Analisis dan Desain Algoritma Hybrid Kriptografi untuk Manajemen Strategi Pengamanan Data Perusahaan

Analysis and Design of Hybrid Cryptography Algoritgm for Management of Company's Data Security Strategy

Ilham
Program Studi Sistem Informasi, Jurusan Teknologi, Fakultas Saintek,
Universitas Islam Negeri Sunan Ampel Surabaya
Jl. Ahmad Yani 117, Surabaya 61121, Jatim
Email: ilham@uinsby.ac.id

Abstrak— Penelitian ini dilakukan untuk mencoba mengimpelementasi suatu algoritma hybrid guna melakukan pengamanan dokumen suatu perusahaan. Dengan meningkatnya serangan cybercrime pada perusahaan-perusahaan terutama berkaitan dengan data dan dokumen penting, menuntut perusahaan untuk lebih meningkatkan kewaspadaan dan pengamanan pada data atau dokumen mereka. Data perusahaan seharusnya hanya boleh diketahui oleh orang-orang tertentu dalam perusahaan dan tidak diperkenankan orang lain yang tidak berhak untuk mengakses data-data tersebut sehingga diperlukan pengamanan dan perlindungan yang ketat terhadap data data yang digunakan dan harus dijamin dalam batas-batas tertentu. Salah satu metode yang dapat digunakan untuk mengamankan data dengan melakukan enkripsi dan dekripsi adalah algoritma hybrid (RSA dan ElGamal). Hasil dari uji coba pengamanan dokumen ini dengan akurasi rata-rata 85,57%.

Kata Kunci — Dokumen, Enkripsi, Dekripsi, RSA, ElGamal

Abstract— This research was conducted in order to test the implementation of a hybrid algorithm for keeping documents of a company secure. The increase of cybercrime attacks on major companies , related to their data dan important documents, demands companies to increase alertness and the security of their data or documents. Companies data should only be able to be accessed by certain authorized personnel and others should not be given access to them, therefore strict security and protection of those data is needed and those should be guaranteed within certain boundaries. One of the methods that can be utilized for securing data by performing encryption and decryption is hybrid algorithm (RSA and ElGamal). Testing result of securing documents by this method indicates average accuracy of 85.57%.

Keywords : Documents, Encryption, RSA And ElGamal

I. PENDAHULUAN

Kebutuhan data dan informasi menjadi suatu hal yang tidak dapat dipisahkan lagi dari setiap aspek kehidupan. Kemajuan teknologi komputer dan telekomunikasi bagaikan pisau bermata dua. Disatu sisi kita dimudahkan dengan adanya teknologi itu, namun disisi lain aspek kejahatan dengan menggunakan teknologi ini juga semakin meningkat sebagai contoh banyak informasi milik perusahaan yang hanya boleh diketahui oleh orang-orang tertentu dalam perusahaan untuk itu keamanan data yang digunakan harus terjamin dalam batasan tertentu.

Untuk itu dibutuhkan suatu aplikasi yang dapat mengenkripsi dan mendekripsi suatu data atau dokumen. Algoritma RSA merupakan salah satu teknik mengamankan data dengan cara melakukan enkripsi dan dekripsi menggunakan 1 pasang kunci yaitu kunci publik dan kunci privat.

Teknik ini sangat membantu proses pengamanan data, karena hanya orang yang mempunyai kunci privat saja yang dapat menguraikan isi file tersebut. Sama halnya dengan algoritma ElGamal yang merupakan salah satu algoritma kriptografi kunci publik yang pada umumnya digunakan untuk digital signature yang kemudian bisa digunakan untuk melakukan enkripsi dan dekripsi.

Aplikasi pengamanan dokumen menggunakan algoritma kriptografi asimetris *RSA* dan *ElGamal* diharapkan dapat membantu dalam menjaga keamanan dan kerahasiaan dokumen-dokumen penting sehingga tidak dapat diketahui oleh orang lain.

II. TINJAUAN PUSTAKA

A. RSA (Rivest-Samir-Adleman)

Algoritma *RSA* adalah sebuah algoritma yang bekerja per blok data yang mengelompokkan plainteks menjadi blok-blok terlebih dahulu sebelum dilakukan enkripsi sehingga menjadi cipherteks.

