Pengamanan *Data File Audio* Menggunakan Metode *Rijndael Advanced Encryption Standard*

Ngarap Im Manik[1], Yan Marchell Suharja[2]
Jurusan Matematika, Binus University[1,2]
Jl.Kebon Jeruk Raya 27, Jakarta Barat, Indonesia[1,2]
Email : manik@binus.edu[1]

Abstrak

Untuk menyelesaikan masalah keamanan pengiriman data maka cara yang paling sederhana ditempuh dengan cara melakukan enkripsi. Enkripsi dilakukan ketika data akan dikirim. Proses ini akan mengubah suatu data asal menjadi data rahasia yang tidak dapat dibaca. Sementara itu, proses dekripsi dilakukan oleh penerima data yang dikirim tersebut. Tetapi hal tersebut tidak lagi menjadi satu jaminan keamanan data. Cara lain yang biasa ditempuh selain enkripsi adalah melakukan penyamaran informasi (steganografi). Penelitian ini bertujuan untuk untuk menggabungkan dua upaya pengamanan data tersebut. Pengiriman informasi yang telah di enkripsi dengan menggunakan algoritma Rijndael Advanced Encription Standard (AES) akan disamarkan melalui media file audio dengan metode Least Significant Bit (LSB). Dengan kombinasi dua metode ini dapat mengenkripsikan data dan menyisipkannya ke dalam file .wav. Presentase keberhasilan dalam menyisipkan dan mengambil kembali data yang disisipkan mencapai sempurna 100% dan dapat menyisipkan data dengan perbandingan besar maksimal 1:8 dari file .wav yang menjadi carrier.

Kata Kunci: *Rijndael AES, Steganograpfi, Least Significant Bit (LSB)*

Abstract

To solve the security problems of data transmission is the simplest way pursued by means of encryption. Encryption is performed when data is sent. This process will transform the original data to classified data is unreadable. Mean while, the decryption process performed by a recipient who sent the data. But that is no longer a guarantee of data security. Another common way encryption is taken apart to disguise information (steganography). This study aimed to combine these two data security efforts. Transmission of information that has been encrypted using the algorithm Rijndael Advanced Encription Standard (AES) will be disguised through media audio file to the method Least Significant Bit (LSB). With the combination of these two methods can encrypt the data and insert it into the file. Wav. Percentage of success in the insert and retrieve data that is inserted to achieve a perfect 100% and can insert data with a maximum ratio of 1:8 of the file. Wav is a carrier.

Keywords*: Rijndael AES, Steganography, Least Significant Bit (LSB)*

1. Pendahuluan

Dewasa ini perkembangan teknologi komputer dan jaringan komputer, khususnya internet sangatlah cepat dan telah menjadi salah satu kebutuhan dari sebagian besar manusia. Dengan kemudahan yang diberikan teknologi tersebut, perpindahan informasi tidak lagi dibatasi, baik oleh jarak maupun waktu. Namun maraknya perkembangan teknologi juga menyebabkan pergeseran fungsi dari teknologi komputer dan jaringan oleh sebagian orang, baik hal yang sengaja maupun tidak. Hal yang cukup membahayakan dari masalah tersebut salah satunya adalah mengurangi sistem keamanan penyimpanan informasi dalam komputer yang terhubung dengan jaringan ke luar komputer sehingga gangguan-gangguan dari pihak luar dalam proses perpindahan informasi sedikit banyak tidak dapat dielakkan. Dengan tujuan meminimalkan efek gangguan tersebut telah mendorong perkembangan teknologi kriptografi dan steganografi.

Kriptografi merupakan studi matematika yang mempunyai hubungan dengan aspek keamanan informasi seperti integritas data, keaslian entitas dan keaslian data.[4][5]. Kriptografi menggunakan berbagai macam cara dalam upaya mengamankan suatu data. Pengiriman data dan penyimpanan data

melalui media elektronik memerlukan suatu proses yang dapat menjamin keamanan dan keutuhan dari data yang dikirimkan tersebut. Data tersebut harus tetap rahasia pada saat melalui proses pengiriman, dan harus utuh pada saat data tersebut sampai di tujuan. Untuk memenuhi kebutuhan tersebut, dilakukan teknik enkripsi dan dekripsi terhadap data yang akan dikirimkan.

