PENERAPAN SISTEM KRIPTOGRAFI MENGGUNAKAN ALGORITMA ELGAMAL PADA APLIKASI EMAIL BERBASIS WEB

Devie R Suchendra[1], Charel Samuel M[2]
[1]Teknik Informatika,Institut Teknologi Telkom, [2]Teknik Informatika,STMIK LPKIA
Jln. Soekarno Hatta No. 456 Bandung 40266, Telp. +62 22 75642823, Fax. +62 22 7564282
Email :deviersuchendra@gmail.com[1], shaunfeel.shauntriva@gmail.com[2]

Kriptografi adalah ilmu yang bersandarkan pada teknik matematika untuk berurusan dengan keamanan dan informasi seperti kerahasiaan, keutuhan data dan otentikasi entitas. Dengan kata lain kriptografi melakukan penyandian terhadap teks asli (*plaintext*) menjadi teks sandi (*chippertext*). Untuk melakukan penyandian sebuah pesan asli menjadi pesan sandi (enkripsi) atau mengubah kembali pesan sandi menjadi pesan asli (dekripsi) diperlukan kunci rahasia. Salah satu sistem kriptografi adalah Sistem Kriptografi Elgamal. Sistem Kriptografi Elgamal merupakan sistem kriptografi dengan kunci asimetri, dimana kunci untuk mengenkripsi dan mendekripsi adalah berbeda.
Untuk menerapkan Sistem Kriptografi Elgamal, akan dibuat sebuah aplikasi *email* berbasis *web*. Aplikasi *email* tersebut akan dilengkapi dengan proses enkripsi untuk dekripsi dengan Sistem Kriptografi Elgamal terhadap isi pesannya untuk mengetahui bagaimana pembangkitan kunci rahasia, proses enkripsi, dan proses dekripsi pada Sistem Kriptografi Elgamal. Serta untuk mengetahui berapa besarnya perubahan data pada pesan setelah proses enkripsi, dan waktu yang dibutuhkan sebuah sistem komputer untuk melakukan proses enkripsi dan dekripsi menggunakan Sistem Kriptografi Elgamal.

Kata kunci : *Kriptografi Elgamal, Enkripsi, Dekripsi, Email*

1. Pendahuluan

Perkembangan pesat teknologi informasi salah satunya berdampak pada aspek komunikasi. Dewasa ini, kita dapat berkomunikasi dengan orang lain dengan cepat dan mudah dengan adanya *internet*.

Internet menyediakan layanan komunikasi contohnya untuk pengiriman pesan, surat (*email*), *chatting*, bahkan *video chat*. Namun, ternyata internet tidak terlalu aman untuk berkomunikasi khususnya untuk pengiriman pesan yang sifatnya penting dan rahasia.

Oleh sebab itu terdapat teknik penyandian untuk merahasiakan isi pesan agar tidak dapat dimengerti oleh pihak-pihak yang tidak berhak. Teknik penyandian atau kriptografi dapat dibedakan menjadi dua jenis berdasarkan pendistribusian kuncinya, yaitu Kriptografi kunci simetri dan Kriptografi dengan kunci asimetri. Kriptografi dengan kunci simetri adalah sistem kriptografi dimana kunci untuk mengenkripsi dan mendekripsi adalah sama.

Sistem kriptografi kunci asimetri memiliki kunci untuk enkripsi dan kunci untuk dekripsi yang berbeda. Kunci untuk enkripsi disebut juga sebagai kunci publik yang bersifat tidak rahasia sehingga dapat didistribusikan melalui saluran tidak aman. Sedangkan kunci dekripsi disebut juga kunci privat sehingga bersifat rahasia dan harus dijaga kerahasiaanya oleh pemegang kunci. Salah satu Sistem kriptografi kunci asimetri adalah Elgamal.