Langkah-langkah pembuatan kunci algoritma *RSA* yang harus dilakukan adalah sebagai berikut:

1) Pilih dua bilangan prima p dan q dimana $p \neq q$
2) Hitung nilai n dengan persamaan

$$n = p \times q \quad (1)$$

Dan hitung juga ϕn dengan persamaan

$$\phi(n) = (p-1) \times (q-1) \quad (2)$$

3) Menentukan nilai *e* secara acak, $1 < e < \phi$ dan prima relatif terhadap ϕ. Prima relatif maksudnya jika faktor persekutuan terbesar (*FPB*) dari *e* dan ϕ sama dengan satu ($FPB(e, \phi) = 1$). Dan hitung bilangan bulat positif *d* yang unik, $1 < d < \phi$ dimana $d = e^{-1} \bmod \phi$ sehingga $e.d = 1 \pmod{\phi}$ atau $e.d = k(\phi) + 1$ untuk suatu bilangan bulat *k*.
4) Didapat kunci publik berupa pasangan *(n, e)* dan kunci privat *(n, d)* dimana *e* adalah kunci publik, *d* adalah kunci privat dan *n* adalah modulus.

Adapun langkah-langkah yang dilakukan dalam melakukan proses enkripsi algoritma *RSA* adalah sebagai berikut :

a. Menggunakan kunci public berupa pasangan *((e, n)*.
b. Representasi pesan atau data menjadi blok-blok dan diubah menjadi bilangan bulat positif *M*
c. Hitung nilai C dengan persamaan :

$$C = M^e \bmod n \quad (3)$$

d. Dan dihasilkan cipherteks (C) yang merupakan hasil dari enkripsi

Adapun langkah-langkah dalam melakukan proses dekripsi dengan algoritma RSA adalah sebagai berikut :

1) Menggunakan kunci privat yang berupa pasangan *(d, n)*.
2) Representasi pesan atau data menjadi blok-blok dan diubah menjadi bilangan bulat positif *C*
3) Hitung nilai M dengan persamaan

$$M = C^d \bmod n \quad (4)$$

4) Dan dihasilkan plainteks (M) yang merupakan hasil dari dekripsi dan merupakan pesan atau data yang sebenarnya.

B. *Algoritma ElGamal*

Sama halnya dengan algoritma *RSA*, algoritma *ElGamal* juga sebuah algoritma yang bekerja per blok data yang mengelompokkan plainteks menjadi blok-blok sebelum dilakukan enkripsi sehingga menghasilkan cipherteks. Langkah-langkah yang dilakukan dalam pembuatan kunci algotima *ElGamal* adalah sebagai berikut :

1) Menentukan bilangan prima p
2) Mennentukan bilangan g dan x dimana $g < p$ dan $1 \leq x \leq p-2$.
3) Hitung nilai y dengan persamaan

$$y = g^x \bmod p \quad (5)$$

4) Kunci publik yang dihasilkan adalah {p, g, y} dan kunci privat {p, x}

Adapun langkah-langkah yang dilakukan dalam proses enkripsi algoritma *ElGamal* adalah sebagai berikut :

1) Menggunakan sebuah kunci publik {p, g, y}
2) Representasi data menjadi blok-blok sehingga diperoleh bilangan bulat M.
3) Menentukan bilangan acak *k* dimana $1 \leq k \leq p-1$
4) Hitung nilai a dengan persamaan

$$a = g^k \bmod p \quad (6)$$

5) Dan hitung nilai b dengan persamaan

$$b = y^k M \bmod p \quad (7)$$

6) Diperoleh ciphertext untuk karakter M tersebut dalam blok (a, b)

Adapun langkah-langkah dalam melakukan proses dekripsi atau mengembalikan seperti semula algoritma *ElGamal* adalah sebagai berikut

