ISSN : 1978-6603

JURNAL Ilmiah SAINTIKOM Sains dan Komputer

APLIKASI KRIPTOGRAFI ASIMETRIS DENGAN METODE *DIFFIE-HELLMAN* DAN ALGORITMA ELGAMAL UNTUK KEAMANAN TEKS

Purwadi[#1], Hendra Jaya[#2], Ahmad Calam[#3]

[#1,2,3] Program Studi Komputer, STMIK Triguna Dharma
Jl. A.H. Nasution No. 73 F-Medan
E-mail: ahmadcalam@ymail.com

Abstrak

Keamanan dokumen merupakan salah satu hal yang sangat penting dalam penukaran data, khusunya pertukaran data didunia maya yang didalamnya terdapat banyak ancaman pada saat proses itu dilakukan. Keamanan data, khususnya untuk dokumen teks bagi suatu organisasi yang mengasumsikan bahwa dokumen tersebut bernilai rahasia (private and confidential). Salah satu aspek keamanan dalam dokumen teks adalah keaslian, bentuk dan isinya harus sesuai dengan yang dimaksud oleh pembuat. Permasalahannya adalah bagaimana cara menggabungkan metode Diffie-Hellman dengan Algoritma Elgamal untuk proses keamanan data teks dan bagaimana menerapkan dan merancang aplikasi kriptografi Asimetris dengan metode Diffie-Hellman dan Algoritma Elgamal untuk mengamankan teks. Untuk menjawab permasalahan tersebut, hasil penelitian ini menjelaskan; Pertama, Diperlukan instalasi aplikasi untuk setiap perangkat yang akan digunakan untuk melakukan enkripsi dan dekripsi, Kedua, Aplikasi yang dibangun dapat menerapkan metode pertukaran kunci Diffie Hellman dan menghasilkan kunci baru dan Aplikasi ini dapat melakukan enkripsi dan dekripsi Elgamal dengan menggunakan kunci yang telah dibangkitkan dengan metode Diffie Hellman

Kata Kunci: Kriptografi Asimetris, Metode Diffie-Hellman, Algoritma Elgamal, Keamanan Teks.

Abstract

Document security is one thing that is very important in the exchange of data, especially data exchange virtual world in which there are many threats when the process is done. Data security, especially for text documents for an organization that assumes that the documents are valuable secret (private and confidential). One aspect of security in a text document is authenticity, form and content shall be in accordance with that intended by the manufacturer. The problem is how to combine the Diffie-Hellman method with Elgamal algorithm to process text data security and how to implement and design applications Asymmetric cryptography with Diffie-Hellman method and algorithm for secure text Elgamal. To answer these problems, this research describes; First, the application installation is required for each device that will be used to perform encryption and decryption, Second, applications are built to implement the Diffie-Hellman key exchange method and generate a new key and this application can do Elgamal encryption and decryption using the key that has been generated by the method Diffie Hellman

Keywords: *Asymmetric cryptography, Diffie-Hellman method, Algorithm Elgamal, Text Security.*

PENDAHULUAN

Informasi pada saat sekarang ini telah menjadi komoditi yang memiliki peranan yang sangat besar. Kemampuan mengakses dan menyediakan informasi secara cepat dan akurat menjadi sangat esensial bagi organisasi, baik berupa organisasi komersial (perusahaan), perguruan tinggi, lembaga pemerintahan, maupun individual (pribadi).

Keamanan dokumen merupakan salah satu hal yang sangat penting dalam penukaran data, khusunya pertukaran data didunia maya yang didalamnya terdapat banyak ancaman pada saat proses itu dilakukan. Keamanan data, khususnya untuk dokumen teks bagi suatu organisasi yang mengasumsikan bahwa dokumen tersebut bernilai rahasia (private and confidential). Salah satu aspek keamanan dalam dokumen teks adalah keaslian, bentuk dan isinya harus sesuai dengan yang dimaksud oleh pembuat.

Semakin banyak data yang diproses dengan komputer, maka ancaman terhadap pengamanan data akan semakin meningkat. Salah satu serangan terhadap keamanan adalah interception. Pihak yang tidak berwenang berhasil mengakses data atau informasi, seranganan ini biasa terjadi pada jaringan LAN atau bahkan pada komputer yang tidak berhubungan jaringan sekalipun.

Cara mengatasi keamanan data digunakan tehnik penyandian, dikenal dengan nama tehnik kriptografi. Tehnik ini disandikan atau dienkripsi menjadi data yang tidak dapat dibaca. Data rahasia yang telah dienkripsi dan diterima oleh penerima dapat diubah lagi atau di dekripsi ke data asli sehingga dapat dipahami.

Pada penelitian kali ini dirancang aplikasi kriptografi Asimetris dengan menggunakan metode Diffie-Hellman dan algoritma Elgamal untuk keamanan data teks. Pada penggabungan antara dua metode yaitu Diffie-Hellman dan algortima ElGamal, algoritma Diffie-Hellman akan digunakan sebagai pembangkit kunci, dengan kata lain dalam kasus ini akan terjadi pertukaran kunci, dimana pengirim dan penerima akan memiliki angka-angka rahasia tertentu kemudian saling bertukar untuk dihitung dan mendapatkan kunci publik dan kunci rahasia. Setelah memperoleh kunci publik dan kunci rahasia akan dilanjutkan dengan proses perhitungan untuk enkripsi dan dekripsi dengan menggunakan algoritma ElGamal.