Adapun permasalahan yang diangkat adalah sebagai berikut :

1. Bagaimana proses pembangkitan kunci publik dan kunci privat pada Sistem Kriptografi Elgamal ?
2. Bagaimana proses enkripsi dan proses dekripsi pada sebuah pesan teks menggunakan Sistem Kriptografi Elgamal ?
3. Bagaimana perubahan besarnya data yang terjadi pada sebuah pesan berupa *plaintext* yang diubah menjadi *chippertext* pada Kriptografi Elgamal ?
4. Berapa lama waktu yang dibutuhkan sebuah sistem komputer untuk melakukan proses enkripsi dan dekripsi pada sebuah pesan teks menggunakan Algoritma Kriptografi Elgamal ?

Adapun tujuan dari perancangan sistem yang baru adalah sebagai berikut :

1. Mengetahui proses pembangkitan kunci publik dan kunci privat pada Sistem Kriptografi Elgamal.
2. Mengetahui proses enkripsi dan dekripsi pada sebuah pesan teks menggunakan Sistem Kriptografi Elgamal.
3. Mengetahui perubahan besarnya data yang terjadi pada sebuah pesan yang telah dienkripsi menggunakan Sistem Kriptografi Elgamal dari *plaintext* menjadi *chippertext*.
4. Mengetahui waktu yang dibutuhkan sebuah sistem komputer untuk melakukan proses enkripsi dan proses dekripsi pada sebuah pesan teks menggunakan Sistem Kriptografi Elgamal.

1.1 Landasan Teori

Penemu sistem kriptografi Elgamal adalah Taher Elgamal pada tahun 1984. Sistem kriptografi Elgamal bersandar pada asumsi kesulitan persoalan logaritma diskrit. Algoritma pembangkit kunci Elgamal :

1. Pilih bilangan prima p besar sebagai basis grup perkalian (Z^*_p, x).
2. Pilih α sebagai akar primitif pada grup (Z^*_p, x).
3. Pilih d yang memenuhi $1 \leq d \leq p-2$.
4. Hitung $\beta = \alpha^d \bmod p$.
5. Kpublik = (p, α, β), Kprivat = d.

Perhitungan proses enkripsi Sistem Kriptografi Elgamal :

1. Input : Kpublik = (p, α, β), P ϵ (Z^*_p, x).
2. Output : C1, C2 ϵ (Z^*_n).
3. Proses : r ← (Z^*_n) { r dipilih acak}, $C1 = \alpha^r \bmod p$, $C2 = (P \times \beta^r) \bmod p$.

Perhitungan proses dekripsi Sistem Kriptografi Elgamal :

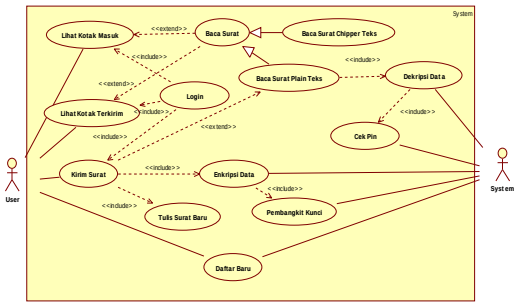
1. Input : Kprivat = d, C1, C2 ϵ (Z^*_p, x).
2. Output : P ϵ (Z^*_n).
3. Proses : $P = [C2 \times (C1^d)^{-1} \bmod p$.

2. Gambaran Perangkat Lunak

2.1. Aliran Proses

2.1.1. Use Case Diagram

Use case diagram menggambarkan hubungan antara aktor dengan sistem yang akan dibuat.



Gambar 1. Use Case Diagram

Tabel 1 Use Case Scenario Pembangkit Kunci

Nama Use Case	Pembagkit Kunci
Tujuan	*System* membangkitkan kunci publik dan kunci privat.
Kondisi Keberhasilan	Terbentuk kunci publik *p, α, β* dan kunci privat *d.*
Kondisi Gagal	Kunci gagal terbentuk.
Actor Utama	*System.*
Pemicu	*User* menekan tombol kirim untuk mengirim *email*.
Aliran Proses Utama	1. Pilih bilangan prima *p*, dimana $255 \leq p \geq 1000$. 2. Pilih *α* sebagai akar primitif pada grup (Z^*_p). 3. Pilih d yang memenuhi $1 \leq d \geq p\text{-}2$. 4. Hitung $\beta = \alpha^d \bmod p$. 5. Terbentuk Kunci Publik *p, α, β.* 6. Terbentuk Kunci Privat *d*. 7. Simpan kunci yang terbentuk ke dalam *database* untuk keperluan dekripsi.

