

TEKNIK KEAMANAN FILE MENGGUNAKAN KRIPTOGRAFI DENGAN ALGORITMA VERNAM CIPHER

Hernalom, Belathika Gornea

Abstrak

Untuk menjamin keamanan dari suatu file dibutuhkan suatu proses penyandian. Enkripsi dilakukan ketika file akan dikirim, proses ini akan mengubah suatu file asal menjadi file rahasia yang tidak dapat dibaca. Sementara itu proses dekripsi dilakukan oleh penerima file yang dikirim tersebut. File rahasia yang diterima akan diubah kembali menjadi file semula.

Untuk mengamankan file yaitu dengan mengimplementasikan kriptografi untuk penyandian file, contohnya adalah algoritma Vernam Cipher. Sebagai contoh system kriptografi simetri, algoritma ini cukup mampu untuk mengamankan file. Dengan itu maka dibuatlah program aplikasi kriptosistem yang digunakan untuk melakukan pengamanan file sehingga hanya orang-orang tertentu saja yang dapat mengolah file.

Kata Kunci: file, kriptografi, algoritma vernam chipher

Abstrak

To ensure the safety of a file requires a process of encoding. Encryption is done when the file is sent, the process will convert a source file into a confidential file that can not be read. In the meantime the decryption process performed by the receiver of the sent file. Secret files received will be converted back into the original file.

To secure a file is to implement cryptography for encryption of files, for example, is Chipher Vernam algorithm. For example cryptographic system symmetry, this algorithm is capable enough to secure files. With that then made cryptosystem application program used to perform file a safety measure so that only certain people are able to process the file.

Key Word: file, criptography, vernam chipher algorithm

PENDAHULUAN

A. Latar Belakang Masalah

Teknologi komunikasi dan informasi berkembang dengan pesat dan memberikan pengaruh besar bagi kehidupan manusia. Seiring dengan perkembangan teknologi sekarang ini yang semakin pesat maka proses pengiriman data dapat dilakukan dengan mudah dan melalui berbagai macam media yang telah ada antara lain, melalui media internet dengan menggunakan fasilitas *e-mail*, melalui transfer data antar perangkat *mobile* (*handphone*, PDA dan *flashdisk*) maupun dengan teknologi radio *frequency* (*bluetooth*, IrDA, GPRS) hingga dengan menggunakan jaringan komputer.

Teknik keamanan pengiriman data dipasang untuk mencegah pencurian, kerusakan, dan penyalahgunaan data yang terkirim melalui jaringan komputer. Kriptografi digunakan untuk mengamankan data-data penting pada sebuah *file*. Data yang terkandung dalam *file* disandikan atau dienkripsi untuk diubah menjadi simbol tertentu sehingga hanya orang tertentu saja yang dapat mengetahui isi dari data tersebut.

Dalam perkembangan kriptografi saat ini, telah banyak tercipta algoritma-algoritma yang dapat digunakan untuk mengubah data asli (*plain text*) menjadi simbol tertentu (*cipher text*). Salah satu contohnya adalah algoritma Vernam Cipher.

Berdasarkan pada uraian diatas, maka penulis mengambil judul “Teknik Keamanan File Menggunakan Kriptografi Dengan Algoritma Vernam Cipher” yang diharapkan dapat berguna untuk proses pengamanan data sehingga diharapkan tidak terjadi pencurian atau penyadapan data.

B. Rumusan Masalah

Rumusan masalah yang muncul dari latar belakang yang telah di sajikan di atas adalah sebagai berikut :
Bagaimana menerapkan teknik keamanan file menggunakan kriptografi dengan algoritma vernam cipher ?

C. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menerapkan teknik keamanan *file* menggunakan kriptografi sehingga akan menghasilkan sebuah perangkat lunak yang dapat mengimplementasikan algoritma Vernam Cipher pada Kriptografi.

LANDASAN TEORI

A. Pengertian File

File atau berkas adalah sekumpulan data (informasi) yang berhubungan yang diberi nama dan tersimpan di dalam media penyimpanan sekunder (*secondary storage*). *File* memiliki ekstensi. Ekstensi berkas merupakan penandaan jenis berkas lewat nama berkas. Ekstensi biasanya ditulis setelah nama berkas dipisahkan dengan sebuah tanda titik. Pada sistem yang lama (MS-DOS) ekstensi hanya diperbolehkan maksimal 3 huruf, contohnya : exe, bat, com, txt. Batasan itu dihilangkan pada sistem yang lebih baru (*Windows*).