LANDASAN TEORITIS

2.1 Keamanan Komputer

Masalah keamanan merupakan salah satu aspek terpenting dari sebuah sistem informasi. Masalah keamanan sering kurang mendapat perhatian dari para perancang dan pengelola sistem informasi. Masalah keamanan sering berada diurutan setelah tampilan, atau bahkan diurutan terakhir dalam daftar hal-hal yang dianggap penting. Apabila mengganggu performansi sistem masalah keamanan sering tidak diperdulikan, bahkan ditiadakan. Informasi menentukan hampir setiap elemen dari setiap kehidupan manusia. Informasi sangat penting artinya bagi kehidupan karna tanpa informasi maka hampir semua tidak dapat dilakukan dengan baik. Contohnya jika kita membeli tiket penerbangan dan membayarnya dengan menggunakan kartu kredit, informasi mengenai diri kita akan disimpan dan di kumpulkan serta digunakan oleh Bank dan penerbangan. Hanya sedikit hal yang bias dilakukan di dunia modern tanpa melibatkan pengumpulan, penukaran,pembuatan atau pengaksesan informasi. Saat ini informasi

sudah menjadi komoditas yang sangat penting. Bahkan ada yang mengatakan bahwa kita sudah berada disebuah information/based society.

Kemampuan untuk mengakses dan menyediakan informasi secara cepat dan akurat menjadi sangat esiensial bagi sebuah organisasi, baik organisasi komersial (perusahaan), perguruan tinggi, lembaga pemerintahan, maupun organisasi social.

2.2 Kriptografi

Pengertian Kriptografi

Kriptografi berasal dari bahasa Yunani, Menurut bahasa tersebut kata kriptografi dbagi menjadi dua, yaitu“kryptos” dan“graphein”. Kryptos berarti secret (rahasia) dan graphein berarti writing (tulisan). Menurut Teminologinya Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari sutau tempat ke tempat yang lain. Dalam perkembangannya, kriptografi digunakan untuk mengidentifkasikan pengiriman pesan dengan tanda tangan digital dan keaslian pesan dengan sidik jari digital (Fingerprint).

Sejarah kriptografi

Kriptografi mempunyai sejarah yang sangat menarik dan panjang. Kriptografi sudah digunakan 4000 tahun yang lalu, diperkenalkan oleh orang-orang Mesir lewat Hieroglyph. Jenis tulisan ini bukanlah bentuk standart untuk menuliskan pesan. Dikisahkan pada zaman Romawi kuno pada suatu saat Julius Caesar ingin mengirimkan pesan rahasia kepada seorang Jenderal di medan perang. Pesan tersebut harus dikirimkan melalui seorang kurir, karena pesan tersebut mengandung rahasia, Julius Caesar tidak ingin pesan rahasia tersebut sampai terbuka dijalan. Julius Caesar kemudian memikirkan bagaimana mengatasinya, ia kemudian mengacak pesan tersebut hingga menjadi suatu pesan yang tidak dapat dipahami oleh siapapun terkecuali oleh jenderalnya saja. Tentu sang jenderal telah diberitahu sebelumnya bagaimana membaca pesan teracak tersebut. Yang dilakukan Julius Caesar adalah mengganti semua susunan alphabet dari A, B, C yaitu A menjadi D, B menjadi E, C menjadi F, dan seterusnya.

Dari ilustrasi tersebut, beberapa istilah kriptografi dipergunakan untuk menandai aktivitas rahasia dalam mengirim pesan. Apa yang dilakukan Julius Caesar yang mengacak pesan, disebut sebagai enkripsi. Pada saat jenderal merapikan pesan yang teracak itu, proses itu disebut dekripsi. Pesan awal yang belum diacak dan pesan yang telah dirapikan disebut plaintext, sedangkan pesan yang telah diacak disebut ciphertext.

Pada zaman Romawi juga telah ada alat pembuat pesan rahasia yang disebut Scytale yang digunakan oleh tentara Sparta. Scytale merupakan suatu alat yang memiliki pita panjang dari daun Papirus dan ditambah dengan sebatang silender. Mula-mula pengirim pesan menuliskan pesannya diatas pita papyrus yang digulung pada batang silender. Setelah itu pita dilepaskan dan dikirim. misalkan batang silender cukup lebar untuk menulis 6 huruf dan bisa memuat huruf huruf secara melingkar. Jika pengirim ingin mengirimkan pesan :

“TOLONG SAYA DISERANG”

Maka ia menulis batang silender:

T O L O N G
S A Y A D I
S E R A N G

Jika pitanya dilepas dari batang silinder, maka tulisan yang muncul tulisan diatas Pita adalah: TSSOAELYROAANDNGIG.

Untuk membaca pesan yang dikirim, penerima pesan harus melilitkan kembali pita tersebut pada batang silinder yang memiliki diameter yang sama. Yang menjadi kunci dalam penyandian Scytale adalah diameter batang atau jumlah huruf yang dapat ditulis secar melingkar (dalam hal ini 3 huruf). Penyandian dengan Scytale sangat mudah dipecahkan karena kriptanalis hanya perlu menerka jumlah huruf yang dapat ditulis secara melingkar pada batang silinder yang digunakan, apalagi karena jumlah huruf yang dapat ditulis secara melingkar pada suatu batang silinder relatif sedikit (maksimum adalah setengah dari jumlah huruf yang tertulis pada pita) kunci hasil deskripsi:

1. TSSOAELYROAANDNGIG
2. TSALRANNISOEYOADGG
3. TOLONGSAYADISERANG

Karena deskripsi dengan k = 3 menghasilkan pesan yang bermakna maka disimpulkan bahwa pesan yang dikirim adalah TOLONG SAYA DISERANG. (Dony Ariyus, 2008: 13-15)

Tujuan Kriptografi

Tujuan dari kriptografi yang juga merupakan aspek keamanan informasi adalah sebagai berikut :

1. **Kerahasiaan (confidentiality)**
   adalah layanan yang digunakan untuk menjadi isi informasi dari semua pihak kecuali pihak yang memiliki otoritas terhadap informasi. Ada beberapa pendekatan unutk menjaga kerahasiaan, dari pengamanan secar fisik hingga penggunaan algoritma matematika yang membuat data tidak dapat dipahami. Istilah lain yang senada dengan confidentiality adalah secrecy dan privacy.