Tabel 2 Use Case Scenario Enkripsi Data

Nama Use Case	Enkripsi Data
Tujuan	*System* melakukan proses enkripsi pada isi *email* yang dikirimkan user.
Kondisi Keberhasilan	*Output* yang disimpan pada database merupakan *chipertext* hasil proses enkripsi.
Kondisi Gagal	*System* gagal merubah isi *email* menjadi *chipertext*.
Actor Utama	*System.*
Pemicu	*User* menekan tombol kirim pada saat akan mengirim *email*.
Use Case yang terlibat	Pembangkit Kunci.

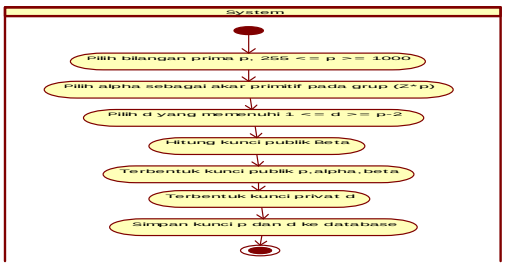
Nama Use Case	Enkripsi Data
Aliran Proses Utama	1. Include::Pembangkit Kunci 2. Memparsing tiap – tiap karakter yang ada pada isi *email*. 3. Konversi tiap karakter ke dalam notasi desimal. 4. Bangkitkan bilangan acak *r*, $1 \leq r \geq p$. 5. *Chipertext* pertama, $C1 = \alpha^r \bmod p$. 6. *Chipertext* kedua, $C2 = (P \times \beta^r) \bmod p$. 7. Gabungkan kembali *chipertext* dari masing – masing karakter menjadi sebuah teks. 8. Simpan teks yang utuh dan merupakan *chipersteks* ke dalam database.

Tabel 3 Use Case Scenario Dekripsi Data

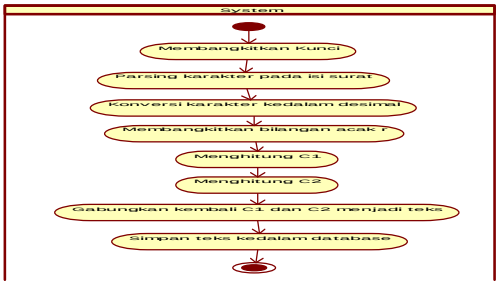
Nama Use Case	Dekripsi Data
Tujuan	*System* mengubah isi surat yang berupa *chipertext* menjadi *plaintext.*
Kondisi Keberhasilan	*System* menampilkan *plaintext* dari isi surat melalui proses dekripsi.
Kondisi Gagal	*System* gagal melakukan dekripsi pada *chipertext*.
Actor Utama	*System*.
Pemicu	*System* memvalidasi nomor *PIN* yang diinput oleh *user* adalah benar.
Aliran Proses Utama	1. Include::Cek PIN 2. *System* mengambil kunci privat dan kunci publik pada *database*. 3. *System* memparsing isi surat menjadi *chipertext* yang masing-masing terdiri dari 2 bilangan integer *C1* dan *C2*. 4. Plaintext $P = \{C2 \times (C1^{d-1})\} \bmod p$ 5. Konversi bilangan integer *P* menjadi karakter ASCII. 6. Susun kembali karakter ASCII tersebut menjadi sebuah teks.

2.2. Activity Diagram

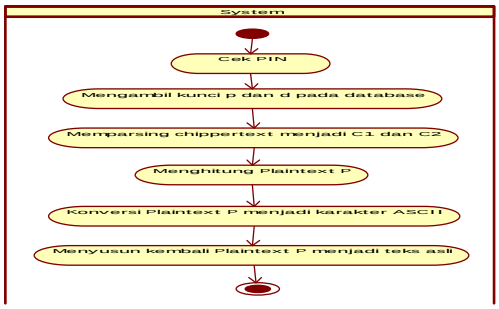
Activity Diagram menggambarkan aliran proses (workflow) yang terjadi pada aplikasi Email dengan Sistem Kriptografi Elgamal.