Enkripsi dilakukan pada saat pengiriman dengan cara merubah data asli menjadi data rahasia, sedangkan proses dekripsi dilakukan pada saat proses penyampaian pesan ke tujuan dengan cara merubah data rahasia tadi kembali ke data asli. Tujuan dari dua proses ini adalah agar pada saat proses pengiriman, data asli tidak dapat diketahui oleh pihak yang tidak berkepentingan. Enkripsi dalam hal ini dapat diartikan sebagai kode atau *cipher*. Sebuah sistem pengkodean menggunakan suatu tabel atau kamus yang telah didefinisikan sebelumnya. Pengkodean dilakukan dengan algoritma tertentu untuk mengkodekan semua aliran data *bit* dari suatu pesan asli (*plaintext)* menjadi pesan rahasia *(ciphertext)*. Karena sistem ini dapat dilakukan secara otomasi, maka teknik ini dapat digunakan dalam sistem keamanan jaringan komputer.

Pada tahun 1977, *National Institute of Standard and Technology (NIST)* mengumumkan suatu algoritma standar penyandian data yaitu *Data Encryption Standard (DES)*. Kelebihan dari *DES* ini terletak pada panjan kuncinya yaitu 56-*bit*. Sejalan dengan perkembangan perangkat keras dan meluasnya penggunaan jaringan komputer mengakibatkan penggunaan *DES* menjadi tidak aman lagi [8]. Untuk memenuhi kebutuhan akan sistem keamanan yang lebih, maka *National Institute of Standard and Technology (NIST)* pada tahun 1997 mengumumkan bahwa sudah saatnya membuat standar algoritma penyandian baru yang diberi nama *Advanced Encryption Standard (AES)*. Algoritma *AES* ini dibuat dengan tujuan menggantikan algoritma *DES*. Setelah melalui beberapa tahap seleksi, algoritma *Rijndael* ditetapkan sebagai algoritma kriptografi *AES* pada tahun 2000.

Demi meningkatkan keamanan dari suatu data yang akan dikirim maka data tersebut dapat melalui proses steganografi. Kriptografi dan steganografi merupakan hal yang berbeda. Kriptografi memungkinkan pihak ketiga mengetahui adanya *ciphertext*. Sedangkan steganografi dapat menyembunyikan *ciphertext* tersebut kedalam suatu media penampung baik itu gambar, video, teks ataupun suara sehingga pihak ketiga tidak akan menyadari keberadaan *ciphertext* tersebut.[6]

