APLIKASI ENKRIPSI EMAIL DENGAN MENGGUNAKAN METODE BLOWFISH BERBASIS J2SE

Sasono Wibowo[1], Suprayogi[2]
[1]Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
[2]Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
Jl. Nakula I No. 5 – 11 Semarang, 50131, Telp. (024) 3517261, Fax (024) 3520165
E-mail : sasono.wibowo@dsn.dinus.ac.id[1], suprayogismg@gmail.com[2]

Abstrak

Seiring berkembangnya Teknologi Informasi yang sangat pesat dan semakin mudahnya orang untuk melakukan komunikasi maka permasalahan barupun muncul dalam bidang penyampaian informasi dan komunikasi, karena 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. Sehingga kebutuhan akan keamanan dalam melakukan komunikasi menjadi hal yang sangat penting. Banyak perangkat keras dan perangkat lunak dengan fitur dan teknologi baru yang diciptakan untuk memenuhi tuntutan dalam bidang tersebut. Dari fenomena tersebut maka dibutuhkan suatu metode yang dapat menjaga kerahasiaan informasi khusunya email. Metode yang dimaksud adalah dengan cara proses enkripsi. Dengan aplikasi kriptografi (enkripsi dan deskripsi) yang menerapkan algoritma simetris blowfish diharapkan isi pesan atau informasi dari email akan aman dan tidak bocor kepada penyadap atau pihak yang tidak bertanggung jawab.

Kata Kunci : *enkripsi, dekripsi,email, blowfish.*

Abstract

As the development of information technology is very rapid and increasingly easy for people to communicate then problems arise in the field of delivery of information and communication, because of the ease of accessing a communication media by everyone has had implications 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 allowed to access. Thus the need for security in communication becomes very important . Many hardware and software with new features and technology that was created to meet the demands in the field. Of these phenomena we need 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 implements symmetric blowfish algorithm is expected contents of the email message or the information will be secure and not leaked to eavesdroppers or parties who are not permitted .

Keywords: *encryption, decryption, email, blowfish.*

1. PENDAHULUAN

Dengan perkembangan teknologi informasi dan komunikasi yang berkembang sangat pesat namun juga dibarengi dengan tuntutan keamanan dan kerahasiaan informasi yang disampaikan, banyak perangkat keras dan perangkat lunak dengan fitur dan teknologi baru yang diciptakan untuk memenuhi tuntutan dalam bidang 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* [1]. 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 [2].

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 keilmuan 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 [3].

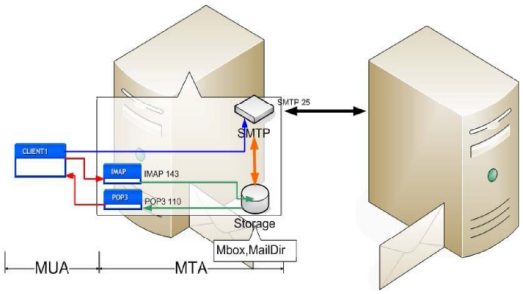
Berdasarkan paparan tersebut diatas, maka perlu kiranya dibangun suatu aplikasi penyandian informasi atau pesan yang akan dikirimkan atau disimpan memanfaatkan media *email* yang ditujukan untuk membantu mengatasi masalah keamanan data sehingga orang lain tidak dapat mengetahui isi dari pada informasi atau pesan tersebut.

Surat elektronik atau pos elektronik (bahasa Inggris: *email*) adalah sarana mengirim surat melalui jalur jaringan komputer (misalnya internet). Dengan surat biasa pada umumnya pengirim perlu membayar per pengiriman (dengan membeli perangko), tetapi surat elektronik umumnya biaya yang dikerluarkan adalah biaya untuk membayar sambungan internet. *Email* merupakan aplikasi TCP/IP yang paling banyak digunakan. *Email* adalah pesan yang terdiri atas kumpulan string ASCII dalam format RFC 822. Sistem *email* yang beroperasi di atas jaringan berbasis pada model *store and forward*. Sistem ini mengaplikasikan sebuah sistem server *email* yang menerima,

meneruskan, dan mengirimkan, serta menyimpan pesan-pesan user, dimana user hanya perlu untuk menghubungkan komputer mereka kedalam jaringan.

