

PEMBUATAN APLIKASI KRIPTOGRAFI ALGORITMA

BASE64 MENGGUNAKAN JAVA JDK 1.6

HAYATUN NUFUS

Jurusan Sistem Informasi

Fakultas Ilmu Komputer dan Teknologi Informasi

Universitas Gunadarma

hello_nufuzZ@yahoo.co.id

06 September 2009

ABSTRAKSI

Keamanan data merupakan hal yang sangat penting dalam menjaga kerahasiaan informasi terutama yang berisi informasi sensitif yang hanya boleh diketahui isinya oleh pihak yang berhak saja, apalagi jika pengirimannya dilakukan melalui jaringan publik, apabila data tersebut tidak diamankan terlebih dahulu, akan sangat mudah disadap dan diketahui isi informasinya oleh pihak-pihak yang tidak memiliki wewenang.

Salah satu cara yang digunakan untuk pengamanan data adalah menggunakan sistem kriptografi yaitu dengan menyediakan isi informasi (plaintext) tersebut menjadi isi yang tidak dipahami melalui proses enkripsi (encipher), dan untuk memperoleh kembali informasi yang asli, dilakukan proses deskripsi (decipher), disertai dengan menggunakan kunci yang benar. Untuk membangun aplikasi yang terkomputerisasi ini, saya selaku penulis menggunakan Java JDK 1.6 dan NetBeans IDE 6.0 sebagai aplikasi pendukungnya. Aplikasi ini dibuat dalam Platform Windows sehingga memudahkan pemakai untuk menggunakannya.

Dalam pembuatan desain implementasi teknik kriptografi untuk pengaman file teks dengan menggunakan algoritma kriptografi Standard RFC 1521 Base 64 Alphabet yang digunakan untuk implementasi enkripsi dan deskripsi file teks sebagai komunikasi yang aman.

Kata Kunci : Kriptografi, Algoritma Base64

1. PENDAHULUAN

1.1 Latar Belakang

Ada cara yang lebih baik untuk mengamankan filetext agar sulit diketahui oleh pihak-pihak yang tidak diinginkan yaitu dengan cara mengenkripsi (encrypt) pesan (file) tersebut menjadi karakter-karakter acak yang tidak dimengerti sehingga hanya bagi seseorang yang memiliki kunci (key) yang dapat mengembalikan pesan kebentuk semula.

Saat ini, banyak algoritma-algoritma kriptografi bermunculan sebagai teknik untuk mengamankan data. Algoritma ini pada dasarnya dibagi menjadi algoritma klasik dan modern. Algoritma klasik beroperasi dalam mode karakter, sedangkan algoritma modern beroperasi dalam mode bit.

Oleh karena penulis ingin menanggapi masalah keamanan data dan sebagai bahan dalam penyusunan skripsi, maka penulis mencoba mngembangkan aplikasi kriptografi yang menggunakan gabungan dari teknik kriptografi klasik namun beroperasi pada mode bit, yaitu Algoritma base64 yang lebih mudah dalam pengimplementasiaanya. Aplikasi ini dibuat dengan menggunakan bahasa pemrograman Java JDK 1.6.

2. LANDASAN TEORI

2.1 Java

- Java adalah sebuah bahasa pemrograman komputer berbasisan kepada *Object Oriented Programming* (pemrograman berbasisan objek) yang sederhana dan tidak tergantung pada *platform*

yang digunakan. Bahasa ini dikembangkan oleh Sun Microsystems Corp. dan memiliki banyak keunggulan,

seperti sederhana, ukurannya kecil, dan *portable* (dapat dipindah-pindahkan di antara bermacam *platform* dan sistem operasi).