1) Menggunkan kunci privat {p, x}
2) Representasi data menjadi blok-blok sehingga diperoleh bilangan bulat C.
3) Hitung nilai $(a^x)^{-1}$ dengan persamaan

$$(a^x)^{-1} = a^{p-1-x} \bmod \rho \quad (8)$$

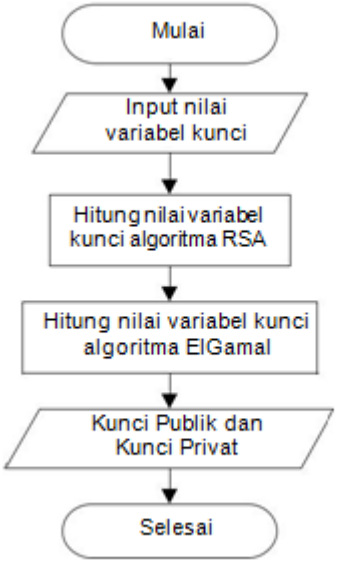
4) Hitung nilai M dengan persamaan

$$m = b/a \bmod \rho = b(a^x)^{-1} \bmod \rho \quad (9)$$

5) Dan dihasilkan plainteks (M) yang merupakan hasil dari dekripsi dan merupakan pesan atau data yang sebenarnya.

III. DESAIN SISTEM

A. *Proses Pembuatan Kunci*



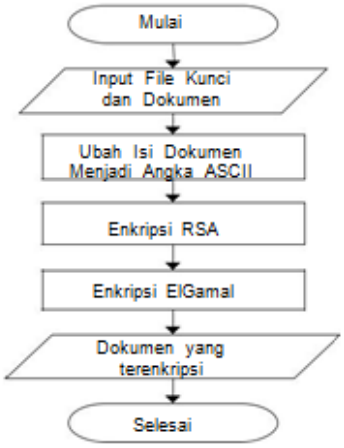
Gambar 1. Flowchart Pembangkitan Kunci

Secara umum proses pembuatan kunci dalam penelitian ini telah digambarkan pada flowchart sistem (Gambar 3.1) yaitu dengan memasukkan inputan variabel-variabel kunci yang digunakan Untuk membuat kunci publik dan kunci privat algoritma *RSA* dan *ElGamal*, kemudian dilakukan

perhitungan variabel-variabel pembentukan kunci *RSA* dan kemudian dilakukan perhitungan variabel-variabel pembentukan kunci *ElGamal* sehingga diperoleh kunci publik dan kunci privat.

B. Proses Enkripsi Dokumen

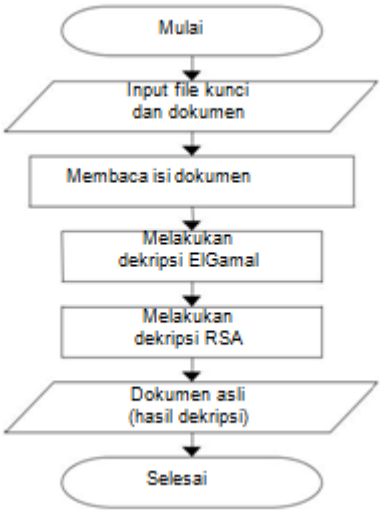
Adapun langkah-langkah dalam melakukan enkripsi dokumen dijelaskan dalam flowchart pada Gambar 2 berikut



Gambar 2. Flowchart Enkripsi Dok.

Yang pertama dilakukan dalam proses enkripsi ini adalah memilih dokumen dan file kunci prublik yang telah dibuat sebelumnya, kemudian mengubah isi dokumen menjadi karakter-karakter ASCII dan kemudian dilakukan proses enkripsi dengan algoritma *RSA* dan dilanjutkan dengan proses enkripsi algoritma *ElGamal* sehingga diperoleh dokumen yang telah terenkripsi hasil dari enkripsi kedua metode tersebut.

C. Proses Deskripsi Dokumen.



Gambar 3. Flowchart Dekripsi Dokumen

Adapun penjelasan flowchart (Gambar 3) proses dekripsi dokumen dilakukan dengan memilih dokumen yang telah terenkripsi dan sebuah file kunci privat yang merupakan pasangan file kunci publik yang digunakan untuk mengenkripsi dokumen. Selanjutnya membaca isi dokumen yang kemudian dilakukan proses dekripsi dengan algoritma *ElGamal* dan dilanjutkan dengan proses dekripsi algoritma *RSA* sehingga dihasilkan dokumen yang sebenarnya (dokumen sebelum dilakukan enkripsi)

IV. PENGUJIAN SISTEM

A. Pembuatan Kunci

Pembuatan kunci ini digunakan untuk melakukan membuat file kunci publik dan file kunci privat yang nantinya akan digunakan untuk melakukan proses enkripsi dan proses dekripsi dokumen.