Email yang dikirim belum tentu akan diteruskan komputer penerima (*end user*), tapi disimpan atau dikumpulkan dahulu dalam sebuah komputer server (*host*) yang akan online secara terus menerus (*continue*) dengan media penyimpanan (*storage*) yang relatif besar dibanding komputer biasa. Hal ini bisa diibaratkan dengan sebuah kantor pos, jika seseorang mempunyai alamat (*mailbox*), maka dia dapat memeriksa secara berkala jika dia mendapatkan surat. Komputer yang melayani penerimaan *email* secara terus-menerus tersebut biasa disebut dengan *mailserver* atau *mailhost*.

Secara umum, sistem kerja dari *email* adalah seperti gambar dibawah ini :



Gambar 1. Sistem kerja *email*

Simple Mail Transfer Protocol (SMTP) adalah suatu protokol yang digunakan untuk mengirimkan pesan e-mail antar server, yang bisa dianalogikan sebagai kantor pos. Ketika kita mengirim sebuah e-mail, komputer kita akan mengarahkan e-mail tersebut ke sebuah SMTP server, untuk diteruskan ke mail-server tujuan. Mail-server tujuan ini bisa dianalogikan sebagai kotak pos di pagar depan rumah kita, atau kotak PO BOX di kantor pos. Email-email yang terkirim akan "bertengger" di tempat tersebut hingga Si pemiliknya mengambilnya. Urusan pengambilan e-mail tersebut tergantung kapan dipenerima memeriksa account e-mailnya.

Post Office Protocol version 3 adalah suatu protokol yang berfungsi untuk menarik atau mengambil email dari server email yang digunakan. Untuk menggunakan POP3 bisa dari Microsoft Outlook. biasanya untuk menggunakan POP3 di perlukan settingan:

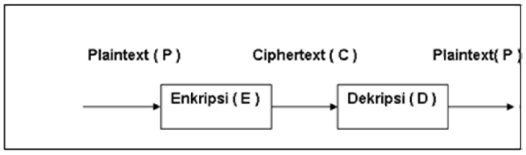
1. Email Address : contoh —> anda@domainanda.com
2. Incoming Mail (POP3, IMAP or HTTP) server : mail.doaminanda.com
3. Outgoing (SMTP) server : mail.domainanda.com
4. Account Name : anda@domainanda.com
5. Password : password yang telah anda buat sebelumnya

Protokol POP3 dibuat karena desain dari sistem surat elektronik yang mengharuskan adanya server surat elektronik yang menampung surat elektronik untuk sementara sampai surat elektronik tersebut diambil oleh penerima yang berhak. Kehadiran server surat elektronik ini disebabkan kenyataan hanya sebagian kecil dari komputer penerima surat elektronik yang terus-menerus melakukan koneksi ke jaringan internet. Protokol ini di spesifikan pada RFC 1939.

Sistem kriptogrfi atau *cryptosystem* adalah sebuah algoritma ditambah semua kemungkinan *plaintext, ciphertext* dan *kunci* [4]. Dalam sistem ini, seperangkat parameter yang menentukan transformasi *cipher* tertentu disebut suatu set kunci. Proses

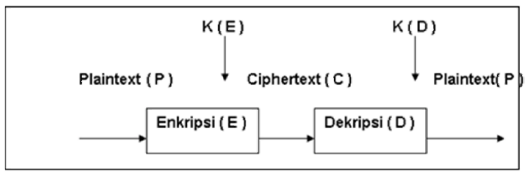
enkripsi dan dekripsi diatur oleh satu atau beberapa kunci kriptografi. Secara umum, kunci-kunci yang digunakan untuk proses enkripsi dan dekripsi tidak perlu identik, tergantung pada sistem yang digunakan.

Pesan yang akan dienkripsi yang dimisalkan *plaintext* (P), proses enkripsi dimisalkan enkripsi (E), proses dekripsi dimisalkan dekripsi (D), dan pesan yang sudah dienkripsi disebut *ciphertext* yang dimisalkan *ciphertext* (C) maka dapat digambarkan pada gambar berikut ini :



Gambar 2. Proses enkripsi dan deskripsi

Data atau informasi yang akan dienkripsi (*plaintext*) diacak oleh suatu kunci yang telah ditentukan kemudian *output* dari proses enkripsi (*ciphertext*) dikembalikan kebentuk aslinya oleh sebuah kunci yang sama.



Gambar 3. Proses enkripsi dan deskripsi dengan kunci K

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*).