- Java™ Cryptography Extension (JCE) adalah sarana frame kerja yang diimplementasikan dalam pembuatan algoritma kriptografi dalam encryption, key generation, and decryption
- Didukung untuk enkripsi dengan kunci simetrik, asimetrik, blok, dan cipher aliran.
- JCE sudah tersedia didalam optional package (Extension) pada Java™ 2 SDK. SunJCE

provider juga sudah tersedia secara otomatis diregistrasi didalam java.security, security properties.

2.2 Graphical User Interface (GUI)

GUI merupakan suatu metode untuk antar muka komputer berbasis grafis. GUI digunakan dalam pembuatan program aplikasi dengan mempertimbangkan dua aspek yaitu keindahan tampilan dan kemudahan dalam penggunaan program.

2.3 IDE NetBeans 6.0

- Netbeans sebagai IDE (Interface Development Environment) ditujukan untuk memudahkan pemrograman interface
- Memiliki fitur lebih baik dari software seperti yaitu Module Matisse GUI (Graphical User Interface)

Builder yang bersifat Kriptografi adalah ilmu dan
lightweight untuk memudahkan seni untuk menjaga keamanan pesan
perancangan layout yang bertujuan menjaga kerahasiaan

- Pemrograman dilakukan informasi yang terkandung dalam data dengan konsep free-design sehingga informasi tersebut tidak dapat
- Memudahkan untuk membuat diketahui oleh pihak yang tidak bertanggung jawab.
aplikasi desktop dengan bertanggung jawab.
fasilitas yang dapat langsung Dalam menjaga kerahasiaan
digunakan di berbagai platform data, kriptografi mentransformasikan
tanpa harus menginstal data jelas (*plaintext*) ke dalam bentuk
software pendukungnya data sandi (*ciphertext*) yang tidak
- Memiliki fitur debugger untuk dapat dikenali. Ciphertext inilah yang mengetahui dimana terjadi kemudian dikirimkan oleh pengirim error (*sender*) kepada penerima (*receiver*).
- Memiliki fitur tooltip dimana Setelah sampai di penerima, ciphertext dapat diketahui cara perbaikan tersebut ditransformasikan kembali ke untuk error dalam bentuk plaintext agar dapat
- Berlisensikan Sun Public dikenali.
License atau *open source*

2.4 Kriptografi

2.5 Algoritma Base64

Algoritma Base64 merupakan algoritma yang menggunakan salah satu konsep algoritma enkripsi modern

yaitu algoritma *Block Cipher* yang yang berupa operasi pada mode bit namun algoritma Base64 ini lebih mudah dalam pengimplementasiannya dari algoritma-algoritma yang lainnya.

3. ANALISA DAN PEMBAHASAN MASALAH

3.1 Analisa Masalah

Base64 adalah metoda yang untuk melakukan encoding (penyandian) terhadap data binary menjadi format 6-bit character. Pada algoritma ini, rangkaian bit-bit palainteks dibagi menjadi blok-blok bit dengan panjang yang sama, biasanya 64 bit yang direpresentasikan dengan karakter ASCII. Base64 menggunakan karakter A – Z, a – z dan 0 – 9 untuk 62 nilai pertama, sedangkan 2 nilai terakhir digunakan symbol (+ dan /).

Standar yang penulis gunakan adalah MIME (Multipurpose Internet

Mail Extensions)/RFC 1521. RFC ini menegaskan sebuah standar untuk implementasi Base64 terhadap data binary dan melampirkan sebuah karakter padding “=” jika terdapat kekurangan pada byte.

Dalam streaming base64, spesifikasi mengharuskan setiap baris menjadi paling banyak 76 basis-64 karakter.

Tabel 3.1 Index Base64

Base64 Encoding Table							
Value	Char	Value	Char	Value	Char	Value	Char
0	A	16	Q	32	G	48	w
1	B	17	R	33	H	49	x
2	C	18	S	34	I	50	y
3	D	19	T	35	J	51	z
4	E	20	U	36	K	52	0
5	F	21	V	37	L	53	1
6	G	22	W	38	M	54	2
7	H	23	X	39	N	55	3
8	I	24	Y	40	O	56	4
9	J	25	Z	41	P	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	t	61	9
14	O	30	e	46	u	62	+
15	P	31	f	47	v	63	/

Di bawah ini merupakan sebuah contoh mudah mengkonversi kata “nuf” (decimal 110, 117, 102) menjadi dasar Notasi 64.