B. Halaman Enkripsi Dokumen

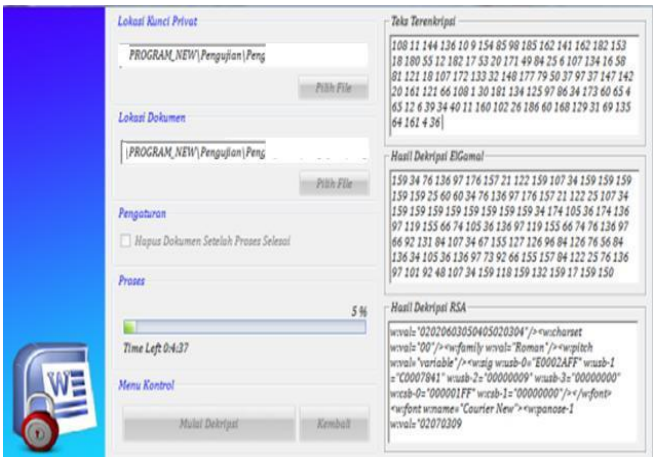
Halaman enkripsi dokumen ini merupakan halaman yang digunakan untuk melakukan enkripsi terhadap dokumen. Dibutuhkan file kunci public yang dibuat pada halaman pembuatan kunci untuk melakukan enkripsi dokumen. Dokumen hasil dari enkripsi dokumen mempunyai ekstensi yang sama dengan dokumen aslinya. Tampilan halaman enkripsi dokumen dapat dilihat pada Gambar 4 berikut

Gambar 4.Halaman Enkripsi Dokumen

C. Halaman Dekripsi Dokumen

Halaman dekripsi dokumen ini merupakan halaman yang digunakan untuk mendekripsi atau mengembalikan dokumen yang telah terenkripsi menjadi dokumen aslinya atau seperti dokumen sebelum dilakukan proses enkripsi. Dibutuhkan file kunci privat untuk melakukan proses dekripsi ini, file kunci privat yang digunakan merupakan pasangan dari file kunci publik yang digunakan untuk

mengenkripsi dokumen. Adapun tampilan dari halaman dekripsi dokumen ini dapat dilihat pada Gambar 5 berikut.



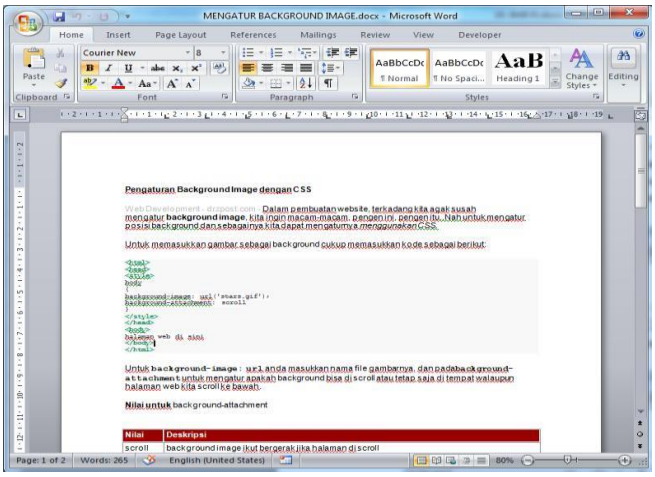
Gambar 5..Halaman Deskripsi Dokumen

D. *Pengujian Sistem*

Pengujian sistem dilakukan dengan cara menguji keakuratan dalam pengembalian dokumen seperti aslinya setelah dilakukan enkripsi. Dokumen yang diujikan adalah dokumen yang mempunyai ekstensi doc dan docx.