2. **Integritas data**
   adalah layanan penjagaan pengubahan data dari pihak yang tidak berwenang. Untuk menjaga itegritas data, sistem harus memiliki kemampuan unutuk mendeteksi manipulasi pesan oleh pihak-pihak yang tidak berhak, penyisipan, penghapusan dan pensubtitusian data lain kedalam pesan yang sebenarnya. Didalam kritografi, layanan ini direalisasikan dengan menggunakan tanda-tanda digital (Digital Signature). Pesan yang telah ditanda tangani menyiratkan bahwa pesan yang dikirim adalah asli.

3. **Otentikasi**
   adalah layanan yang berhubungan dengan indentifikasi, baik mengiidentifikasi kebenaran pihak-pihak yang berkomunikasi maupun mengidentifikasi kebenaran sumber pesan. Dua pihak yang saling berkomunikasi harus dapat mengotentikasi satu sama lain sehingga ia dapat memastikan sumber pesan. Pesan yang dikirim melalui saluran komunikai juga harus diotentikasi asalnya. Otentikasi sumber pesan secara implisit juga memberikan kepastian integritas data, sebab jika pesan telah dimodifikasi berarti sumber pesan sudah tidak benar. Oleh karena itu, layanan integritas data selalu dikombinasikan dengan layanan ini direalisasikan dengan menggunakan tanda tanda digital (digital signature). Tanda-tanda digital menyatakan sumber pesan.

4. **Nirpenyangkalan (non-repudition)**
   adalah layanan untuk mencegah entitas yang berkomunikasi melakukan penyangkalan, yaitu pengirim pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan.

Algoritma Kriptografi

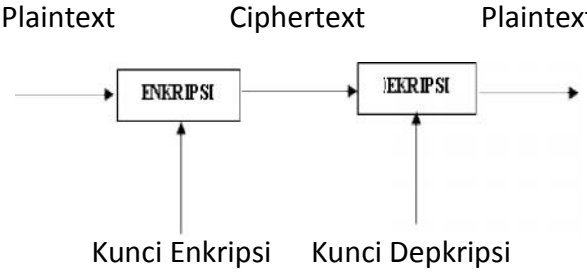
Ditinjau dari asal usulnya, kata algoritma mempunyai sejarah yang sangat menarik. Kata ini muncul di dalam kamus Webster sampai akhir tahun 1957. Kata algorism mempunyai arti proses perhitungan dalam bahasa arab. Algoritma berasal dari nama penulis buku arab yang terkenal, yaitu Abu Ja'far Muhammad Ibnu Musa al-khuwarizmi (al-khuwarizmi dibaca oleh orang barat sebagai algorism). Kata algorism lambat laun berubah menjadi algorithm.

Definisi terminologi adalah urutan langkah-langkah logis untuk menyelesaikan masalah yang disusun secara sistematis. Algoritma kriptografi merupakan langkah-langkah logis bagaimana menyembunyika dari orang-orang yang tidak berhak atas orang-orang tersebut.

Algoritma kriptografi terdiri dari 3 fungis dasar, yaitu :

1. Enkripsi: merupakan hal yang sangat penting dalam kriptografi, merupakan pengamanan data yang dikirimkan agar terjaga kerahasiaannya. Pesan asli disebut Plaintext, yang diubah menjadi kode-kode yang tidak dimengerti. Enkripsi ini diartikan dengan Cipher atau kode. Sama halnya dengan kita tidak mengerti akan sebuah kata maka kita akan melihatnya didalam kamus atau daftar istilah. Beda halnya dengan enkripsi, untuk mengubah teks asli kebentuk teks kode kita menggunakan algortima yang dapat mengkodekan data yang kita inginkan.

2. Deskripsi: merupakan kebalikan dari enkripsi. Pesan yang telah di enkripsi dikembalikan kebentuk asalnya (teks asli), disebut dengan dekripsi pesan. Algoritma yan digunakan untuk deskripsi tentu berbeda dengan algoritma yang digunakan untuk enkripsi.



Gambar 2.1 Diagram proses enkripsi dan dekripsi

3. Kunci: yang dimaksud disini adalah kunci yang dipakai untuk melakukan enkripsi dan dekripsi. Kunci terbagi menjadi dua bagian, kunci rahasia (Private Key) dan kunci umum (Public Key).

Keamanan dari algoritma kriptografi tergantung pada bagaimana algoritma itu bekerja, oleh sebab itu algoritma semacam ini disebut dengan algoritma terbatas. Algoritma terbatas merupakan algoritma yang dipakai sekelompok orang untuk merahasiakan pesan yang mereka kirim. Jika salah satu dari anggota kelompok itu keluar maka algoritma yang dipakai diganti yang baru. Jika tidak maka hal itu biasa menjadi masalah dikemudian hari (Ariyus, 2008: 43).