B. Kriptografi

a. Gambaran Umum Kriptografi

Kriptografi merupakan sebuah ilmu yang digunakan untuk penyandian data. Kriptografi telah dikenal dan dipakai cukup lama sejak kurang lebih tahun 1900 sebelum masehi pada prasasti-prasasti kuburan.

b. Enkripsi dan Dekripsi

Enkripsi adalah transformasi data dalam bentuk yang tidak dapat terbaca dengan sebuah kunci tertentu. Tujuannya adalah untuk meyakinkan privasi dengan menyembunyikan informasi dari orang-orang yang tidak ditujukan, bahkan mereka yang memiliki akses ke data terenkripsi. Sedangkan dekripsi merupakan kebalikan dari enkripsi, yaitu transformasi data terenkripsi kembali ke bentuknya semula.

c. Algoritma Vernam Cipher

Vernam cipher merupakan deretan karakter kunci yang dibangkitkan secara acak, yang ditemukan oleh Mayor J. Maugborne dan G. Vernam tahun 1917. Algoritma ini merupakan algoritma berjenis *symmetric key* yang artinya bahwa kunci yang digunakan untuk melakukan enkripsi dan dekripsi merupakan kunci yang sama. Aturan enkripsi yang digunakan pada algoritma vernam cipher :

- Enkripsi: $ci = (pi + ki) \bmod 26$
- Dekripsi: $ci = (pi - ki) \bmod 26$

Dimana :

$ci = \text{cipher teks}$

$pi = \text{plainteks}$

$ki = \text{kunci}$

Apabila diketahui teks asli : “*onetimepad*”

Dengan kunci : “*tbfrgfarfm*”

Diasumsikan $a = 0, b = 1, \dots, z = 25$

d. Netbeans

Netbeans merupakan *Integrated Development Environment* atau IDE. Suatu IDE adalah lingkup pemrograman yang diintegrasikan ke dalam suatu aplikasi perangkat lunak yang menyediakan pembangun GUI, suatu *text editor*, suatu *compiler* atau *interpreter* dan suatu *debugger*.

e. Unified Modelling Language (UML)

Unified Modelling Language adalah bahasa standar yang digunakan untuk menjelaskan dan memvisualisasikan artifak dari proses analisis dan desain berorientasi objek. UML menyediakan standar pada notasi dan diagram yang bisa digunakan untuk memodelkan suatu sistem. UML dikembangkan oleh 3 orang pendekar “berorientasi objek”, yaitu Grady Booch, Jim Rumbaugh dan Ivar Jacobson.

METODE PENELITIAN

A. Metode Pengumpulan Data

Metode studi pustaka dilakukan dengan mengumpulkan data ataupun informasi dari berbagai buku, jurnal yang berkaitan dengan kriptografi dengan algoritma vernam cipher dan juga masalah keamanannya.

B. Metode Analisis

Terkait masalah dalam penelitian yang penulis lakukan dalam tahapan menganalisa dari permasalahan yang terjadi, maka saya mencoba menganalisa hal tersebut dengan beberapa analisa sebagai berikut:

a. Analisa Sistem Berjalan

Menggambarakan proses kriptografi pada *file* yang sedang di enkripsi dan dekripsi dengan menggunakan algoritma vernam cipher, dan menerapkan algoritma vernam cipher pada *file* tersebut.

b. Perumusan Masalah

Perumusan masalah yang ada adalah bagaimana mengamankan *file* yang akan dikirim ke penerima *file* tersebut agar dalam proses pengiriman data tersebut tidak dapat dicuri,disadap atau disalahgunakan oleh orang yang tidak bersangkutan.

c. Usulan Pemecahan Masalah

Dari permasalahan yang telah diuraikan di atas, diperlukan adanya sebuah aplikasi yang dapat menjaga kerahasiaan dari sebuah proses pengiriman data *file* tersebut sehingga keberadaannya tidak terdeteksi oleh pihak lain yang tidak berhak atas *file* tersebut. Aplikasi tersebut dapat mengamankan proses pengiriman data.

