bit, dan

apabila kurang dari 64-bit maka kita tambahkan bitnya, supaya dalam operasi nanti sesuai dengan datanya.

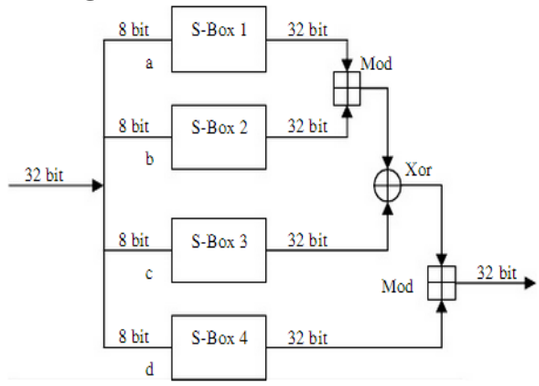
4. Hasil pengambilan tadi dibagi 2, 32-bit pertama disebut XL, 32-bit yang kedua disebut XR.
5. Selanjutnya lakukan operasi XL = XL xor Pi dan XR = F(XL) xor XR
6. Hasil dari operrasi diatas ditukar XL menjadi XR dan XR menjadi XL.
7. Lakukan sebanyak 16 kali, perulangan yang ke-16 lakukan lagi proses penukaran XL dan XR.
8. Pada proses ke-17 lakukan operasi untuk XR = XR xor P17 dan XL = XL xor P18.
9. Proses terakhir satukan kembali XL dan XR sehingga menjadi 64-bit kembali.

Blowfish menggunakan jaringan Feistel yang terdiri dari 16 buah putaran.



Gambar 1 : Jaringan *Feistel* untuk Algoritma *Blowfish*

Dalam algoritma *Blowfish* juga terdapat fungsi f. Berikut ini gambar mengenai fungsi f tersebut.



Gambar 2 : Fungsi F dalam *Blowfish*

3. Metode Penelitian

a. Ruang Lingkup Penelitian

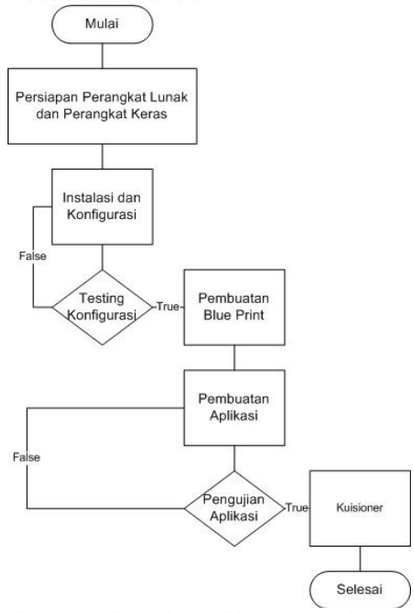
pembuatan atau implementasi aplikasi penyandian pada pesan (email) yang berupa *text* (String) menjadi blok bit dengan menggunakan algoritma *blowfish* untuk menyandikan bit pada pesan (email) sehingga memperoleh sebuah *chipertext*.

b. Kuesioner

Tabel 1 : Tabel Kuisioner

No	Uraian Pertanyaan	NILAI				
		1	2	3	4	5
A	**Kemampuan Software**					
1	Apakah fungsi Enkripsi dan Dekripsi email dapat berjalan dengan baik ?					
2	Bagaimana kinerja program aplikasi ?					
3	Apakah fungsi pengiriman dan pengunduhan email dapat berjalan dengan baik ?					
B	**Interaksi Manusia dan Komputer**					
1	Apakah pengguna dapat menggunakan dengan baik ?					
2	Bagaimana bentuk desain program ?					
3	Apakah program dapat berjalan dengan lancar ?					