'01101110 01110101 01100110'

Ini 8-bit string dibagi ke dalam dua set 6 bit dan 4 blok.

'011011 100111 010101 100110'

Untuk mengkonversi 6-bit angka, maka rubah lagi ke dalam bentuk decimal yang didapatkan 27, 39, 20, dan 38 yang jika dilihat pada dasar abjad-64 pada tabel diatas, maka didapatkan alphabet **"bNUm"**. Namun jika string biner tidak tepat dibagi dalam 6-bit dan urutan biner tidak merupakan ukuran 3 byte, maka Base64 mengaturnya dengan menambahkan padding pada bit terakhir. Sebagai contoh kata **"nufus"**, maka akan dipisah menjadi **"nuf"** dan **"us+1byte"**.

Dalam kasus ini, jika diberikan contoh di mana satu byte yang tersisa, maka perlu tambahan dua byte pad dengan semua zeros sampai akhir urutan biner. Kemudian satu byte

tersebut dapat diwakili dengan satu byte dari 64 karakter dari Base64 diikuti oleh dua karakter padding.

Padding karakter yang telah ditentukan adalah '='.

Mari mempertimbangkan contoh string "s" dengan decimal 115 berikut :

'01110011'

Contoh pad single-byte dengan dua byte dari angka nol.

'00000001 00000000 00000000 '

Sekarang biner tersebut di set urutan enam byte.

'000000 "010000"000000 "000000'

Setelah di set, maka ditemukan karakter berdasarkan Base64 adalah **"AQ=="**. Begitu pula pada penambahan-penambahan padding lainnya.

Selanjutnya, contoh proses dekripsi. Contoh **"bNUm"** diatas, lalu

dirubah menjadi angka Index 27, 39,

20, dan 38.

Kemudian ubah menjadi 6-bit biner.

'011011 100111 010101 100110'

Set 6-bit, diubah kedalam 8-bit string.

'01101110 01110101 01100110'

Maka akan dihasilkan angka **decimal**

110, 117, dan 102 dimana karakter-

karakter semulanya adalah **“nuf”** jika

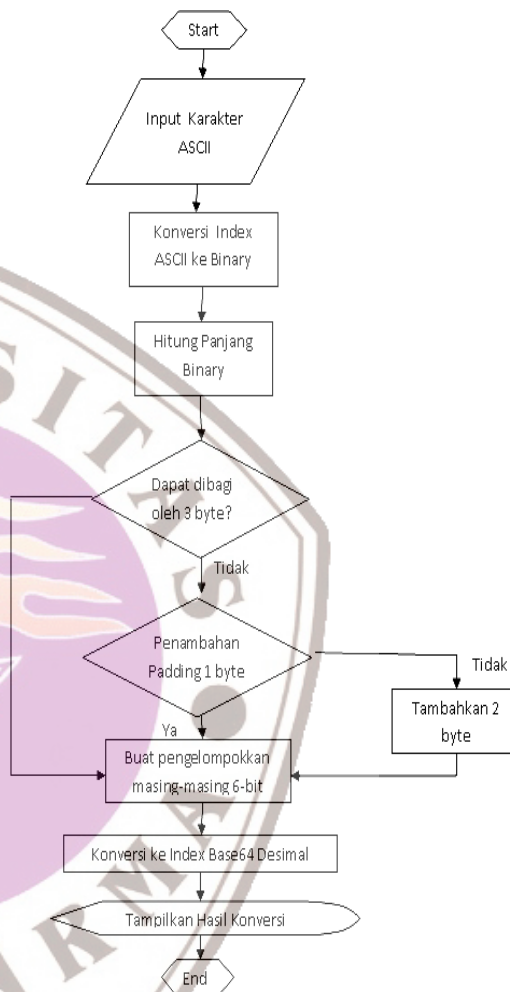
dilihat pada tabel ASCII.