Algoritma *Blowfish* dibuat untuk digunakan pada komputer yang mempunyai mikroprosesor besar (32 bit ke atas dengan *cache* data yang besar). Pada saat itu banyak sekali rancangan algoritma yang ditawarkan, namun hampir semua terhalang oleh paten atau kerahasiaan pemerintah Amerika. Schneier menyatakan bahwa *blowfish* bebas paten dan akan berada pada domain publik. Dengan pernyataan Schneier tersebut *blowfish* telah mendapatkan tempat di dunia kriptografi, khususnya bagi masyarakat yang membutuhkan algoritma kriptografi yang cepat, kuat, dan tidak terhalang oleh lisensi.

Blowfish dirancang dan diharapkan mempunyai kriteria perancangan yang diiginkan sebagai berikut [4]:

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.
4. 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.

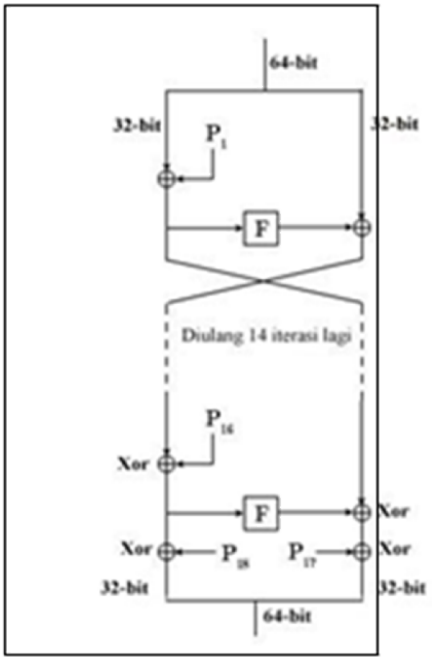
Enkripsi data, proses ini terjadi di dalam jaringan feistel dan terdiri dari iterasi fungsi sederhana sebanyak 16 kali putaran. Setiap putaran terdiri dari permutasi key-dependent serta substitusi kunci dan data-dependent. Semua operasi merupakan XOR dan penjumlahan (*addition*) pada variable

32 bit. Operasi penambahan yang terjadi hanya merupakan empat indeks array data lookup pada setiap iterasi.

Blowfish menggunakan subkunci besar yang harus dihitung sebelum enkripsi dandekripsi data. Algoritma *Blowfish* menerapkan jaringan Feistel yang terdiri dari 16 putaran. *Input* adalah elemen 64-bit, X untuk alur algoritma enkripsi dengan metode *Blowfish*dijelaskan sebagai berikut :

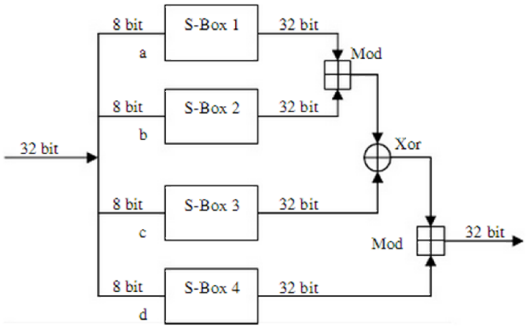
1. Bentuk inisial P-array sebanyak 18 buah (P1, P2, ..., P18) masing-masing bernilai 32-bit. Array P terdiri dari delapan belas kunci 32-bit subkunci P1,P2, ..., P18
2. Bentuk S-box sebanyak 4 buah masing-masing bernilai 32-bit yang memiliki masukan 256. Empat 32-bit S-box masing-masing mempunyai 256 entri :
   $S_{1,0}$,$S_{1,1}$,.....,$S_{1,255}$
   $S_{2,0}$,$S_{2,1}$,.....,$S_{2,255}$
   $S_{3,0}$,$S_{3,1}$,.....,$S_{3,255}$
   $S_{4,0}$,$S_{4,1}$,.....,$S_{4,255}$
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 terdiridari 16 buah putaran.



Gambar 4. Jaringan *Feistel* untuk Algoritma *Blowfish*

Algoritma *Blowfish* memiliki keunikan dalam hal proses dekripsi, yaitu proses dekripsi dilakukan dengan urutan yang sama persis dengan proses enkripsi, hanya saja pada proses dekripsi P1, P2, …, P18 digunakan dalam urutan yang terbalik. Dalam algoritma *Blowfish* juga terdapat fungsi f. Berikut ini gambar mengenai fungsi f tersebut.