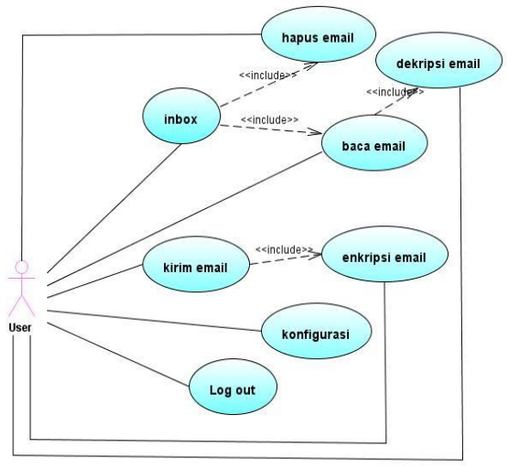
c. Metode Penelitian



Gambar 3 : *Flow Chart* Penelitian

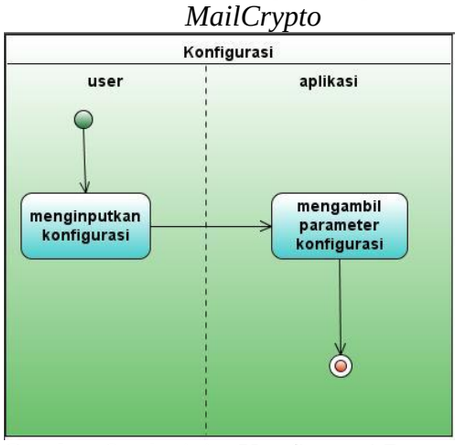
4. Perancangan dan Hasil Implementasi

a. Unit Bahasa Pemodelan

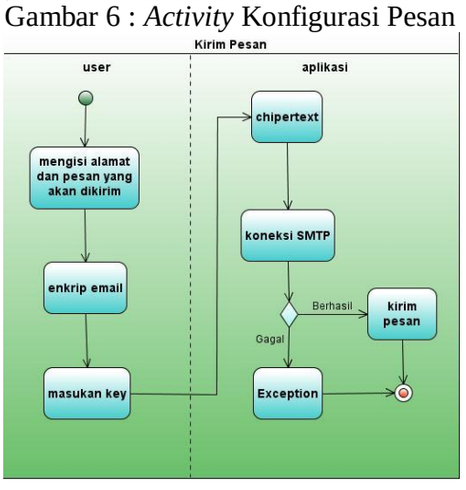


Gambar 4 : *Use Case* Aplikasi *MailCrypto*

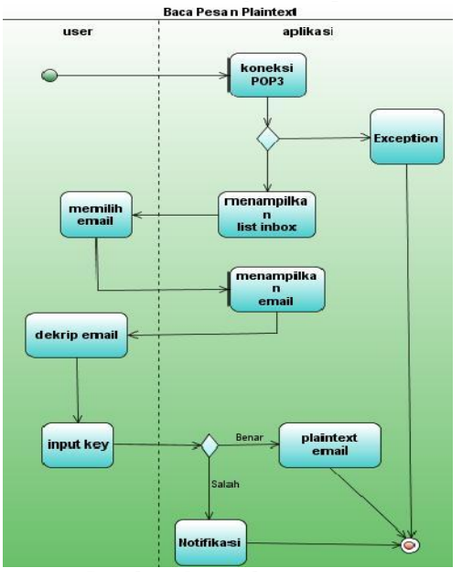
Gambar 5 : *Class Diagram* Aplikasi *MailCrypto*



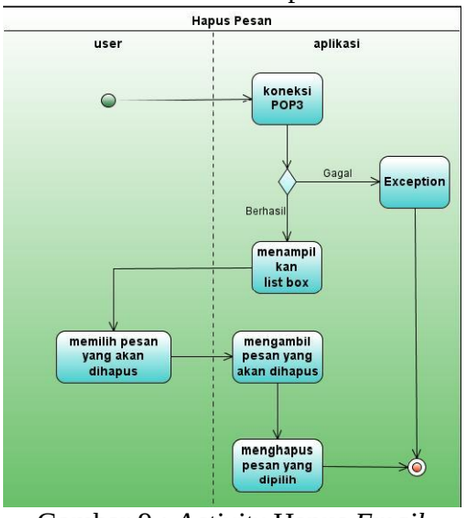
Gambar 6 : *Activity* Konfigurasi Pesan



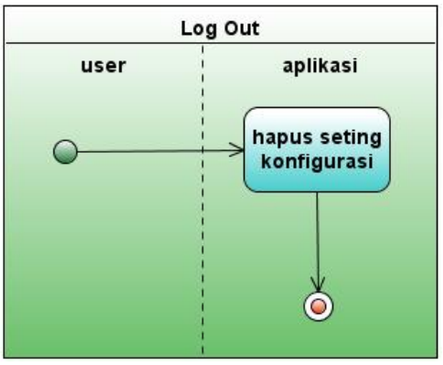
Gambar 7 : *Activity* Enkrip Pesan



Gambar 8 : Dekrip Pesan



Gambar 9 : *Activity* Hapus *Email*



Gambar 10 : *Activity Log Out*

b. Implementasi

Pada tahap ini penulis mentransformasikan dari blueprint ke dalam program. Penulis memanfaatkan kamus fungsi standar dari java yang mendukung jalannya aplikasi ini. Berikut daftar kamus fungsi yang penulis gunakan :

- **java.io.***
  Berfungsi untuk menangani *input output* serta menangani *exception* untuk *input ouput*.
- **java.util.Properties**
  Berfungsi sebagai penyimpan konfigurasi yang nantinya dipakai untuk

menyambungkan ke akun *mail server* yang dituju.