ANALISA DAN PERANCANGAN

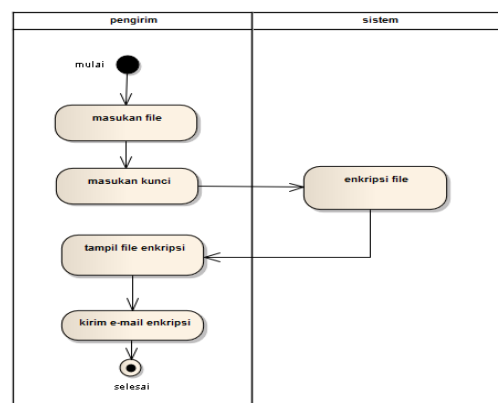
A. Analisa Proses Kerja Sistem

Analisa proses kerja sistem menjelaskan alur proses dari aplikasi yang dibangun.

Data *File* dienkripsi pengirim dari *plainteks* menjadi *Cipherteks*, diterima oleh penerima didekripsi dari *cipherteks* menjadi *plainteks*.

a. Proses enkripsi pesan

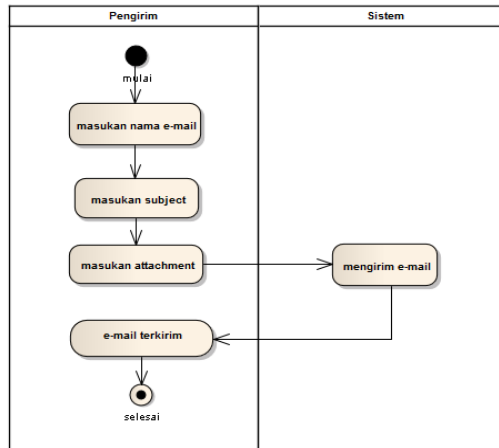
Proses ini pengirim memasukan *file* dan kunci. *File* di enkripsi dari *plainteks* menjadi *cipherteks*, Kemudian Tampil *file* yang sudah di enkripsi.



Gambar 1. Proses enkripsi pesan

b. Proses pengiriman pesan

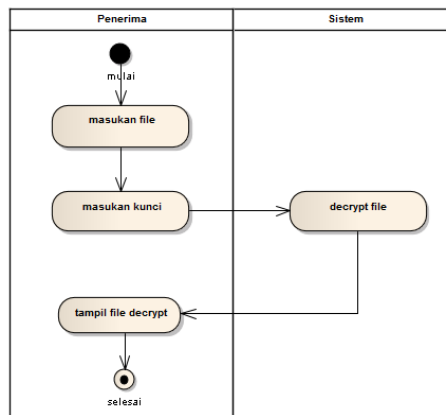
Proses ini pengirim mengirimkan *e-mail* kepada penerima dengan memasukkan *e-mail* penerima, *subject* dan *attachment* yang akan dikirim kemudian langsung dikirim kepada si penerima tersebut.



Gambar 2. Proses pengiriman pesan

c. Proses menerima pesan

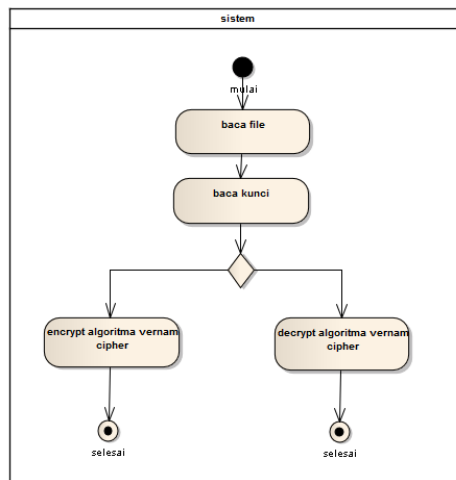
Proses ini penerima memasukkan *file* dan kunci. *File* di dekripsi dari *cipherteks* menjadi *plainteks*, Kemudian Tampil *file* yang sudah di dekripsi.



Gambar 3. Proses menerima dan mendekripsi pesan

B. Analisa Penerapan Algoritma Vernam Cipher

Algoritma vernam cipher diterapkan ketika data *file* sedang di enkripsi dan dekripsi, dan untuk melakukan enkripsi dan dekripsi kunci yang digunakan merupakan kunci yang sama.