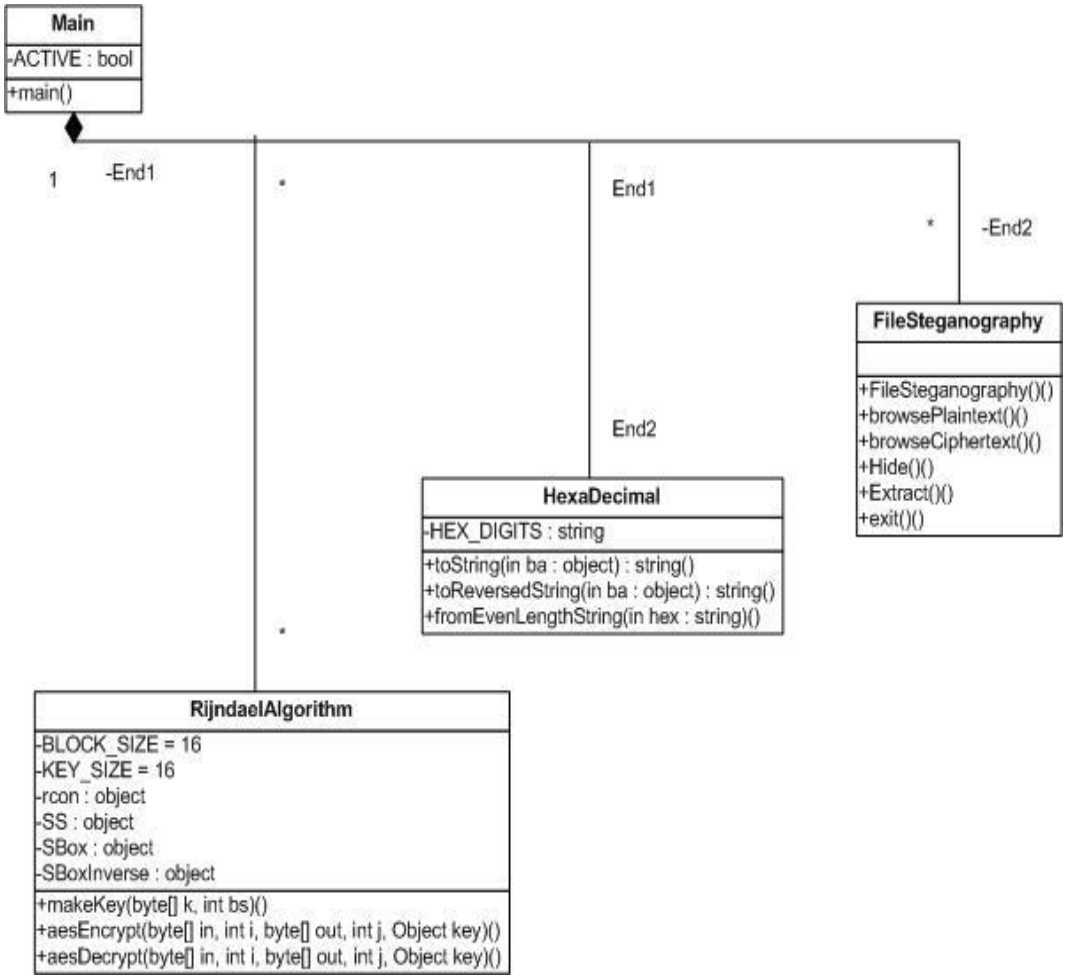
Makalah ini membahas tentang penerapan metode matematika dalam proses enkripsi dan dekripsi, untuk mengamankan data dari pihak yang tidak berkepentingan dengan mengimplementasikan kriptografi dengan metode *Rijndael Advanced Encryption Standard 128 bit* pada data berupa *plaintext* menjadi *ciphertext.* Kemudian *ciphertext* tersebut akan disamarkan dengan cara menyisipkannya ke dalam sebuah *file* audio dengan format *.wav*. Kemudian untuk memudahkan prosesnya dilakukan dengan membuat sebuah program yang dapat menggambarkan bagaimana *plaintext* dapat disimpan kedalam *file* audio tanpa merubah kualitas dari *file* audio itu sendiri.

2. Metode Penelitian

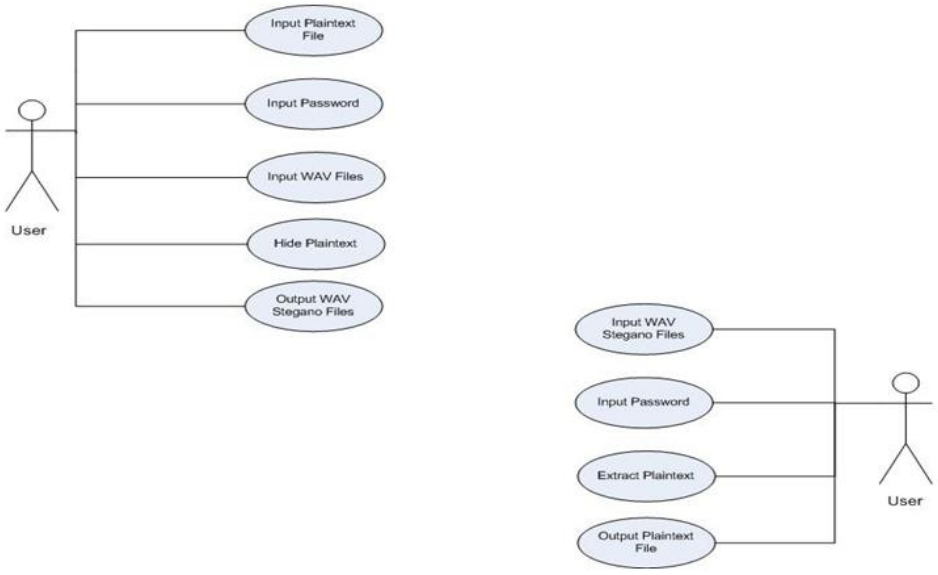
Melihat masalah keamanan data yang ada, maka penyelesaiannya dilakukan dengan merancang suatu program aplikasi yang dapat menggabungkan kriptografi dengan steganografi. Kriptografi yang akan digunakan adalah dengan menggunakan algoritma *Rijndael AES*, kemudian hasil dari enkripsi tersebut akan disisipkan ke dalam audio *file* dengan metode Least Significant Bit. Medium audio *file* yang dipilih adalah file yang bertipe WAV, karena merupakan tipe *uncompressed audio file* yang mempunyai jumlah bit yang cukup banyak, sehingga dapat memuat jumlah pesan dalam jumlah yang cukup besar, dan hasil audio yang telah disisipkan juga tidak akan terlalu berpengaruh pada kualitasnya.[**6][8**].