Gambar 2 Activity Diagram Pembangkit Kunci



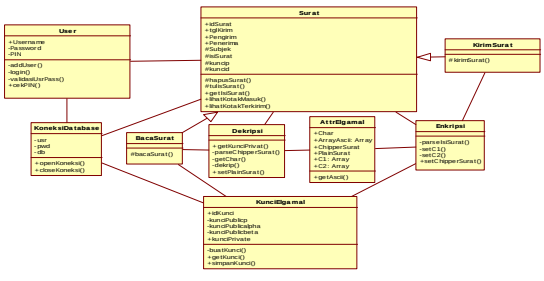
Gambar 3 Activity Diagram Enkripsi Data



Gambar 4 Activity Diagram Dekripsi Data

2.3. Class Diagram

Class diagram menggambarkan objek – objek yang terdapat pada sistem dan hubungan antara objek – objek tersebut.

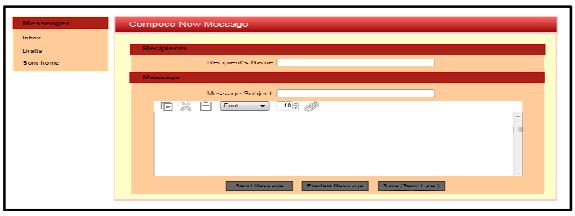


Gambar 5 Class Diagram

2.4. Perancangan Antarmuka

Perancangan antarmuka ini bertujuan untuk memberikan gambaran mengenai bentuk antarmuka

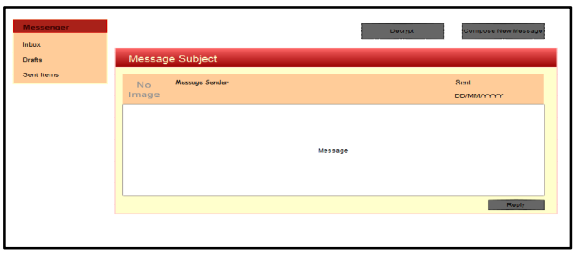
dari perangkat lunak yang akan digunakan oleh user untuk berinteraksi dengan perangkat lunak.



Gambar 6 Antarmuka Kirim Surat

Uraian Cara Penggunaan :

a. *User* menginputkan alamat tujuan *email* pada kotak isian "*Recipient'sName*".
b. *User* menginputkan subjek atau judul *email* pada kotak isian "*MessageSubject*".
c. *User* menginputkan isi surat pada *textarea* yang disediakan.
d. Untuk mengirim *email*, *user* menekan tombol "*SendMessage*".
e. Setelah menekan tombol "*SendMessage*" akan muncul *dialogscreen* "Email telah Terkirim" dan pesan akan dienkripsi.



Gambar 7 Antarmuka Baca Surat

Uraian Cara Penggunaan :

a. Tombol "*Reply*" digunakan untuk membalas surat.
b. Secara *default* isi surat berupa teks sandi atau *chipertext*, untuk mengubahnya menjadi teks asli, user menekan tombol "*Decrypt*". Maka akan muncul *dialogscreenInsertPIN* untuk mendekripsi surat.
c. *System*akan mengubah Isi *email* menjadi plainteks setelah proses dekripsi.

3. Implementasi

Sub bab ini akan menjelaskan langkah-langkah serta rencana jadwal dalam rangka mengimplementasikan aplikasi email dengan Sistem Kriptografi Elgamal yang telah dirancang pada bab sebelumnya.