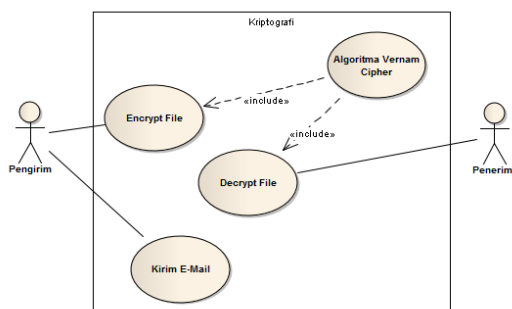
Gambar 4. Proses penerapan algoritma vernam cipher

a. Perancangan Sistem

Perancangan meliputi beberapa diagram *Unified Modeling Language* (UML), diantaranya: diagram *use case*, diagram *activity*, diagram *sequence* dan diagram *class*, perancangan struktur menu dan perancangan antarmuka.

b. Diagram Use Case

Penerima adalah seseorang yang akan menerima *file* yang akan dikirimkan oleh pengirim. Penerima bertugas untuk membuka *file* yang telah dikirimkan tersebut kemudian didekripsi agar *file* tersebut bisa terbaca dengan jelas. Diagram *use case* akan menjelaskan aplikasi yang akan dibangun. Adapun diagram *use case* pada aplikasi kriptografi dengan algoritma vernam cipher sebagai berikut:



Gambar 5. Diagram *use case* aplikasi kriptografi

c. Definisi Aktor

Definisi dari aktor pada aplikasi kriptografi dengan algoritma vernam cipher sebagai berikut:

Aktor pengirim

Aktor pengirim merupakan orang yang ditujukan untuk melakukan pengiriman data *file*.

Aktor penerima

Aktor penerima merupakan orang yang ditujukan untuk menerima data *file*.

Definisi *use case*

Definisi dari *use case* pada aplikasi kriptografi dengan algoritma vernam cipher sebagai berikut:

Enkripsi *File*

Berfungsi untuk mengamankan *file* dan mengubah *file* dari *plainteks* menjadi *cipherteks*.

Dekripsi *File*

Berfungsi untuk mengembalikan *file* seperti semula, mengubahnya dari *cipherteks* menjadi *plainteks*.

Algoritma Vernam Cipher

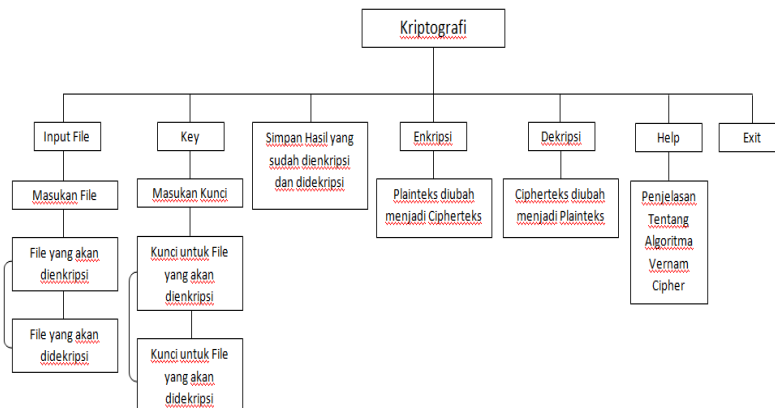
Berfungsi untuk menjalankan proses enkripsi dan dekripsi tersebut.

E-Mail

Berfungsi untuk mengirimkan *e-mail* pada *file* yang sudah dienkrpsi oleh pengirim.

Rancangan Struktur Menu

Rancangan struktur menu dilakukan untuk mempermudah interaksi antara sistem dengan pengguna.



Gambar 6. Rancangan struktur menu

Pada gambar 6 terdapat rancangan struktur menu kriptografi dapat dijelaskan sebagai berikut :

Kriptografi : tulisan yang tersembunyi dengan adanya tulisan yang tersembunyi ini, orang-orang tidak mengetahui bagaimana tulisan tersebut disembunyikan dan tidak mengetahui bagaimana cara membaca maupun menerjemahkan tulisan tersebut.