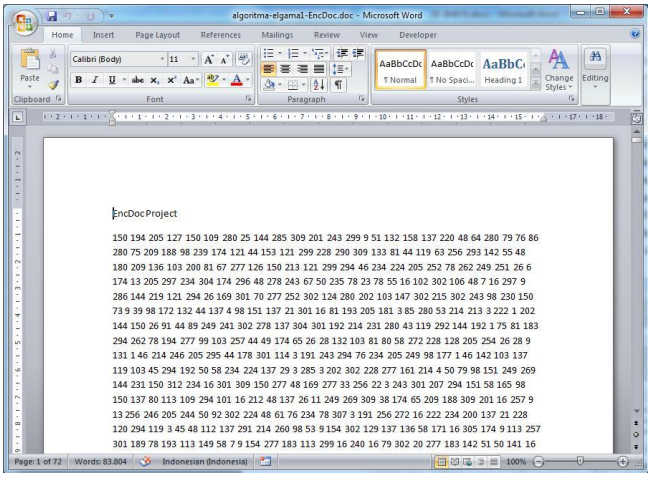
Pengujian Dengan Dokumen Berekstensi docx

Dalam pembuatan kunci pada pengujian ini diinputkan variabel kunci RSA dengan nilai p = 23, q = 7, e = 5 dan d = 53 sedangkan variabel kunci ElGamal yang diinputkan adalah p = 191, g = 3, x = 5 dan y = 52 sehingga file kunci publik yang diperoleh akan berisikan {5, 161, 191, 3, 52} dan file kunci privat yang diperoleh berisikan {53, 161, 191, 5}. Konten isi dari dokumen yang diuji antara lain yaitu variasi teks seperti jenis font teks yang berbeda, warna teks yang berbeda, ukuran teks yang berbeda, tabel yang mempunyai latar belakang yang berwarna, Adapun tampilan dokumen yang digunakan dalam pengujian dapat dilihat pada Gambar 6. berikut



Gambar 6. Dok. Asli Berekstensi docx yang Diuji

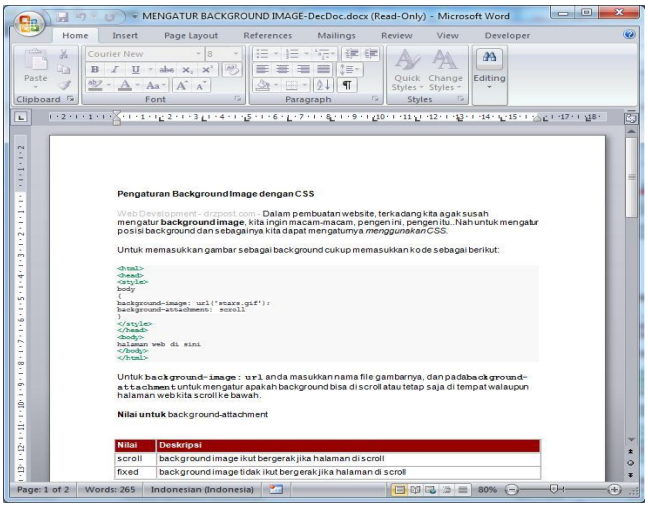
Setelah dokumen tersebut dilakukan proses enkripsi menggunakan kunci publik yang telah dibuat menghasilkan dokumen yang mempunyai ekstensi yang sama yaitu docx. Pada proses enkripsi ini isi dari dokumen asli diubah menjadi angka-angka. Banyaknya angka-angka dan waktu estimasi dalam melakukan proses enkripsi bergantung pada panjangnya kode XML yang terdapat dalam dokumen. Tampilan dokumen hasil enkripsi dapat dilihat pada Gambar 7.



Gambar 7. Dok.Berekstensi docx Hasil Enkripsi

Untuk mengembalikan dokumen tersebut menjadi seperti semula maka dilakukan proses dekripsi dengan menggunakan file kunci privat yang merupakan pasangan dari file kunci publik yang digunakan untuk mengenkripsi dokumen.

Jika file kunci privat tersebut bukan merupakan pasangan dari file kunci publik yang digunakan untuk mengenkripsi dokumen maka dokumen hasil dekripsi tidak bisa tersimpan karena tidak akan menghasilkan kode XML yang benar. Adapun tampilan dokumen hasil dari proses dekripsi dari dokumen yang telah dienkripsi (Gambar 7) dapat dilihat pada Gambar 8 berikut.