Perancangan program ini menggunakan konsep *Object Oriented Programming* untuk mengembangkannya, oleh karena itu dipilih *Unified Modeling Language* (*UML*) untuk merancang arsitektur program. *UML* yang digunakan meliputi perancangan *class diagram* dan *sequence diagram*.[7][2]

Class diagram adalah diagram yang menggambarkan class-class yang digunakan dalam perancangan program dan hubungan antar class seperti terlihat pada gambar 1. sedangkan hubungan dengan pengguna dapat digambarkan seperti pada gambar 2.



Gambar 1. *Class Diagram*

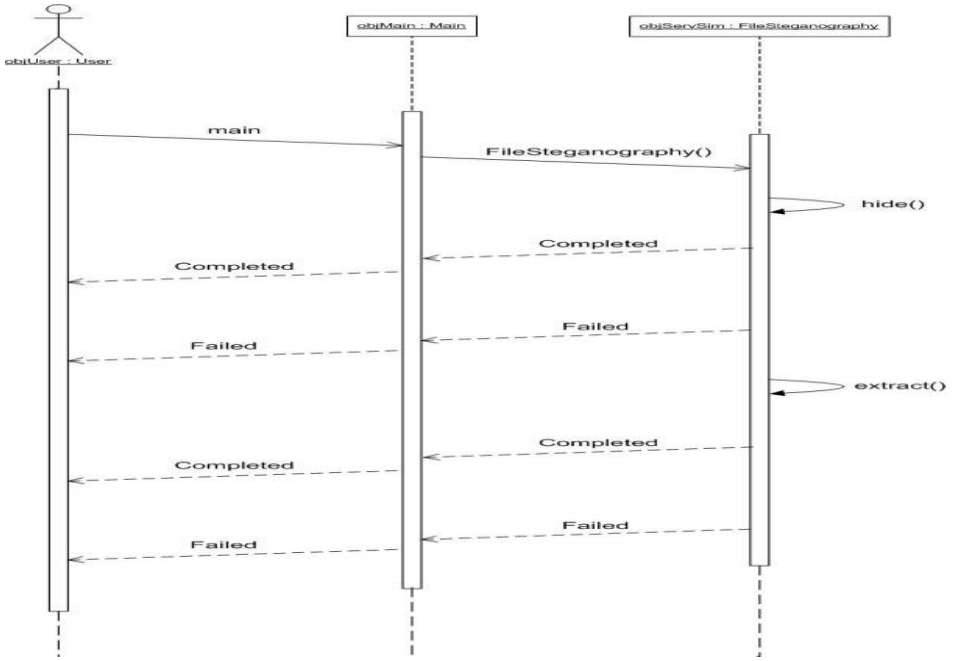


Gambar 2. *Use Case Diagram File Steganography*

Pada *Use Case Diagram* dapat dilihat, pertama *user* memilih file plain- text yang akan dienkripsi dan disisipkan. User juga harus mengisi password dalam format hexadecimal. File carrier, dan file output steganografi juga harus dipilih. Setelah itu, dengan menekan tombol Hide, maka proses enkripsi dengan metode Rijndael AES dan steganografi dengan menggunakan metode LSB akan jalan.[**1][3**].