METODE PENELITIAN

3.1 Analisis Permasalahan

Pembahasan masalah yang terdapat pada bab ini adalah Penggabungan antara dua metode yaitu Diffie-Hellman dan Algoritma ElGamal. Algoritma Diffie-Hellman akan digunakan sebagai pembangkit kunci. Dengan kata lain, dalam kasus ini akan terjadi pertukaran kunci, dimana pengirim dan penerima akan memiliki angka-angka rahasia tertentu kemudian saling bertukar untuk dihitung dan mendapatkan kunci publik dan kunci rahasia. Dibawah ini akan dibahas perhitungan manual algoritma Diffie-Hellman sebagai pembangkit kunci, dan algoritma ElGamal untuk enkripsi dan dekripsi.

3.2 Analisis Algoritma

Untuk menganalisis algoritma yang digunakan, akan dilakukan dengan membuat suatu skenario sebagai berikut:
Ari akan mengirimkan pesan terenkripsi dengan algoritma Elgamal kepada Yani. Dimana isi pesan tersebut adalah: "saya ganteng". Untuk menyelesaikan skenario ini berikut adalah tahapannya.

3.2.1 Pembangkitan kunci dengan algoritma Diffie-Hellman

Untuk membangkitkan kunci, Ari sebagai pengirim akan mengirimkan bilangan acak integer x besar. Yani sebagai penerima pesan juga akan melakukan hal yang sama kemudian mereka akan melakukan pertukaran angka. Dalam skenario ini Ari memilih bilangan 257, dan Yani memilih 281. Karena Yani sebagai penerima maka Yani akan menentukan nilai p = 331 dan g = 2. Kemudian keduanya melakukan perhitungan untuk mendapatkan kunci publik masing-masing.

$$Y_{Ari} = g^{x} \text{mod} p$$

$$Y_{Ari} = 2^{257} \text{mod } 331$$
$$Y_{Ari} = 327$$
$$Y_{Yani} = g^{x} \text{mod} p$$
$$Y_{Yani} = 2^{281} \text{mod } 331$$
$$Y_{Yani} = 62$$

Dari perhitungan diatas diperoleh kunci publik

$$Public_{Ari} = y, g, p$$
$$Public_{Ari} = (327,2,331)$$
$$Public_{Yani} = y, g, p$$
$$Public_{Yani} = (62,2,331)$$

Setelah mendapatkan kunci publik, mereka akan saling bertukar kunci dan kemudian secara terpisah akan melakukan perhitungan untuk mendapatkan kunci rahasia.

$$X_{Ari} = Y_{Yani}^{x} \text{mod} p$$
$$X_{Ari} = 62^{257} \text{mod } 331$$
$$X_{Ari} = 128$$
$$X_{Yani} = Y_{Ari}^{x} \text{mod} p$$

$$X_{Yani} = 327^{281} mod\, 331$$
$$X_{Yani} = 128$$

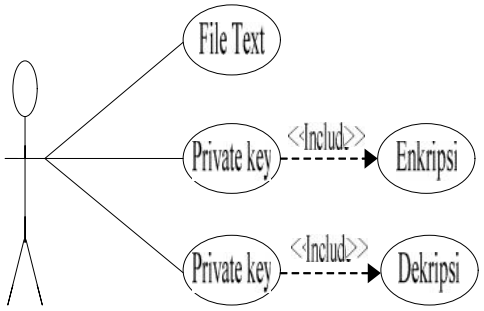
Dari hasil perhitungan diatas maka telah diperoleh kunci rahasia yaitu X = 128

3.3Pemodelan Sistem

Dalam perancangan struktur aplikasi ini dibantu dengan menggunakan bebrapa komponen-komponen pemodelan sistem dari metode Unified Modeling Language. Ada 2 (dua) jenis diagram yang akan digunakan yaitu: Use Case Diagram dan Activity Diagram.

3.3.1 Use Case Diagram

Prilaku beserta tugas – tugas dari tiap – tiap element maupun aktor yang terlibat dalam sistem yang akan dirancang, akan digambarkan dalam diagram use case yang bertujuan untuk memberikan gambaran secara umum tentang sistem yang akan dirancang.



3.4 Perancangan Sistem

Dalam tahap perancangan sistem ini akan dirancang interface dari sistem yang akan dibangun. Dalam sistem yang akan dibangun ini hanya memiliki 1 (satu) interface saja. Hasil rancangan ini nantinya akan diterapkan kedalam bahasa pemrograman Microsoft Visual Basic 2010

3.4.1 Rancangan Interface

Berikut ini akan disajikan rancangan user interface yang akan digunakan dalam sistem yang akan dibangun. Hanya ada 1 (satu) interface yang akan dirancang, karena aktivitas yang akan dilakukan dianggap mampu untuk dijalankan didalam satu interface.

3.4.2 Enkripsi pesan

Untuk enkripsi pesan yang akan dikirim, pesan akan dikonversi kedalam bentuk desimal ASCII.

Tabel 3.1 Konversi Karakter ke ASCII

Char	ASCII
s	115
a	97
y	121
a	97
<spasi>	32
g	103
a	97
n	110
t	116
e	101
n	110
g	103

Selanjutnya akan ditentukan bilangan acak K_i sebanyak jumlah karakter pada pesan yang akan dikirim. Dalam skenario ini jumlah karakter pada pesan adalah n = 12. Nilai K_i mempunyai syarat $1 \leq K_i \leq p - 2$.

$K_i = \{164, 237, 33, 68, 21, 99, 20, 152, 144, 291, 232, 208\}$

Selanjutnya akan dilakukan perhitungan untuk mendapatkan chipertext.

Untuk M_1:

$y = g^{K_1} \bmod p$

$y = 2^{164} \bmod 331$

$y = 165$

$\delta = Y^{K_1} \times M_1 \bmod p$

$\delta = 256^{164} \times 115 \bmod 331$

$\delta = 175$

Perhitungan terus dilanjutkan hingga M_{12}. Hasil perhitungan keseluruhan akan disajikan pada Tabel 3.2 dibawah ini.