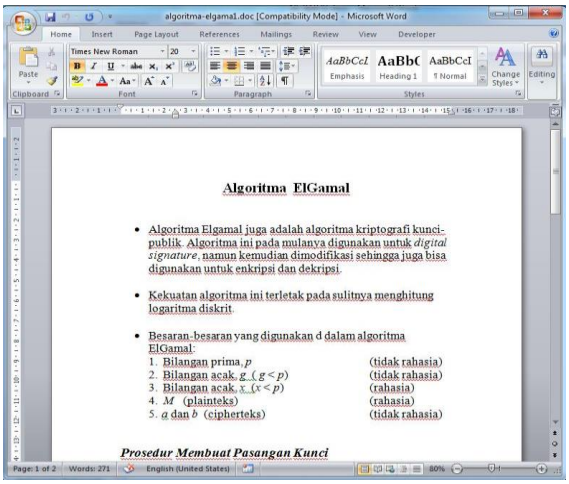
Gambar 8. Dok.Berekstensi Docx Hasil Dekripsi

Dari hasil dekripsi pada Gambar 4.5 tersebut jika dibandingkan dengan dokumen aslinya (dokumen sebelum

dilakukan proses enkripsi) pada Gambar 4.3 dapat mengembalikan isi konten dokumen diantaranya variasi teks seperti warna, font dan ukuran serta dapat mengembalikan isi konten dokumen yang berupa tabel.

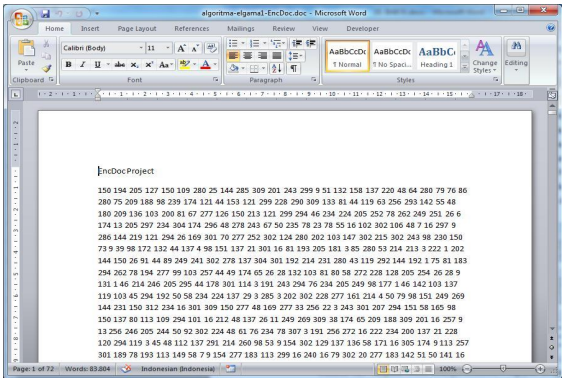
Pengujian Dengan Dokumen Berekstensi doc

Dalam pembuatan kunci pada pengujian ini diinputkan variabel kunci RSA dengan nilai p = 17, q = 13, e = 7 dan d = 55 sedangkan variabel kunci ElGamal yang diinputkan adalah p = 313, g = 3, x = 7 dan y = 309 sehingga file kunci publik yang diperoleh akan berisikan {7, 221, 313, 3, 309} dan file kunci privat yang diperoleh berisikan {55, 221, 313, 7}. Konten isi dari dokumen yang diuji antara lain yaitu variasi teks seperti jenis font teks yang berbeda, ukuran teks yang berbeda, margin serta bullet list. Adapun tampilan dokumen yang digunakan dalam pengujian dapat dilihat pada Gambar 9 berikut



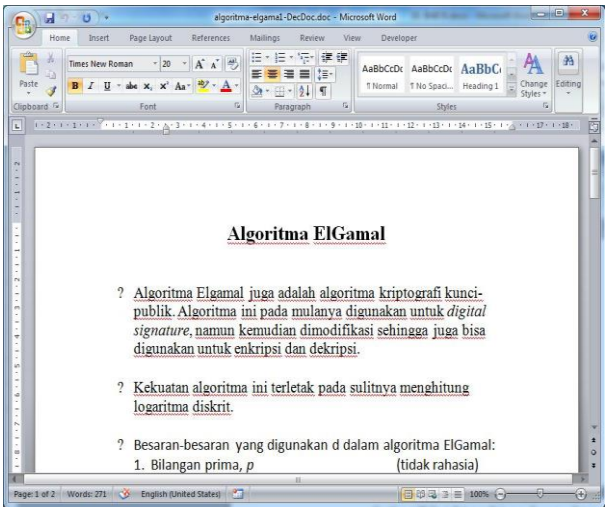
Gambar 9. Dok Asli Berekstensi doc Yang diuji

Setelah dokumen tersebut dilakukan proses enkripsi menggunakan kunci publik yang telah dibuat menghasilkan dokumen yang mempunyai ekstensi yang sama yaitu doc. Pada proses enkripsi ini isi dari dokumen asli diubah menjadi angka-angka. Banyaknya angka-angka dan waktu estimasi dalam melakukan proses enkripsi bergantung pada panjangnya kode XML yang terdapat dalam dokumen. Tampilan dokumen hasil enkripsi dapat dilihat pada Gambar 10 berikut.