Berikut adalah rangkaian aktifitas-aktifitas implementasi yang akan dilakukan :

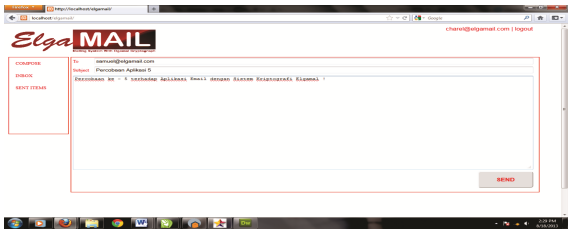
1. Analisis kebutuhan perangkat lunak
   Tahapan analisa spesifikasi perangkat lunak yang akan digunakan oleh sistem.
2. Pemilihan *Hardware* dan *Software*
   Tahapan pemilihan terhadap perangkat keras dan perangkat lunak yang akan digunakan oleh sistem.
3. Instalasi *Hardware* dan *Software*
   Tahapan instalasi terhadap *Hardware* dan *Software* yang sudah dipilih sebelumnya untuk pembuatan sistem.
4. Design Sistem Perangkat lunak
   Tahapan perancangan terhadap sistem yang akan digunakan.
5. Implementasi *coding*
   Merupakan tahapan pembuatan perangkat lunak menggunakan bahasa pemrograman tertentu.
6. Pengujian perangkat lunak
   Tahapan pengujian terhadap perangkat lunak, mulai dari algoritma, fungsi, dan tujuan.
7. Evaluasi dan Perbaikan Perangkat Lunak
   Tahapan memperbaiki sistem bila terdapat kesalahan dan kekurangan yang didapat dari tahapan pengujian perangkat lunak.

3.1 Lingkup dan Batasan Implementasi

Ruang lingkup dan batasan implementasi terdiri dari:

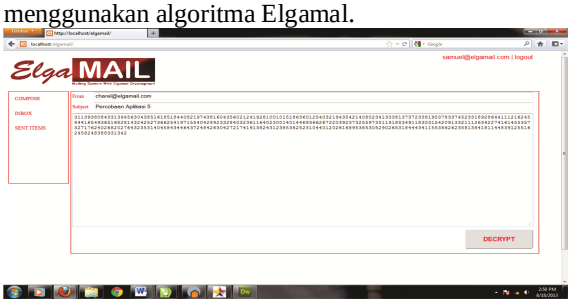
1. Sistem memiliki *software* pendukung seperti *softwarewebservice* (*XAMPP*), kemudian *WebBrowser* (*MozillaFirefox*, *GoogleChrome*).
2. Perangkat lunak yang dikembangkan adalah Aplikasi Email dengan Sistem Kriptografi Elgamal..
3. Aplikasi Email dengan Sistem Kriptografi Elgamal yang dikembangkan merupakan aplikasi *email* yang dapat melakukan pengiriman *email* dari satu akun ke akun lain.
4. Pembuatan kunci Sistem Kriptografi Elgamal dilakukan oleh sistem secara otomatis, sehingga user tidak perlu melakukan proses pembuatan kunci.
5. Pada saat proses enkripsi dan dekripsi isi *email*, kunci diberikan otomatis oleh sistem.

3.2. Implementasi Antarmuka



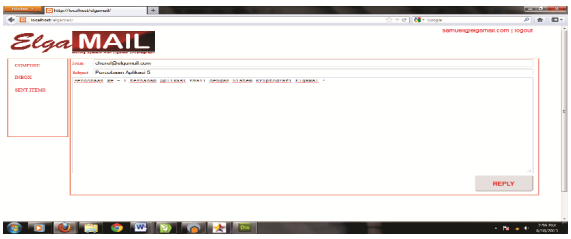
Gambar 11 Antarmuka Kirim Surat

Gambar 11 menampilkan form untuk mengirim email ke akun lain. Isi email akan otomatis dienkripsi menggunakan algoritma Elgamal.



Gambar 12 Antarmuka Baca Surat Chippertext

Gambar 12 menampilkan isi email yang dibaca oleh user. Secara default isi email berupa chippertext hasil enkripsi menggunakan algoritma elgamal.



Gambar 13 Antarmuka Baca Surat Plaintext

Gambar 13 menampilkan isi surat yang sudah didekripsi menggunakan algoritma Elgamal sehingga menghasilkan plaintext.