Tabel 3.2 Hasil Perhitungan Enkripsi

γ	δ	ASCII	Char
165	175	115	s
207	250	97	a
8	276	121	y
256	228	97	a
267	75	32	<spasi>
181	194	103	g
299	28	97	a
4	111	110	n
150	151	116	t
267	185	101	E
203	131	110	n
83	114	103	g

Dari tabel diatas diperoleh chipertext sebagai berikut:
(165, 175) (207, 250) (8, 276) (256, 228) (267, 75) (181, 194) (299, 28) (4, 111) (150, 151) (267, 185) (203, 131) (83, 114)

3.4.3 Dekripsi pesan

Setelah menerima chipertext yang dikirimkan oleh Ari, maka Yani akan melakukan dekripsi terhadap chipertext tersebut agar pesan yang sesungguhnya dapat dibaca dan dipahami oleh Yani. Untuk melakukan dekripsi Yani membutuhkan kunci rahasia, dimana kunci rahasia ini telah didapatkannya melalui perhitungan-perhitungan pada saat pembangkitan kunci. Berikut ini adalah perhitungan dekripsinya.

$$M_i = \delta_i \times \gamma_i^{(r-1-private)} \bmod p$$

Untuk $M_1(165, 175)$

$M_i = 175 \times 165^{(331-1-128)} \bmod 331$

$M_i = 175 \times 165^{(331-1-128)} \bmod 331$

$M_i = 115$

Untuk $M_2(207, 250)$

$M_i = 250 \times 207^{(331-1-128)} \bmod 331$

$M_i = 250 \times 207^{(331-1-128)} \bmod 331$

$M_i = 97$

Untuk $M_3(8, 276)$

$M_i = 276 \times 8^{(331-1-128)} \bmod 331$

$M_i = 276 \times 8^{(331-1-128)} \bmod 331$

$M_i = 121$

Perhitungan ini akan terus dilakukan hingga blok terakhir yang terdapat dari chipertext yang diterima. Tabel 3.3 dibawah ini merupakan hasil perhitungan seluruh blok.

Tabel 3.3 Hasil Perhitungan Dekripsi

Char	ASCII	K_i	$\gamma = g^{Ki} mod p$	$\delta = \gamma^{Ki} \times M_i mod p$
s	115	164	165	175
a	97	237	207	250
y	121	33	8	276
a	97	68	256	228
<spasi>	32	21	267	75
g	103	99	181	194
a	97	20	299	28
n	110	152	4	111
t	116	144	150	151
e	101	291	267	185
n	110	232	203	131
g	103	208	83	114

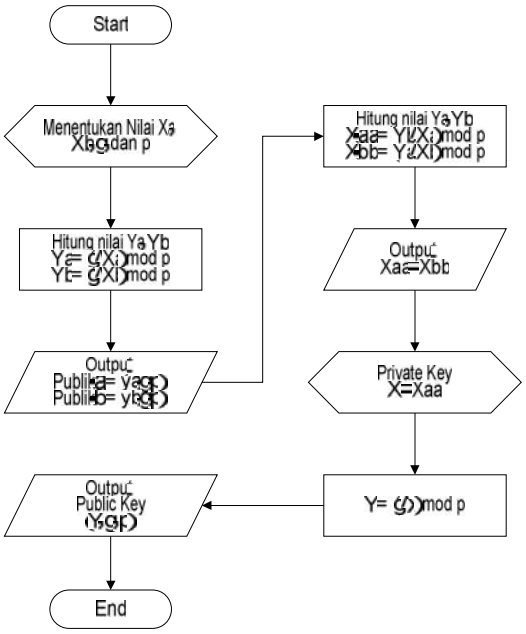
3.5Flowchart Program

Untuk menggambarkan aliran data pada algoritma yang akan diterapkan pada aplikasi yang akan dibangun, digunakan sebuah diagram yang disebut dengan flowchart. Diagram ini akan memberikan gambaran aliran data dari setiap input, proses, maupun output. Dalam pembahasan ini, ada 3 (tiga) flowchart yang akan disajikan, yaitu:

1. Flowchart pembangkitan kunci
2. Flowchart enkripsi
3. Flowchart dekripsi

3.5.1 Flowchart Pembangkitan Kunci

Pada flowchart ini akan menggambarkan proses pembentukan public key dan private key. Gambar 3.1 merupakan flowchart pembangkitan kunci.



Gambar 3.1 Flowchart Pembangkitan Kunci

Keterangan:

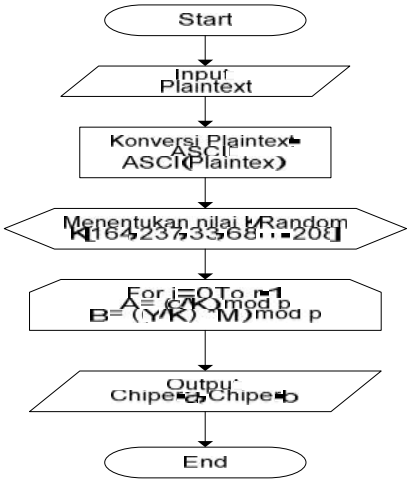
Pengirim harus mempersiapkan angka acak rahasia X_a. Sementara penerima mempersiapkan angka acak X_b, bilangan acak g, dan bilangan prima p, kemudian memberikan nilai g dan p kepada pengirim. Untuk nilai X_a tetap dirahasiakan oleh pengirim dan nilai X_b tetap dirahasiakan oleh penerima.