Gambar 5. Fungsi F dalam *Blowfish*

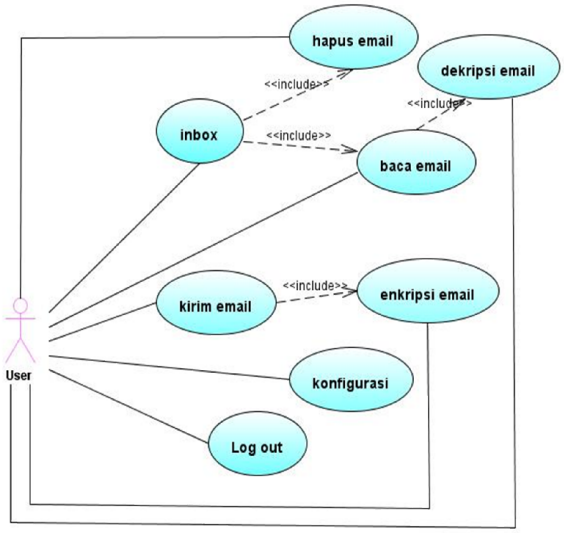
J2SE atau Java 2 Standard Edition merupakan bahasa pemrograman Java untuk aplikasi desktop yang merupakan object-oriented programming [5]. Pada J2SE, terdiri dari dua buah produk yang dikeluarkan untuk membantu dalam membuat aplikasi tanpa tergantung dari platform yang digunakan, yaitu :

1. Java SE Runtime Environment (JRE)
2. Java Development Kit (JDK)

3. METODE

3.1 Use Case Diagram [6]

Dalam bahasa pemodelan ini, peneliti menggunakan 1 (satu) buah actor yaitu *user*. *User* pada aplikasi ini adalah seseorang yang nantinya akan menjalankan aplikasi CryptoMail ini. Di Gambar 6 merupakan pemodelan use case yang peneliti pakai pada pembuatan aplikasi ini.



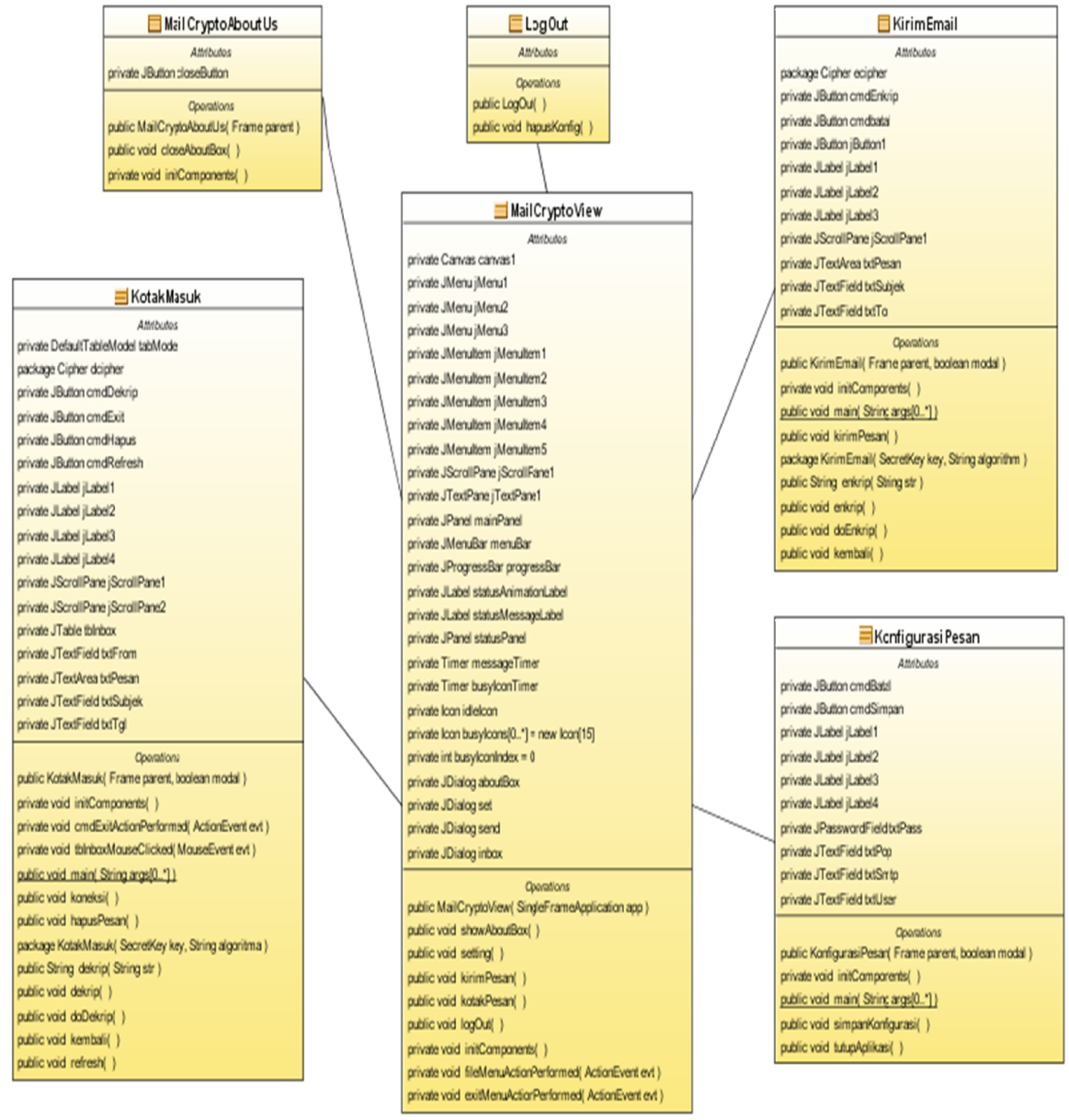
Gambar 6. *Use Case* Aplikasi *MailCrypto*