Berdasarkan contoh-contoh

yang dapat dilihat, maka algoritma

umum proses encoding dari ASCII ke

Base64 adalah sebagai berikut



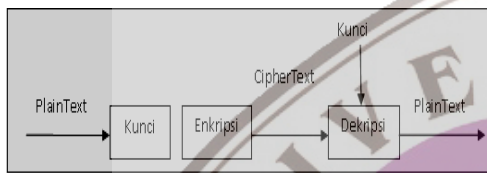
Gambar 3.1 Diagram alur algoritma

konversi ASCII – Base64

Algoritma kriptografi Base64

ini sebenarnya menggunakan algoritma kunci simetris atau disebut juga algoritma kriptografi konvensional, yaitu algoritma yang menggunakan kunci untuk proses

enkripsi sama dengan kunci untuk proses dekripsi. Dibawah ini adalah gambar proses enkripsi dan dekripsi dari aplikasi :

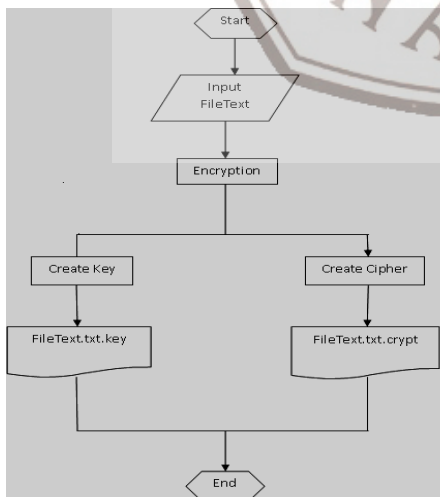


Gambar 3.2 Proses Encrypt Decrypt

Aplikasi

Pada simulasi algoritma Base64 terdiri dari dua tahap besar, yaitu tahap enkripsi dan tahap

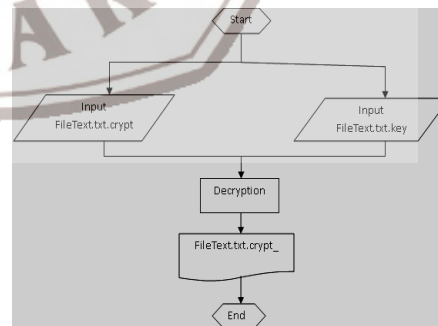
3.2 Metodologi Enkripsi



Gambar 3.3 Flowchart Enkripsi

deskripsi. Tahap pertama adalah pemilihan teks atau informasi (plainteks), yang akan diubah menjadi isi yang tidak dipahami melalui proses enkripsi (encipher), proses tersebut menghasilkan dua file yaitu file enkripsi dan file kunci (yang dinamakan enkripsi konvensional), file kunci digunakan pada saat memperoleh kembali informasi yang asli (decipher).