4. Hasil Pengujian

Berikut ini adalah hasil pengujian dari identifikasi – identifikasi masalah yang dijelaskan pada bab sebelumnya

4.1. Hasil pengujian Pembangkitan Kunci Publik dan Kunci Privat pada Sistem Kriptografi Elgamal.

Prosedur Pembangkitan Kunci Sistem Kriptografi Elgamal :

1. Kunci publik pertama (p) merupakan bilangan prima antara 255 sampai 1000.
2. Kunci privat d didapat dengan cara menentukan secara acak dari bilangan 1 hingga $p - 2$.
3. Kunci publik kedua (α) merupakan elemen primitif dari Z^*_pdimana p adalah kunci publik p. Untuk mencari kunci publik α digunakan algoritma seperti berikut :
   A. α dikatakan elemen primitif Z^*_p bila$\alpha^{\{(p-1)/2\}} mod\ p$ tidak sama dengan 1.
   B. α dimulai dari bilangan 2, 3, 4, 5 dan seterusnya hingga hasil perhitungan $\alpha^{\{(p-1)/2\}} mod\ p$ tidak sama dengan 1.
4. Kunci publik ketiga (β) merupakan hasil perhitungan dengan rumus :
   $\beta = \alpha^d\ mod\ p$

Tabel 5 Pasangan kunci Sistem Kriptografi Elgamal

Kunci Publik p	Kunci Publik α	Kunci Publik β	Kunci Privat d
271	13	102	263
809	3	235	219
719	11	333	636
863	5	464	370
911	7	343	199

Contoh Hasil Pembentukan Kunci Sistem Kriptografi Elgamal.

4.2. Hasil pengujian Proses Enkripsi pada Sistem Kriptografi Elgamal

Gunakan kunci p = 863, α = 5, β = 464, dan d = 370 untuk melakukan enkripsi. Pemilihan pesan yang akan dienkripsi adalah dipilih yang mengandung angka, huruf kapital dan huruf kecil, tanda baca atau simbol. Pesan yang akan dienkripsi adalah : “P@sSw0rd!”. Berikut ini adalah proses enkripsi terhadap pesan tersebut:

Tabel 6 Proses Enkripsi Sistem Kriptografi Elgamal

No	Karakter (T)	Desimal	r	C1 = α^r mod p	C2 = (P x β^r) mod p
1	P	80	120	751	709
2	@	64	502	24	254
3	S	115	64	660	51
4	S	83	271	434	777

Sehingga menghasilkan chippertext :751709242546605143477 7

4.3. Hasil Pengujian Proses Dekripsi pada Sistem Kriptografi Elgamal

Tabel 7 Proses Dekripsi Sistem Kriptografi Elgamal

No	C1	C2	P = {C2 x $C1^{(p-d-1)}$} mod p	Karakter
1	751	709	80	P
2	24	254	64	@
3	660	51	115	S
4	434	777	83	S

terbukti bahwa setiap *chipertext* yang didekripsi kembali menjadi *plaintext*. Kunci yang digunakan adalah kunci sebelumnya, yaitu kunci p = 863, α = 5, β = 464, dan d = 370. Kunci d adalah kunci privat yang sifatnya rahasia, karena kunci d digunakan

untuk melakukan proses dekripsi pada Sistem Kriptografi Elgamal.

4.4. Hasil Pengujian Untuk Mengetahui Besarnya Perubahan Data Setelah Proses Enkripsi.

Tabel 8 Perubahan Data Sebelum dan Sesudah Proses Enkripsi

No	Besar Data Sebelum Enkripsi (Planitext)	Besar Data Setelah Enkripsi (Chippertext)	Persentase Perubahan Data
1	50 Byte	284 Byte	568 %
2	100 Byte	564 Byte	564 %
3	200 Byte	1151 Byte	575,5 %
4	300 Byte	1470 Byte	490 %
5	400 Byte	2239 Byte	559,75 %
Rata – rata perubahan besar data			554,9545 %

4.5. Hasil Pengujian Untuk Mengetahui Waktu yang Dibutuhkan Sistem Untuk Melakukan proses Enkripsi dan Dekripsi.

Tabel 9 Waktu yang Dibutuhkan Untuk Melakukan Proses Enkripsi dan Dekripsi.