3.2 *Class Diagram* [6]

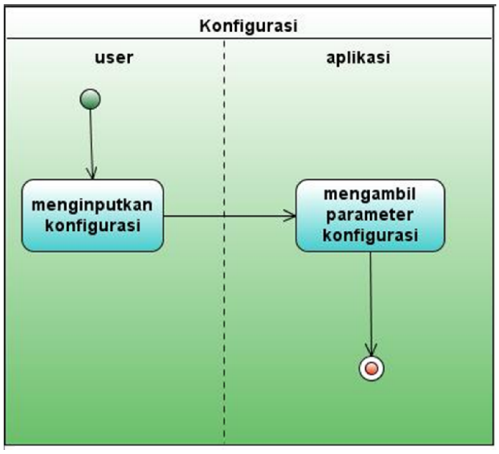
Pada *class* diagram, peneliti menggunakan 6 macam kelas yaitu *class KonfigurasiPesan, KirimEmail, KotakMasuk, LogOut, MailCryptoView, MailCryptoAboutUs*. Kelas-kelas tersebut saling berhubungan dan mempunyai keterkaitan. Gambar 7 menunjukkan gambar *Class Diagram.*

4. HASIL DAN PEMBAHASAN

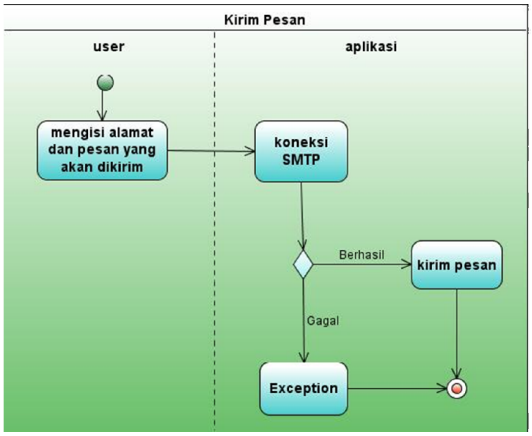
Activity Diagram yang peneliti buat saat ini menggunakan 8 macam model diagram yaitu diagram pada saat kirim pesan, baca pesan, dekrip pesan, konfigurasi pesan, kotak masuk, enkrip pesan, hapus pesan, log out.



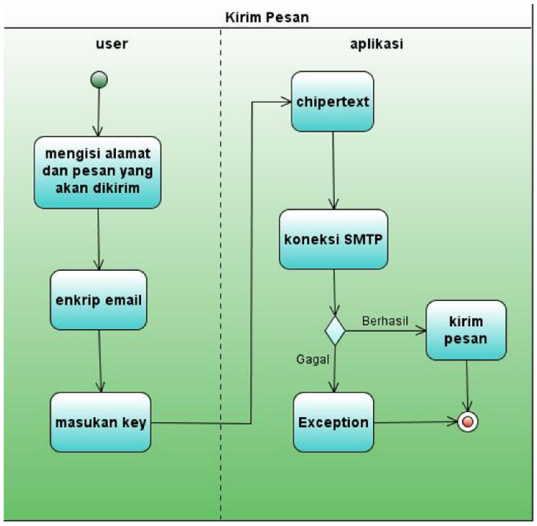
Gambar 7. *Class Diagram* Aplikasi *MailCrypto Activity* Diagram