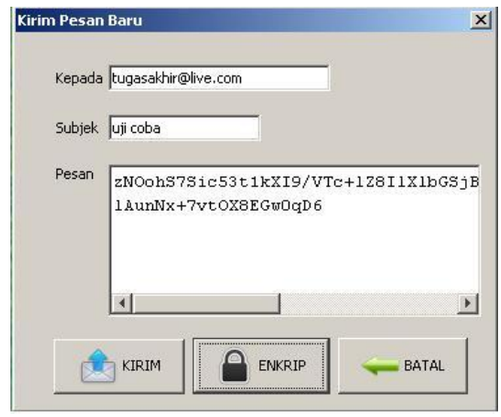
- **java.security.***
  Berfungsi untuk menangani *exception* terhadap kesalahan penggunaan algoritma kriptografi dan algoritma kunci untuk proses enkripsi dan dekripsi.
- **javax.mail.***
  Berfungsi sebagai *session authenticator user* dan *password* agar terhubung dengan akun *mail server*.
- **javax.mail.internet.***
  Berfungsi sebagai manajemen untuk mengirim *email* dari aplikasi ke *mail server*.
- **javax.swing.***
  Berfungsi mendefinisikan fungsi GUI yang ada dalam aplikasi.
- **javax.crypto.***
  Berfungsi sebagai sistem pengenkripsian *plaintext* dan kunci dengan algoritma *Blowfish*.
- **com.sun.mail.imap.protocol.FLAGS**
  Berfungsi untuk menghapus *email* yang ada didalam *inbox mail server*.
- **sun.misc.BASE64Encoder()**
  Berfungsi untuk merubah *plaintext* yang bernilai *byte* yang berbentuk UTF-8 kedalam String.
- **sun.misc.BASE64Decoder()**
  Berfungsi untuk merubah *chipertext* menjadi bernilai *byte* berbentuk UTF-8 sehingga dapat di dekripsikan kembali dengan baik.

Gambar 11 : Main Menu Aplikasi

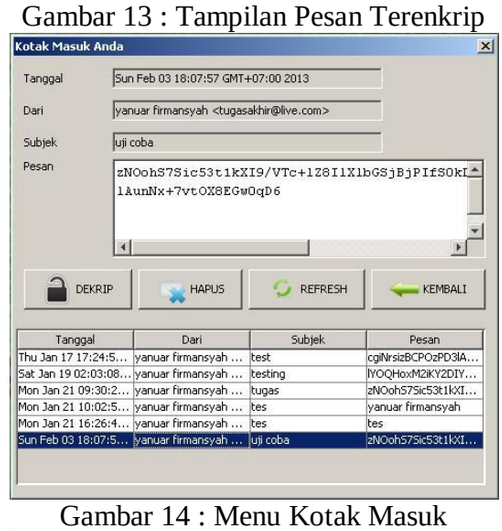
MailCrypto

c. Analisa Percobaan

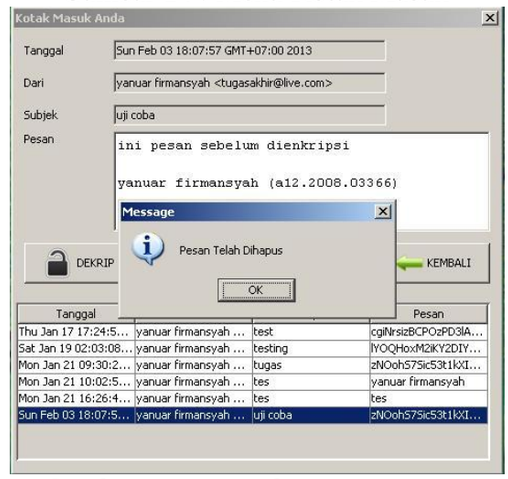
Gambar 12 : Menu Konfigurasi



Gambar 13 : Tampilan Pesan Terenkrip



Gambar 14 : Menu Kotak Masuk



Gambar 15 : Tampilan Hapus Pesan

5. Kesimpulan

1. Aplikasi ini dapat melakukan pengamanan terhadap informasi atau pesan pada *email* dengan metode *Blowfish* (enkripsi dan dekripsi).
2. Penggunaan kamus fungsi standar dari *java* dan *sun microsystem* meminimalkan pembengkakan *coding* pada aplikasi ini.

DAFTAR PUSTAKA

[1] Anonimous. Kriptografi. http://id.wikipedia.org/wiki/Kriptografi. 29 Oktober 2012.

[2] Anonimous. *E-Mail*. http://id.wikipedia.org/wiki/Surat_elektronik. 1 November 2012.

[3] Anonimous. *SMTP*. http://id.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol. 15 November 2012.

[4] Anonimous. *POP3*. http://id.wikipedia.org/wiki/Post_Office_Protocol. 15 November 2012.

[5] Schneier, Bruce. (1996). *Applied Cryptography: Protocols, Algorithms, and Source Code in C, Second Edition (Paperback).* USA: Wiley.

[6] Schneier, Bruce. (1993). *Applied Cryptography :Protocols, Algorithms, and Source Code in C (Paperback).* USA: Wiley.