No	Besarnya data	Waktu Enkripsi	Waktu Dekripsi
1	50 Byte	0.000838 seconds	0.00046 seconds
2	100 Byte	0.001444 seconds	0.00068 seconds
3	200 Byte	0.002524 seconds	0.00267 seconds
4	300 Byte	0.002769 seconds	0.00180 seconds
5	400 Byte	0.004503 seconds	0.00479 seconds

Setelah melakukan pengujian terhadap sistem makan dapat disimpulkan bahwa :

1. Pembentukan Kunci Sistem Kriptografi Elgamal adalah sebagai berikut :
   a. Pilih p yang merupakan bilangan prima, dimana $p > 255$.
   b. Pilih α sebagai akar primitif dari grup p, Perhitungan akar primitif α adalah :$z = \alpha^{(p-1)/2} \bmod p$. jika z tidak sama dengan 1 maka α adalah akar primitif pada grup p.
   c. Pilih d, dimana $1 \leq d \geq p - 2$.
   d. Hitung $\beta = \alpha^d \bmod p$.
   e. Kunci publik (p, α, β), Kunci privat (d).
2. Proses Enkripsi dan Dekripsi pada Sistem Kriptografi Elgamal adalah sebagai berikut :
   a. Proses Enkripsi pada sebuah karakter plaintext menghasilkan dua buah bilangan desimal (*C1* dan *C2*) yang merupakan *chipertextnya*. Berikut ini adalah proses enkripsi pada sebuah *plaintext* :
      1) Ubah plaintext *P* menjadi bilangan desimal sesuai pada tabel ASCII.
      2) Tentukan secara acak bilangan r, dimana $1 \leq r \geq p$.
      3) $C1 = \alpha^r \bmod p$.
      4) $C2 = (P \times \beta^r) \bmod p$.
      5) Didapatkan *chipertextC1* dan *C2*.
   b. Proses Dekripsi menggunakan kunci privat d untuk mengubah chippertext *C1* dan *C2* kembali menjadi *plaintext*. Berikut ini adalah rumus perhitungan dekripsi pada Sistem Kriptografi Elgamal :
      1) $P = \{C2 \times C1^{(p-d-1)}\} \bmod p$.
      2) *P* merupakan bilangan desimal, ubah kembali menjadi karakter sesuai pada tabel ASCII.
3. Rata – rata perubahan besarnya data dari *plaintext* yang melalui proses enkripsi dan menjadi *chipertext* adalah 554,9545 %.
4. Waktu yang dibutuhkan untuk sebuah sistem untuk melakukan proses enkripsi dan dekripsi Sistem Kriptografi Elgamal hingga 400 karakter adalah 0.004503 detik untuk proses enkripsi dan 0.00479 detik untuk proses dekripsi.

Saran atau masukan yang dapat kami berikan untuk menunjang atau pengembangan sistem selanjutnya, sebagai berikut:

1. Pada penelitian selanjutnya diharapkan batas maksimum bilangan prima yang digunakan untuk pembentukan kunci publik p lebih dari 1000.
2. Diharapkan penerapan Sistem Kriptografi Elgamal juga untuk mengenkripsi dan mendekripsi gambar, *file*, dan *video*.
3. Penggunaan bahasa pemrograman yang lain seperti C#, .NET, dan lain - lain.
4. Penerapan Algoritma Kriptografi Elgamal diterapkan pada sistem lainnya seperti transaksi *online*, media komunikasi seperti *handphone*, dan sebagainya untuk penelitian berikutnya.

DAFTAR PUSTAKA

1. Hamilton Kim dan Miles Russel, 2006, *UML 2.0,* Oreilly Media, Sebastop
2. Munir, Rinaldi. 2012, *Matematika Diskrit,* Informatika, Bandung
3. Sadikin, Rifki. 2012, *Kriptografi untuk keamanan jaringan,* Andi, Yogyakarta
4. Tatroe, et all. 2003, *Programming PHP Third Edition,* Oreilly Media, Sebastopol