Pengirim dan penerima secara terpisah akan melakukan perhitungan terhadap nilai Y berdasarkan variabel-variabel yang dimiliki masing-masing. Sehingga pengirim memiliki nilai Y_a tersendiri dan penerima juga memiliki nilai Y_b tersendiri. Hasil perhitungan Y yang dilakukan oleh pengirim dan penerima akan ditukarkan. Sehingga pengirim memiliki nilai Y_b dan penerima memiliki nilai Y_a.

Dari pertukaran nilai itu pengirim dan penerima dapat memiliki kunci publik sementara, dimana kunci publik pengirim tidak sama dengan kunci publik penerima. Kunci publik ini selanjutnya akan digunakan untuk mendapatkan kunci rahasia X, dimana kunci rahasia pengirim adalah X_{aa} dan kunci rahasia penerima adalah X_{bb}. Perhitungan untuk mendapatkan kunci rahasia oleh pengirim dan penerima akan menghasilkan nilai yang sama, dimana $X_{aa} = X_{bb}$. Nilai inilah yang nantinya akan digunakan sebagai kunci rahasia, dan untuk mendapatkan nilai Y baru yang digunakan untuk proses dekripsi.

3.5.2 Flowchart Enkripsi

Pada flowchart ini akan menggambarkan proses pembentukan enkripsi dari plaintext. Gambar 3.2 merupakan flowchart enkripsi.



Gambar 3.2 Flowchart Enkripsi

Keterangan:

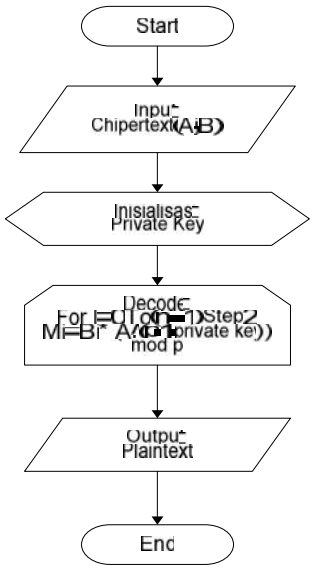
Pada proses enkripsi, pengirim terlebih dahulu memasukkan plaintext yang akan di enkrip menjadi chipertext. Setiap karakter pada plaintext akan dikonversi kedalam bentuk ASCII desimal agar bisa dilakukan perhitungan matematis untuk enkripsi.

Jumlah karakter dinyatakan dengan simbol n, dan setiap karakter dinyatakan dengan simbol M_i, dimana i adalah indeks dari karakter yang akan dienkrip. Setelah mendapatkan nilai desimal dari masing-masing karakter pada plaintext, tahap berikutnya adalah menentukan bilangan acak K_i dengan ketentuan $1 \leq K_i \leq p - 2$. Dimana K_i merupakan himpunan yang jumlah anggotanya sama dengan jumlah karakter pada plaintext.

Untuk mendapatkan chipertext, dilakukan perhitungan terhadap nilai A dan B. Perhitungan ini akan dilakukan sebanyak n, dimana n merupakan banyaknya karakter pada plaintext. Chipertext yang akan dihasilkan menjadi 2 (dua) blok angka untuk setiap karakternya. Dengan kata lain banyaknya blok pada chipertext 2 kali dari banyaknya karakter pada plaintext.

3.5.3Flowchart Dekripsi

Pada flowchart ini akan menggambarkan proses pembentukan dekripsi dari chipertext. Gambar 3.3 merupakan flowchart dekripsi. Proses dekripsi ini dapat dikatakan sebagai pembalikan dari proses enkripsi dimana chipertext yang diperoleh dari proses enkripsi akan dikembalikan menjadi plaintext agar penerima dapat mengetahui isi pesan yang sebenarnya.



Gambar 3.3 Flowchart Dekripsi

Keterangan:

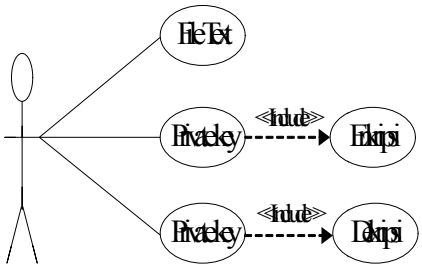
Untuk melakukan dekripsi terhadap chipertext, penerima harus menyiapkan private key. Dimana private key ini akan dibutuhkan pada saat perhitungan. Setelah private key disiapkan, maka penerima akan melakukan perhitungan tiap pasang blok pada chipertext. Seperti yang telah dijelaskan pada proses enkripsi, tiap karakter plaintext akan menghasilkan 2 (dua) blok chipertext. Perhitungan ini akan berulang hingga seluruh pasangan blok selesai di hitung dan menghasilkan plaintext.

3.6 Pemodelan Sistem

Dalam perancangan struktur aplikasi ini dibantu dengan menggunakan bebrapa komponen-komponen pemodelan sistem dari metode Unified Modeling Language. Ada 2 (dua) jenis diagram yang akan digunakan yaitu: Use Case Diagram dan Activity Diagram.

3.6.1 Use Case Diagram

Prilaku beserta tugas – tugas dari tiap – tiap element maupun aktor yang terlibat dalam sistem yang akan dirancang, akan digambarkan dalam diagram use case yang bertujuan untuk memberikan gambaran secara umum tentang sistem yang akan dirancang.