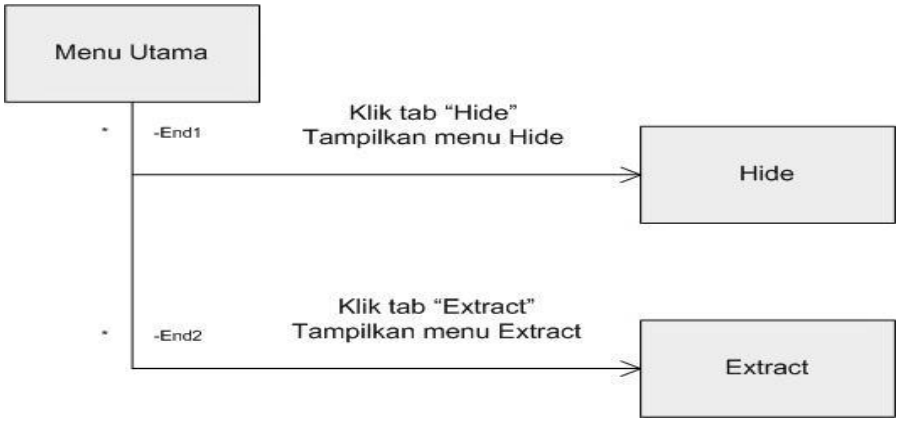
Pada posisi si penerima pesan. User harus menginputkan nama file wav yang akan di ekstrak, password yang diisi juga harus sama dengan password yang diisi pada saat penyisipan data, dan yang terakhir user harus mengisi nama file yang akan menjadi file output. Dengan menekan tombol Extract, maka proses dekripsi dan proses LSB akan jalan.

Sequence diagram adalah diagram yang menunjukkan urutan penukaran pesan oleh sejumlah object (dan seorang aktor yang optional) di dalam melakukan tugas tertentu.Hal ini dapat dilihat dalam gambar 3.