Input *File* : memasukan *file* yang akan dienkrpsi dan dekripsi.

Key : memasukan kunci untuk *file* yang akan dienkrpsi dan dekripsi.

Simpan : simpan hasil yang sudah dienkrpsi dan dekripsi ke folder.

Enkripsi : *plainteks* diubah menjadi *cipherteks*.

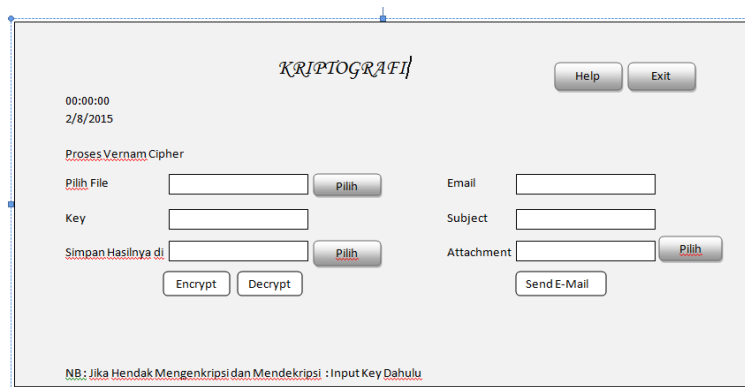
Dekripsi : *cipherteks* diubah kembali menjadi *plainteks*.

Help : akan menjelaskan tentang algoritma vernam cipher.

Exit : keluar dari program.

d. Rancangan Interface

Rancangan tampilan menu utama program teknik keamanan *file* menggunakan kriptografi dengan algoritma vernam cipher dapat dilihat pada gambar 7.



Gambar 7. Rancangan interface

Dibawah ini merupakan keterangan dari gambar 7 diatas, yaitu :

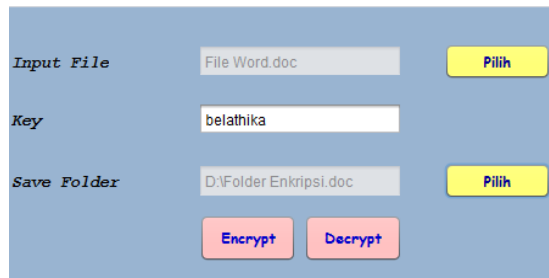
- Frame* atas merupakan nama perancangan yang akan dibuat.
- Waktu dan tanggal untuk mengetahui jam berapa proses pengenkripsi dan dekripsi yang sedang diproses dan pada tanggal berapa proses terlaksanakannya.
- Button* pilih pada pilih *file* untuk memilih *file* yang akan dienkrpsi maupun didekripsi.
- Key* untuk memasukan kunci pada *file* yang akan dienkrpsi maupun didekripsi.
- Button* pilih pada simpan hasilnya di, untuk menyimpan hasil dari enkripsi dan dekripsi pada folder yang sudah dibuat.
- Button* encrypt merupakan proses pengenkripsian *file* yang ingin dienkrpsi.
- Button* decrypt merupakan proses pendekripsian *file* yang ingin didekripsi atau dikembalikan *file* seperti semula.
- E-mail* merupakan penginputan nama *e-mail* yang akan dikirim ke penerima.
- Subject* merupakan keterangan pada *e-mail* yang akan dikirim.
- Button* pilih pada *attachment* untuk memilih *file* enkripsi mana yang akan dikirim kepada penerima.
- Button* send *e-mail* digunakan untuk mengirim *e-mail* yang sudah siap dikirim ke penerima.
- Button* *Help* akan menjelaskan penjelasan mengenai algoritma vernam cipher.
- Button* *Exit* keluar dari program.