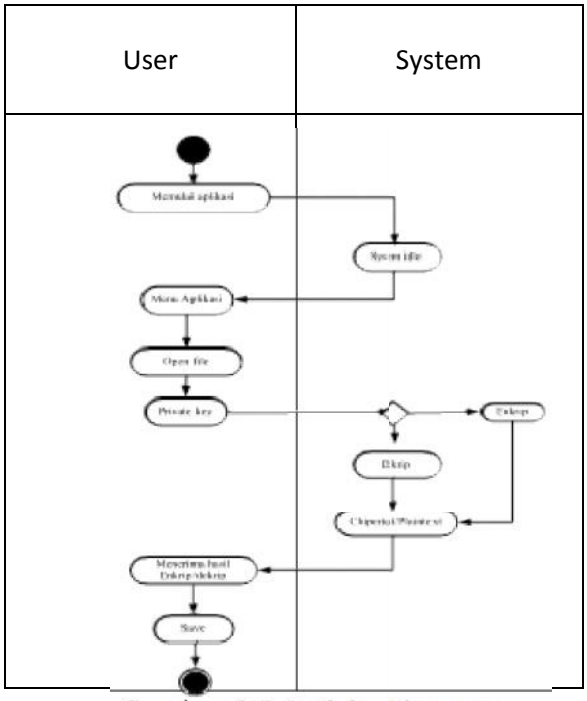
Gambar 3.4 Use Case Diagram

Keterangan:

Pada use case diagram dapat dilihat bahwa untuk memulai proses enkripsi dan dekripsi, user diminta untuk membuka file text. Untuk mendapatkan nilai-nilai variabel lain, aplikasi akan menyediakannya sesuai dengan ketentuan nilai pada tiap variabel nya. Pada enkripsi, aplikasi akan melakukan perhitungan-perhitungan berdasarkan input yang telah diberikan oleh user yaitu file text dan private key. Setelah proses enkripsi selesai, aplikasi akan membuat file text baru yang isinya telah dienkrip. Untuk proses dekripsi, user harus memberikan input private key dan file text yang sudah terenkrip. Kemudian aplikasi akan membuatkan sebuah file text baru hasil dekripsi.

3.6.2Activity Diagram

Berdasarkan use case diagram yang telah dibuat sebelumnya, activity diagram dapat dibuat seperti yang telah disajikan pada Gambar 3.5 dibawah ini.



Gambar 3.5 Activity Diagram

Keterangan:

Pada activity diagram yang telah disajikan diatas, terlihat bahwa user hanya diminta untuk Membukan file text dan memasukkan Private key yang akan di input. Kemudian sistem akan memberikan pilihan kepada user proses enkripsi atau dekripsi. Jika user memilih enkripsi, maka sistem akan melakukan perhitungan terhadap plaintext dan merubahnya menjadi chipertext. Apabila user memilih dekripsi, maka sistem akan mengolah file text menjadi plaintext. Kemudian hasil dari enkripsi dan dekripsi tersebut akan disimpan.

3.7 Perancangan Sistem

Dalam tahap perancangan sistem ini akan membahas mengenai rancangan interface dari sistem yang akan dibangun. Dalam sistem yang akan dibangun ini hanya memiliki 1 (satu) interface saja. Hasil rancangan ini nantinya akan diterapkan kedalam bahasa pemrograman Microsoft Visual Basic 2010

3.7.1 Rancangan Interface

Berikut ini akan disajikan rancangan user interface yang akan digunakan dalam sistem yang akan dibangun. Hanya ada 1 (satu) interface yang akan dirancang, karena aktivitas yang akan dilakukan dianggap mampu untuk dijalankan didalam satu interface.

IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan membahas tentang hasil dan pembahasan dari sistem yang telah dirancang pada bab sebelumnya. Hasil dari implementasi ini akan digunakan untuk menguji apakah program aplikasi yang dibuat sudah sesuai dengan sistem yang telah dirancang pada bab sebelumnya.

4.1 Kebutuhan Sistem

Untuk mengimplementasikan aplikasi Elgamal Cryptosystem yang menjadi pembahasan utama pada penelitian ini dibutuhkan perangkat keras dan perangkat lunak untuk menjalankan aplikasi yang telah dibangun.

4.1.1 Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam pembuatan dan implementasi kriptografi dengan menggunakan metode Elgama lini berupa seperangkat Notebook dengan spesifikasi perangkat keras sebagai berikut:

1. ProcessorIntel Core 2 Duo T5800 2.0 GHz
2. RAM 2 GB
3. Harddisk 320GB
4. LCD Monitor14.1” dengan resolusi 1280 x 800pixels.
5. Graphic Intel GMA 4500

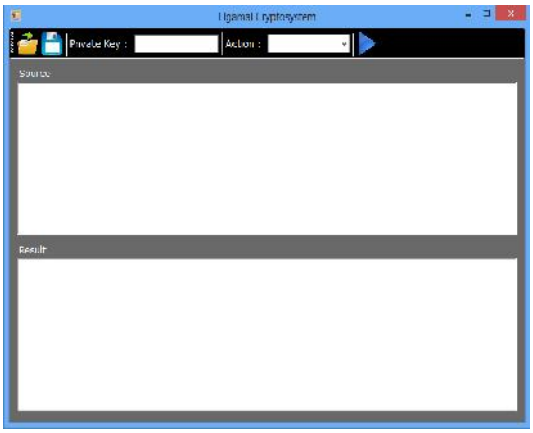
4.1.2Kebutuhan Perangkat Lunak

Sedangkan perangkat lunak yang digunakan untuk membangun aplikasikriptografi dengan menggunakan metode Elgamal ini adalah sebagai berikut:

1. Sistem Operasi Windows Seven Ultimate SP1.
2. Microsoft Visual Basic 2010

4.2 Implementasi Interface

Kriptografi dengan menggunakan metode Elgamal ini dilengkapi dengan antarmuka grafis yang bertujuan untuk memudahkan user dalam penggunaannya. Fungsi dari antarmuka ini adalah untuk memberikan input dan menampilkan output dari aplikasi. Pada aplikasi ini hanya memiliki 1 (satu)interface saja. dimana input dan output disajikan pada form yang sama.