3.3 Metodologi Dekripsi



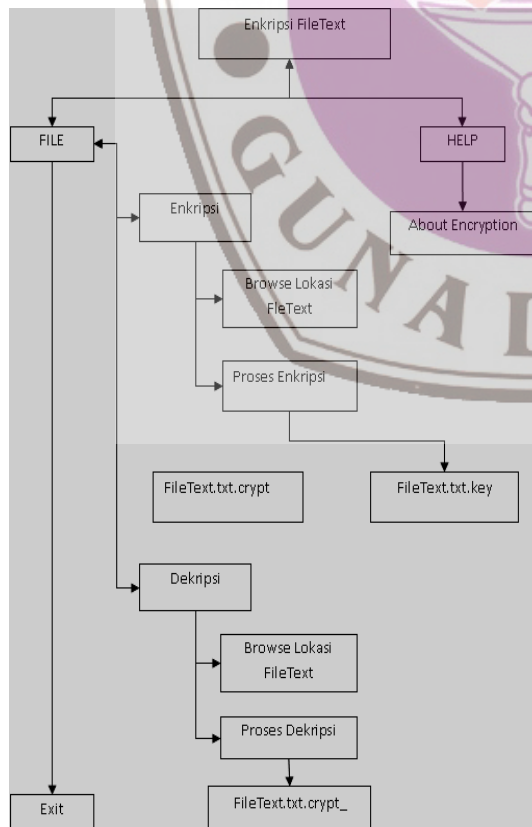
Gambar 3.4 Flowchart Dekripsi

4. Implementasi

4.1 Implementasi Output

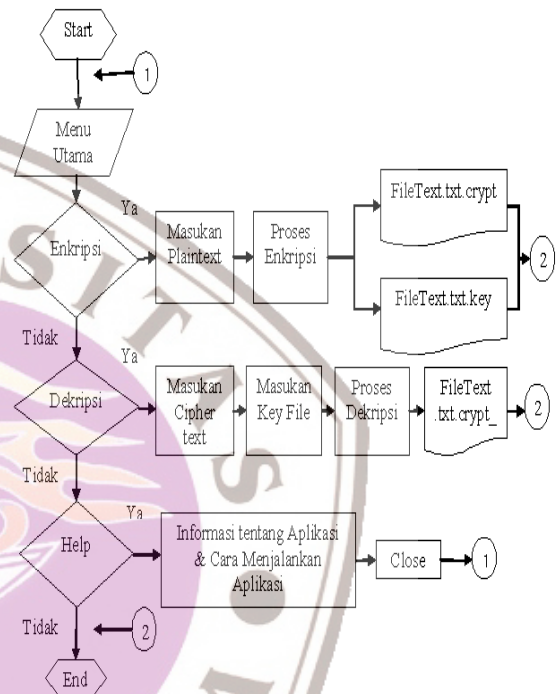
Implementasi perancangan user interface bertujuan mengimplementasikan semua hasil perancangan kedalam dunia nyata, agar dapat dipergunakan oleh user. Implementasi ini menggunakan software Netbeans 6.0 yang berbasis Java.

4.2 Bagan Struktur



Gambar 4.1 Struktur Navigas

4.3 Flowchart Program



Gambar 4.2 flowchart Program

4.4 Tampilan Antarmuka Pengguna (User Interface)

4.4.1 Pembuatan Menu Utama

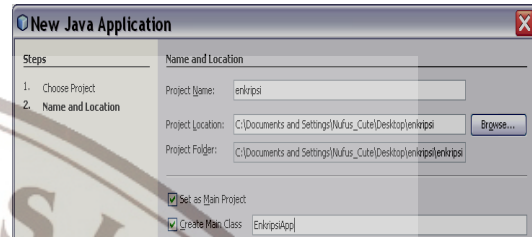
Langkah pembuatan tampilan aplikasi ini menggunakan IDE Netbeans 6.0. langkah-langkahnya adalah sebagai berikut :

1. Jalankan Netbeans 6.0

2. Pilih Menu File kemudian New Project setelah itu akan keluar jendela New Project

Seperti pada gambar dibawah ini :

3. Pada kotak Categories pilih Java dan pada kotak Projects pilih Java Application dan klik Next.



Gambar 4.3 Kotak Dialog New Java Application

4. Ubah nama Project Name sesuai yang diinginkan. Contoh : *enkripsi*.

Sampai pada tahap diatas berarti satu *project* telah tercipta dan untuk selanjutnya tinggal membuat *class* atau *form* seperti dibawah ini.

5. Pilih lokasi yang diinginkan untuk menyimpan project. Contoh : C:\My Document and setting\Nufus_Cute\Desktop\enkripsi.