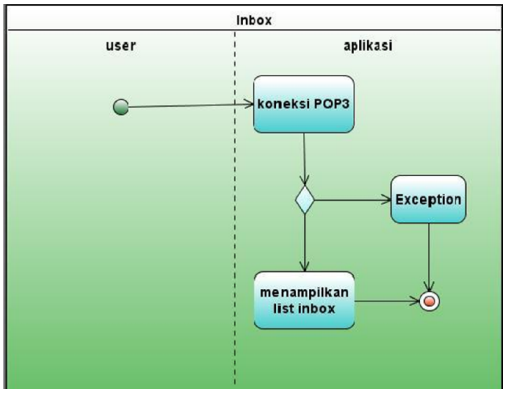
Gambar 8. *Activity* Konfigurasi Pesan



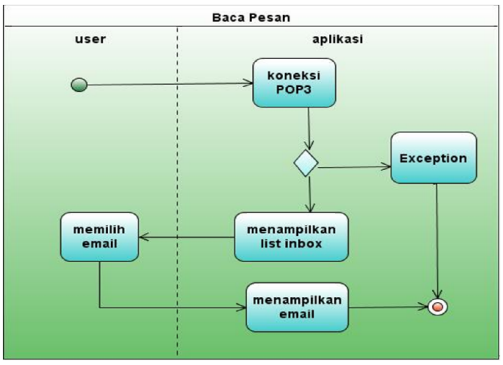
Gambar 9. *Activity* Kirim Pesan



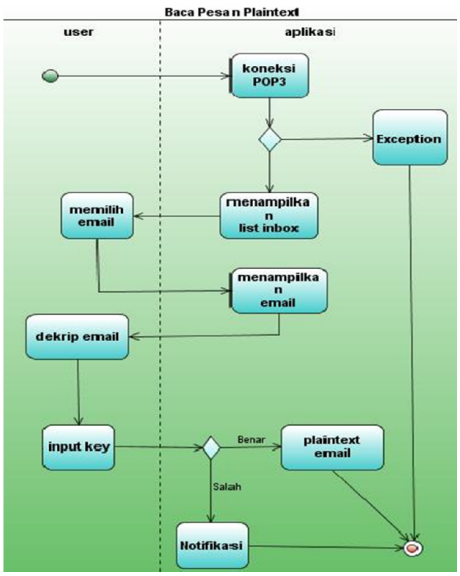
Gambar 10. *Activity* Enkrip Pesan



Gambar 11. *Activity* Kotak Masuk



Gambar 12. Baca Pesan



Gambar 13. Dekrip Pesan

4. KESIMPULAN DAN SARAN

4.1 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.

4.2 Saran

1. Penambahan fungsi yang menyimpan daftar dari POP3 serta SMTP dari beberapa server email, sehingga user tidak perlu mengetikkan konfigurasi untuk POP3 dan SMTP secara manual.
2. Proses Otentikasi ke server *email* yang lebih baik agar aplikasi ini dapat berjalan lancar.
3. Penambahan fungsi *attachment* agar aplikasi ini mampu mengamankan pesan atau informasi dari jenis file apapun tidak hanya berupa text.
4. Penambahan fungsi untuk pembacaan email yang berbentuk *content*.
5. Pembuatan aplikasi berupa Web Service, sehingga aplikasi ini bisa diakses dari platform manapun.

DAFTAR PUSTAKA

[1] Tri Aditya Sasongko (2009). *Rancang Bangun Email Client Pada Perangkat Mobile. Jurnal Penelitian*. Institut Teknologi Sepuluh Nopember.
[2] Munir, Rinaldi (2006). *Kriptografi*. Bandung :Informatika.
[3] Ariyus, D. (2008). *Pengantar Ilmu Kriptografi Teori, Analisis, dan Implementasi.* Yogyakarta: Andi.
[4] Schneier, Bruce. (1996). *Applied Cryptography: Protocols, Algorithms, and Source Code in C, Second Edition (Paperback)*. USA: Wiley.
[5] Haryanto, Bambang. (2011). *Esensi-Esensi Bahasa Pemrograman Java*. Bandung :Informatika.
[6] Pressman RS. 1997. Rekayasa Perangkat Lunak. Edisi ke-2. LN Harnaningrum, penerjemah. Yogyakarta. Andi. Terjemahan dari: *Software Engineering, a Practitioner's Approach. Edisi ke-4. McGraw-Hill Companies, Inc.*