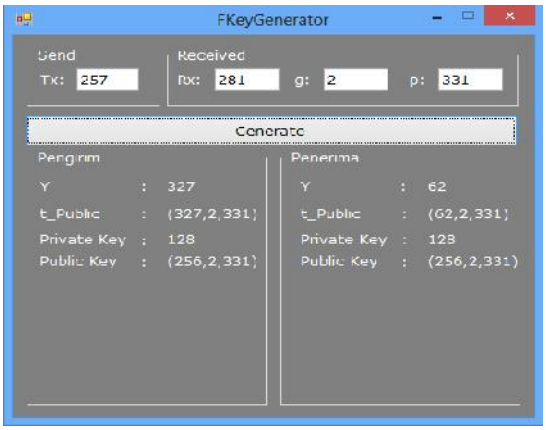
Gambar 4.1 Interface aplikasi kriptografi dengan metode Elgamal

4.3 Uji Coba

Dalam tahap ini akan dilakukan uji coba terhadap aplikasi kriptografi dengan menggunakan metode Elgamal yang telah dibangun. Uji coba akan dilakukan untuk Enkripsi dan untuk Dekripsi

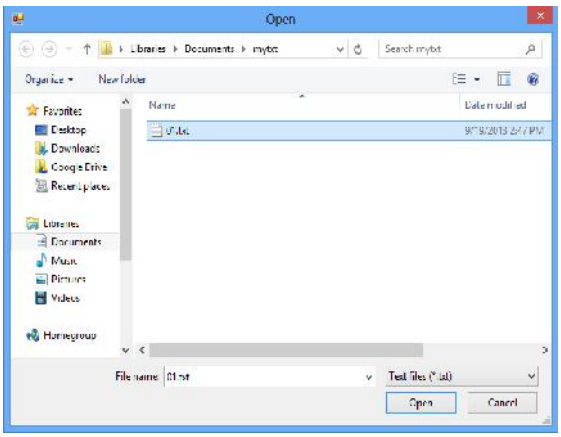
Pengujian 1: Enkripsi

Untuk melakukan enkripsi, pengirim terlebih dahulu melakukan pembangkitan kunci dari bilangan acak yang dipilih.



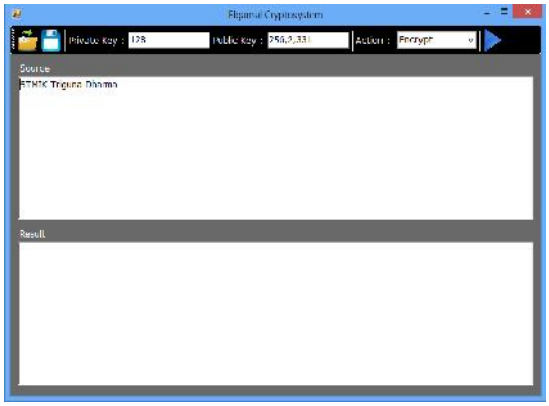
Gambar 4.2 Pembangkitan kunci untuk Enkripsi

Langkah berikutnya adalah membuka file teks yang *akan di* enkripsi. File yang akan dibuka melalui kontrol Open File Dialog.



Gambar 4.3 Membuka file yang akan dienkripsi

Setelah file dibuka, maka isi dari file tersebut akan ditampilkan pada kontrol Rich Text Box pada form utama.



Gambar 4.4 Isi dari file teks sebelum dienkripsi

Untuk melakukan enkripsi, pengirim harus melakukan klik pada tombol biru yang berada pada posisi kanan atas, dan memastikan jenis operasinya adalah Enkripsi.

Selanjutnya, hasil enkripsi tersebut dapat disimpan kedalam bentuk file txt. dengan melakukan klik pada tombol Save.

Pengujian 2: Dekripsi

Untuk melakukan dekripsi, penerima harus membuka file, kemudian memasukkan private key dan public key pada form utama.

Gambar 4.6 Hasil pengujian enkripsi

Untuk melakukan dekripsi terhadap chipertext, maka penerima harus memastikan mode aksi yang dipilih adalah Decrypt, kemudian melakukan klik pada tombol biru yang berada di sudut kanan atas.

Simpulan

Berdasarkan hasil pengujian dan evaluasi yang telah dilakukan sebelumnya, maka dapat diperoleh kesimpulan sebagai berikut:

1. Diperlukan instalasi aplikasi untuk setiap perangkat yang akan digunakan untuk melakukan enkripsi dan dekripsi.

2. Aplikasi yang dibangun dapat menerapkan metode pertukaran kunci Diffie Hellman dan menghasilkan kunci baru dan Aplikasi ini dapat melakukan enkripsi dan dekripsi Elgamal dengan menggunakan kunci yang telah dibangkitkan dengan metode Diffie Hellman

DAFTAR PUSTAKA

Ariyus, Doni. 2006. *Computer Security***.**Yogyakarta: Penerbit Andi Offset.

Sadikin, Rifki. 2012. *Kriptografi Untuk Keamanan Jaringan.* Yogyakarta: Penerbit Andi Offset.

Ramdan, Mangunpraja. 2010. *Peningkatan Keamanan Pertukaran Kunci Diffie-Hellman dengan Pengimbuhan Algoritma RSA*, Bandung: STIE.

http://www.sandrila.co.uk/articles/visio-articles/visio-flowchart-shapes