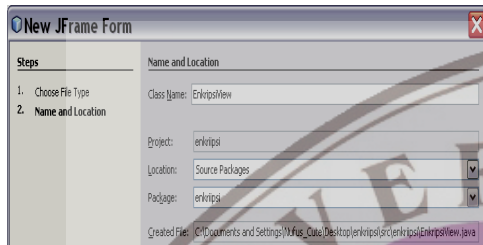
1. Klik kanan pada node *enkripsi* pilih New dan pilih JFrame Form.

6. Centang pada label Set As Main Project dan beri nama kelas pada Create main class. Contoh : *EnkripsiApp*. Setelah itu klik Finish.

2. Pada jendela New JFrame Form, isikan nama *class* untuk *Frame* pada textbox class name misalnya *EnkripsiView*.

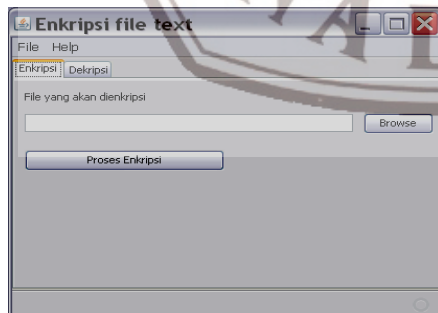
3. Isi juga *package* yang kita inginkan pada textbox package misalnya *enkripsi* kemudian

klik finish. Seperti gambar dibawah ini :



Gambar 4.4 Kotak Dialog New JFrame

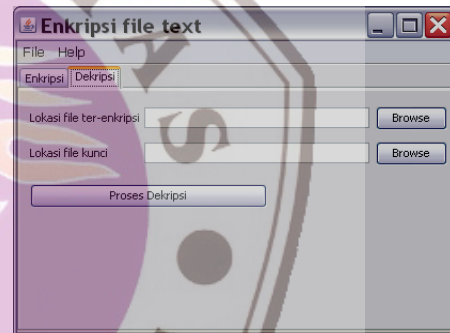
4. Tambahkan komponen-komponen yang dibutuhkan, lalu atur tampilan sedemikian rupa hingga tampak pada gambar di bawah ini :



Gambar 4.5 Tampilan Menu Utama (Encrypter)

Encrypter File adalah tampilan yang muncul pertama kali saat

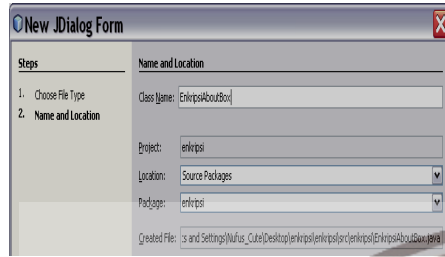
menjalankan program dan memilih menu Enkripsi ,juga merupakan menu utama. Dan tampilan decrypter juga terdapat pada menu utama setelah memilih menu deenkripsi, hasilnya seperti gambar dibawah ini :



Gambar 4.6 Tampilan Decrypter

4.4.2 Pembuatan Tampilan About Encryption

1. Buatlah sebuah jDialog, lalu ubah Class Name menjadi *EnkripsiAboutBox*, atur *package*-nya seperti gambar dibawah ini :



Gambar 4.7 Kotak Dialog New JDialog Form

2. Tambahkan komponen-komponen yang dibutuhkan lalu ubah nama form ini menjadi *About Encryption*, lalu atur tampilan sedemikian rupa hingga tampak pada gambar di bawah ini :



Gambar 4.8 Tampilan About Encryption

4.4.3 Build Program

Langkah-langkah build adalah

sebagai berikut :

1. Sorot project *Enkripsi*
2. Klik kanan dan pilih Build.

Hasil build akan tersimpan pada lokasi penyimpanan project, yaitu pada folder *dist*.