HASIL DAN IMPLEMENTASI

A. Hasil

dari hasil perancangan yang dibuat dengan demikian dapat diimplementasi dalam bentuk prototyping sebagai bahan uji kelayakan dari sebuah sistem yang akan diimplementasikan berikut gambarannya:

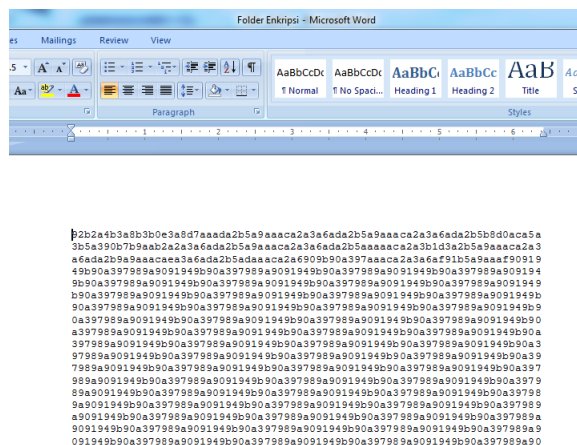
a. Hasil Dari Enkripsi Berbentuk Doc



The screenshot shows a software interface for file encryption. It features three input fields: 'Input File' with the value 'File Word.doc', 'Key' with the value 'belathika', and 'Save Folder' with the value 'D:\Folder Enkripsi.doc'. Each input field has a yellow 'Pilih' (Choose) button to its right. At the bottom of the interface, there are two buttons: a red 'Encrypt' button and a pink 'Decrypt' button.

Gambar 8 Tampilan aplikasi Enkripsi file berbentuk doc

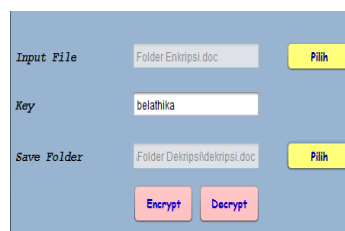
Penjelasan gambar 5.1 merupakan tampilan aplikasi untuk mengenkripsi *file* doc. Dengan mengisi *input file*, *key* dan *save folder* yang akan disimpan lalu ketik *encrypt*, maka akan muncul enkripsian dari *file* tersebut sebagai berikut.



Gambar 9. Tampilan hasil Enkripsi file berbentuk doc

Pada gambar 9 menampilkan hasil dari enkripsi berbentuk doc, dari *plainteks* menjadi *cipherteks*.

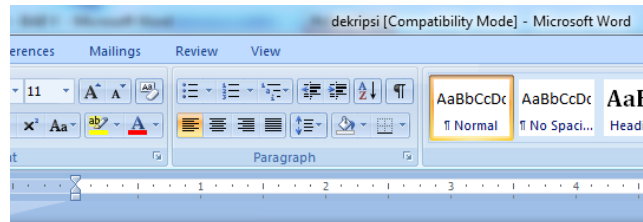
b. Hasil Dari Dekripsi Berbentuk Doc



The screenshot shows a software interface for file decryption. It features three input fields: 'Input File' with the value 'Folder Enkripsi.doc', 'Key' with the value 'belathika', and 'Save Folder' with the value 'Folder Dekripsi\dekripsi.doc'. Each input field has a yellow 'Pilih' (Choose) button to its right. At the bottom of the interface, there are two buttons: a red 'Encrypt' button and a pink 'Decrypt' button.

Gambar 10. Tampilan aplikasi Dekripsi file berbentuk doc

Penjelasan gambar 10 merupakan tampilan aplikasi untuk mendekripsi *file* doc. Dengan mengisi *input file*, *key* dan *save folder* yang akan disimpan lalu ketik *decrypt*, maka akan muncul dekripsi dari *file* tersebut sebagai berikut.



[Belathika Gornea](#)
[011101503125075](#)
[Teknik Informatika](#)

Gambar 11. Tampilan hasil Dekripsi file berbentuk doc

Pada gambar 11 menampilkan hasil dari Dekripsi berbentuk doc, dari *cipherteks* menjadi *plainteks*, mengembalikan seperti semula *file* yang tadinya terenkripsi.