Gambar 10. Dok.Berekstensi doc Hasil Enkripsi

Untuk mengembalikan dokumen tersebut menjadi seperti semula maka dilakukan proses dekripsi dengan menggunakan file kunci privat yang merupakan pasangan dari file kunci publik yang digunakan untuk mengenkripsi dokumen. Jika file kunci privat tersebut bukan merupakan pasangan dari file kunci publik yang digunakan untuk mengenkripsi dokumen maka dokumen hasil dekripsi tidak bisa tersimpan karena tidak akan menghasilkan kode XML yang benar. Adapun tampilan dokumen hasil dari proses dekripsi dari dokumen yang telah dienkripsi (Gambar 4.6) dapat dilihat pada Gambar 11 berikut.



Gambar 11. Dok.Berekstensi doc Hasil Dekripsi

Dari hasil dekripsi pada Gambar 4.8 tersebut jika dibandingkan dengan dokumen aslinya (dokumen sebelum dilakukan proses enkripsi) pada gambar 4.6 dapat mengembalikan isi konten dokumen diantaranya variasi teks seperti warna, font namun tidak dapat mengembalikan margin halaman dan simbol bullet list.

Dari hasil pengujian tersebut dapat disimpulkan bahwa keakuratan dalam mengembalikan dokumen yang telah dienkripsi menjadi seperti dokumen aslinya (dokumen sebelum dilakukan proses pengenkripsian) mempunyai tingkat keakuratan rata-rata sebesar 85,57% dihitung berdasarkan jenis konten yang dapat dikembalikan dan jenis konten yang tidak dapat dikembalikan seperti semula.

IV. KESIMPULAN

Berdasarkan hasil pengujian dapat disimpulkan bahwa aplikasi pengaman dokumen menggunakan algoritma kriptografi asimetris *RSA* dan *ElGamal* dapat melakukan enkripsi dan melakukan dekripsi (pengembalian dokumen seperti semula) dengan keakuratan rata-rata sebesar 85,57%.

Konten dokumen yang berbeda antara konten dokumen hasil proses dekripsi dengan dokumen aslinya adalah margin halaman dan gambar simbol bullet list

DAFTAR PUSTAKA

[1] Fachruddin, Y. 2008. *"Sistem Informasi Biro Keuangan dan Dinas Lain Menggunkan XML Web Sercice Dengan Enkripsi RSA",* Surabaya : STIKOM Surabaya

[2] Ifanto, M. 2009. *"Metode Enkripsi dan Dekripsi Dengan Menggunakan Algoritma ElGamal"*, Bandung : Institut Teknologi Bandung.

[3] Kramer, P. 2002. *"Encryption and Decryption with RSA Algorithm Mathematic and The Computer".* Jakarta : Informatika

[4] Kurniawan, Y. 2004. "*Kriptografi : Keamanan Internet dan Jaringan Komunikasi*", Bandung : Informatika

[5] Mujiarto, D. 2012. *"Aplikasi Pengamanan Data Menggunakan Algoritma RSA (Rivest-Shamir- Adleman)",* Universitas Pembangunan Nasional Veteran. Jawa Timur

[6] Munir, R. 2006. *"Kriptografi",* Informatika, Bandung.

[7] Taufik, M. T., Dwiono, W. and Hartanto, T. 2010. "*Penerapan Algoritma Kriptografi ElGamal untuk Pengamanan File Citra*", Purwokerto:UMP Purwokerto.

[8] Triorizka, A. 2010. "*Penerapan Algoritma RSA Untuk Pengamanan Data dan Digital Signature Dengan .NET*", Yogyakarta : STIMIK AMIKOM. Yogyakarta.

[9] Zelvina, A., Efendi, S. & Arisandi, D. 2012. " Perancangan Aplikasi Pembelajaran Kriptografi Kunci Publik ElGamal untuk Mahasiswa" Sumatera Utara : Universitas Sumatera Utara.

[10] http://agcrypt.wordpress.com/2008/02/25/elgamal-algorithm, "*ElGamal Algorithm*". 2008. (accessed October 24, 2013)

[11] http://ilmukomputerindonesia-komputer.blogspot.com/ p/visual-basic-60.html, "*Visual Basic 6.0*". (accessed October 28, 2013)

.