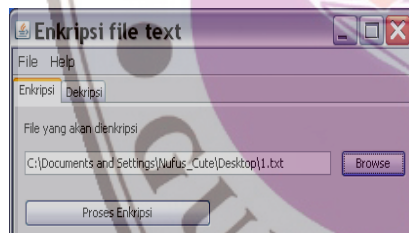
3. Aplikasi yang dihasilkan bentuk file yang bertipe JAR.

Untuk menjalankan aplikasi ini, klik 2 kali pada file JAR dalam folder *dist* dan aplikasi akan dijalankan sama seperti saat me-running program di Netbeans.

4.4 Cara Menggunakan Aplikasi

Aplikasi ini dibuat hanya untuk encrypt dan decrypt pada file text agar pesan tersebut dapat sampai ke tangan orang yang tepat dan dapat dipergunakan sesuai fungsinya. Berikut adalah cara penggunaan aplikasi, yaitu :

1. Jalankan aplikasi yang telah di-*build* pada folder dist.
2. Setelah muncul tampilan menu utama (Encrypt), lalu pilih menu *Enkripsi* dan masukkan file text yang akan di-encrypt kemudian tekan tombol *Proses Enkripsi*, seperti gambar dibawah ini :



Gambar 4.9 Enkripsi File Text
Maka akan dihasilkan file text yang telah terenkripsi beserta file kuncinya.

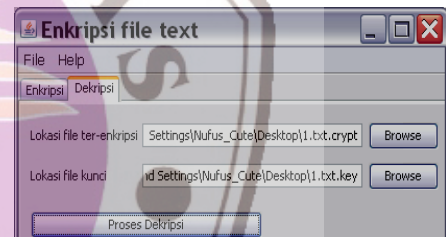


Gambar 4.10 Encrypt File



Gambar 4.11 Key File

3. Untuk men-decrypt, masih pada tampilan menu utama lalu pilih menu *Dekripsi* kemudian masukkan file ter-encrypt beserta kuncinya lalu tekan tombol *Proses Dekripsi* seperti pada gambar dibawah ini :



Gambar 4.12 Dekripsi File Text

4. Lihat hasilnya pada lokasi asal file tersebut, akan ditemukana file yang telah di-decrypt yang bersi file semula.

5. KESIMPULAN

Aplikasi ini dibuat bertujuan sebagai salah satu cara yang lebih baik yang mempermudah bagi siapa saja untuk dapat

mengamankan sendiri filetext (berisi teks rahasia) dengan cara memilih filetext yang akan di-encrypt maka secara otomatis akan terbentuk ciphertext (text yang sudah disandikan) bersama dengan key file yang kemudian digunakan untuk mengembalikan ciphertext ke bentuk teks semula. Aplikasi ini dirancang sebagai Desktop Application.

6. REFERENSI

[1] Hartati, A. Sri, *Pemrograman GUI Swing Java dengan Netbeans 5*, Yogyakarta, Andi, 2006

[2] "How to Encode a String to Base64 WithJava"
http://www.dimgt.com.au/decode_decrypt.html, 2 April 2009, 10:55.

[3] "Base64 Encoder/Decoder in Java"

http://www.dimgt.com.au/encode_encrypt.html,
15 Juni 2009, 15:00.

[4] "Tutorial Base64"
<http://www.source-code.biz>, 23 Juni 2009, 13.30.

[5] "Pemanfaatan MIME Base64.pdf"
<http://www.kbcafe.com/artikel/>,
23 Juni 2009, 20.30.

[6] Kadir, Abdul, *Dasar Pemrograman Java 2*, Yogyakarta, Andi, 2003

[7] Munir, Rinaldi, *Kriptografi*, Bandung. Informatika, 2006

[8] Supandi, Ir. Yuniar, *Belajar Semua Edisi Java 2 untuk Segala Tingkat*, Jakarta, P.T Elex Media komputindo, 2009