B. Implementasi

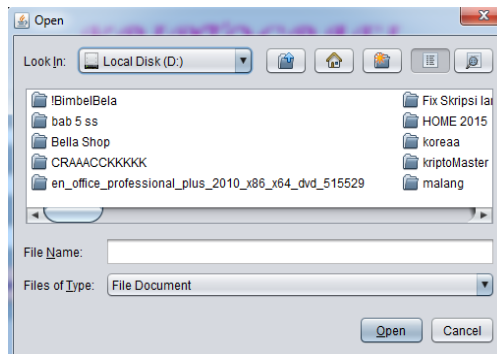
Bab ini merupakan implementasi dan pengujian terhadap sistem yang telah dibangun. Setelah analisis dan perancangan selesai dilakukan, kemudian dilanjutkan dengan implementasi pada bahasa pemrograman yang digunakan.

a. Implementasi Antarmuka

Implementasi antarmuka pada aplikasi kriptografi adalah sebagai berikut :

b. Tampilan menu pilih input file

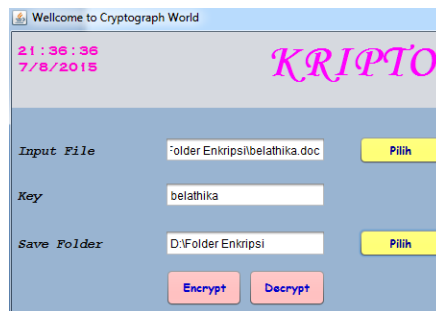
Menu *input file* menampilkan *file* mana yang akan dienkripsi maupun didekripsi. Menu ini difungsikan agar pengirim dan penerima lebih mudah menginput *file* yang mana yang akan dienkripsi dan dekripsi. Menu ini menampilkan yang mana *file* yang akan dienkripsi atau didekripsi. dengan memilih *folder* lalu pilih *open* dan akan muncul di menu pilih *file* untuk *file* yang sudah dipilih.



Gambar 12. Tampilan menu pilih input file

c. Tampilan menu key

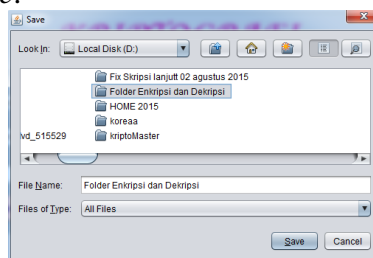
Tampilan menu *key* merupakan lanjutan dari menu *input file*. menu ini merupakan proses kunci untuk *file* yang akan dienkripsi atau didekripsi. Jadi sebelum mengenkripsi atau mendekripsi *file* harus menginput *key* terlebih dahulu.



Gambar 13. Tampilan menu key

d. Tampilan menu pilih save folder

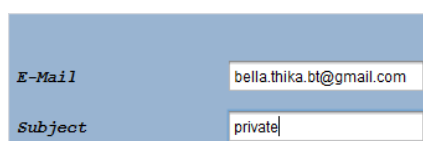
Menu ini menampilkan dimana hasil dari enkripsi atau dekripsi akan disimpan sehingga pengirim dan penerima dapat membuat sebuah folder tersendiri untuk menyimpan *file* yang sudah di enkripsi maupun sudah didekripsi dengan memilih *folder* lalu klik *save*.



Gambar 14. Tampilan menu pilih save folder

e. Tampilan menu e-mail dan subject

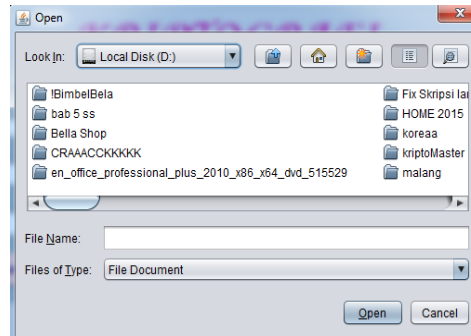
Menu ini adalah proses *input* nama *e-mail* dan *input subject* yang akan dikirim ke penerima.



Gambar 15. Tampilan menu e-mail dan subject

f. Tampilan menu pilih *attachment*

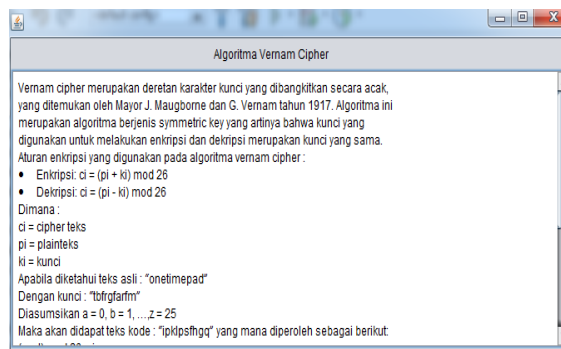
Menu ini untuk menampilkan lampiran enkripsi yang akan dikirim ke penerima.



Gambar 16. Tampilan menu pilih attachment

g. Tampilan menu *help*

Menu ini untuk menampilkan keterangan algoritma vernam cipher, mengetahui perhitungan pada algoritma vernam cipher.



Gambar 17. Tampilan menu *help*

KESIMPULAN DAN SARAN

A. Kesimpulan

Kesimpulan dari penelitian yang dilakukan dalam teknik keamanan *file* menggunakan kriptografi dengan algoritma vernam cipher adalah sebagai berikut:

1. Algoritma vernam cipher dapat digunakan dalam mengamankan data berupa *file* yang diproses dengan enkripsi dan dekripsi.
2. Algoritma vernam cipher dapat diterapkan menggunakan java netbeans.
3. Dengan penambahan menu pengiriman *e-mail* dapat mengirimkan *file* yang dienkripsi secara langsung tanpa harus membuka aplikasi lainnya untuk mengirimkan *e-mail*.
4. Dengan adanya aplikasi kriptografi ini pengirim maupun penerima *file* dapat mengamankan *file* dengan di enkripsi mengubah *file* dari *plainteks* menjadi *cipherteks* dan didekripsi mengubah kembali dari *cipherteks* menjadi *plainteks*.

B. Saran

Adapun saran yang dapat dijadikan acuan untuk pengembangan penelitian ini adalah sebagai berikut:

1. Harus dilakukan pengujian lebih lanjut tentang keamanan algoritma kriptografi lainnya selain algoritma vernam cipher.
2. Perlu pengembangan lebih lanjut dengan menggabungkan algoritma vernam cipher dengan algoritma yang lainnya agar memiliki tingkat keamanan yang lebih baik.

DAFTAR PUSTAKA

- Adams, Carlisle M. 1997. *The CAST-128 Encryption Algorithm*. Canada: Entrust Technologies.
- Ariyus, Dony. 2006. *Kriptografi: Keamanan Data dan Komunikasi*. Yogyakarta: Graha Ilmu.
- Bishop, David. 2003. *Introduction to Cryptography with Java Applet*, Jones and Bartlet Computer Science.
- C. Adams. 1997. "Constructing Symmetric Ciphers Using the CAST Design Procedure," *Designs, Codes and Cryptography*, v. 12, n.3, Nov 1997.
- Callas, J, L Donnerhacke, H Finney, and RThayer. (1998). *OpenPGP Message Format*. RFC 2440 Edition.
- Guntman, Peter, *Cryptography and Data Security*, University of Auckland. <http://www.cs.auckland.ac.nz/pgut001>.
- Hankerson D.R. 2000. *Condng Theory and Cryptography*, 2^{ed} Edition, New York: Marcel Dekker.
- M. Bellare, R. Canetti, and H. Karwczyk. 1996. "Keying Hash Functions For Message Authentication," *Advances in Cryptology, CRYPTO 1996 Proceedings* Springer-Verlag.
- M. Blaze, W. Diffie, R. Rivest, B. schneier, T. Shimomura, E. Thompson, and M. Weiner, 1996. "Minimal Key Lengths for Symmetric Ciphers to Provide Adequate Commercial Securty," Jan 1996.
- Nichols, R, 1999. *ICSA Guide to Cryptography*. New York: McGraw-Hill Piper, Fred and Murphy, Sean. 2002. *Cryptography: A Very Short Introduction*, Oxford University Press.
- R. Anderson and E. Biham. 1996. "Two Practical and Povably Secure Block Ciphers: BEAR and LION," *Fast Software Encryption*, Third International Workshop Proceedings, Springer-Verlag.
- R. Anderson and E. Biham. 1996. "Tiger: A Fast New Hash Function," *Fast Software Encryption*, Third International Workshop Proceedings, Springer-Verlag.
- R. F. Chruchhouse. 2002. *Code and Ciphers: Julius Caesar, the enigma and the Internet*, Cambridge University Press, 2002.
- Rhee. Man Young. 2003. *Internet Security: Cryptoghpic Principles, Algorithms, and Protocols*, England: John Wiley & Son Ltd.
- Schneier, Bruce. 1996. "The Blowfish Encryption Algorithm – One Year Later". Dr. Dobb`s Journal