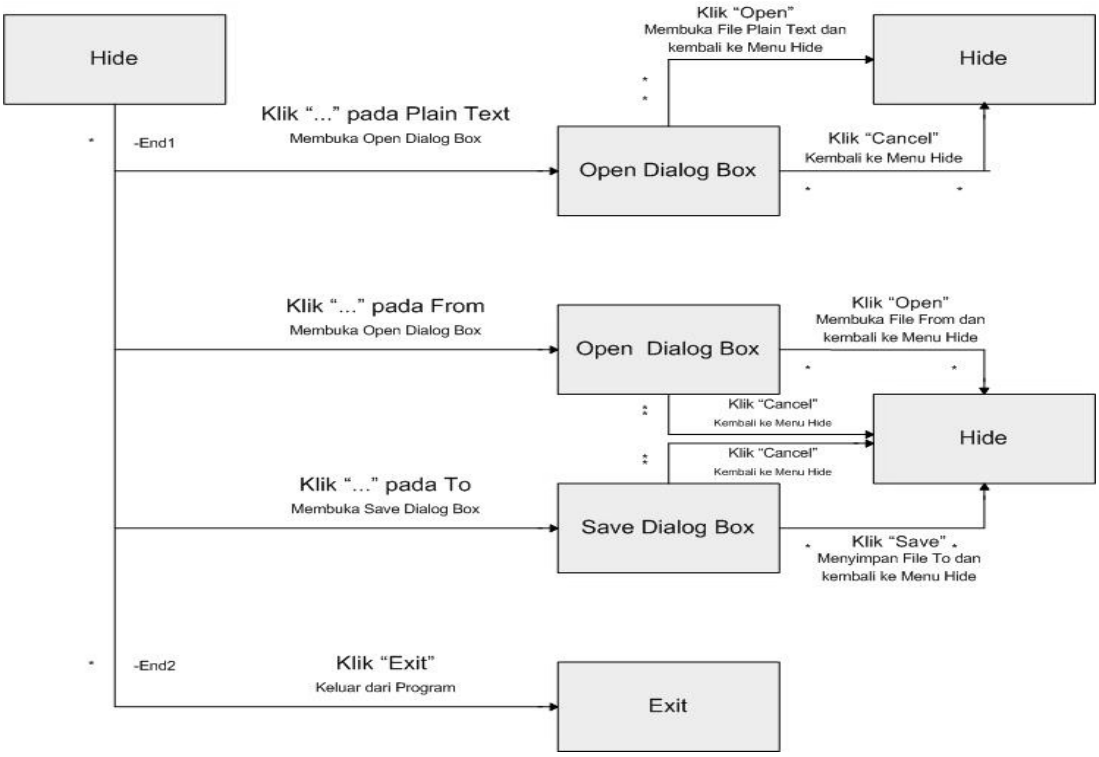


Gambar 3. *Sequence Diagram File Steganography*

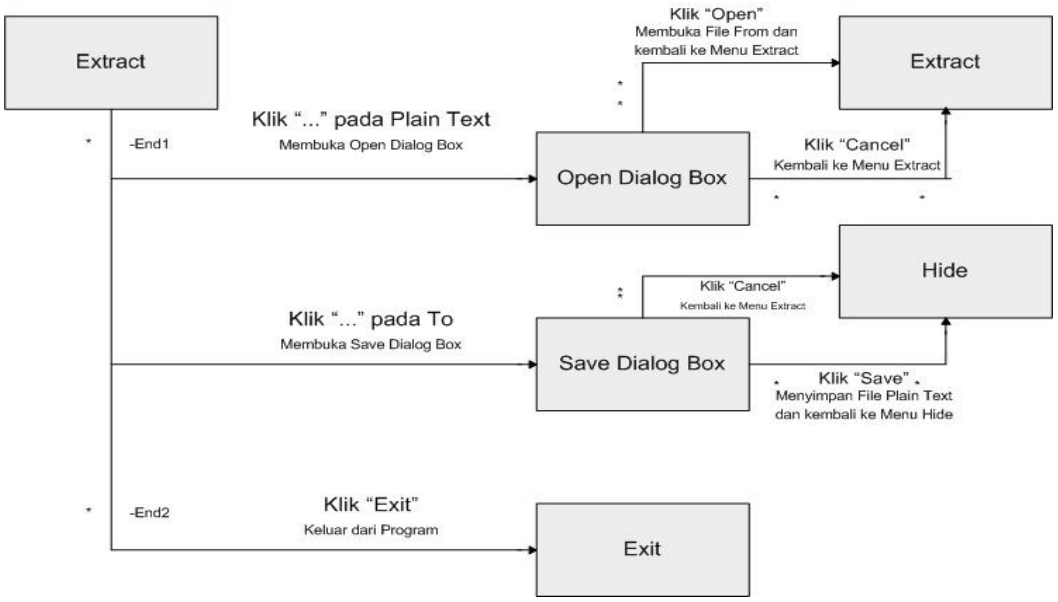
STD atau *State Transition Diagram* menggambarkan sebuah sistem yang *real-time* dan sistem yang *on-line*. STD merupakan suatu keadaan yang menggambarkan suatu keadaan pada waktu tertenu (Yourdon, 2006). Perubahan keadaan dapat terjadi karena suatu kejadian dan sebagai akibat dari kejadian tersebut maka akan muncul suatu aksi yang menyebabkan keadaan berubah. STD membantu dalam memberikan gambaran secara keseluruhan dari program. STD untuk program makalah ini dapat dilihat pada gambar 4, 5 dan gambar 6.



Gambar 4. *State Transition Diagram* Menu Utama



Gambar 5. *State Transition Diagram tab Hide*



Gambar 6. *State Transition Diagram tab Extract*

Selanjutnya untuk mengatur program diatur melalui menu-menu yang terdapat pada program ini adalah seperti yang dijelaskan pada tabel 1:

Tabel 1. Menu-Menu Program Aplikasi

Menu	Tujuan
File Steganography	Membuka layar *File Steganography*
Tab Hide	Membuka layar *Hide*
Tab Extract	Membuka layar *Extract*
Exit	Keluar dari program

3. Hasil dan Analisis

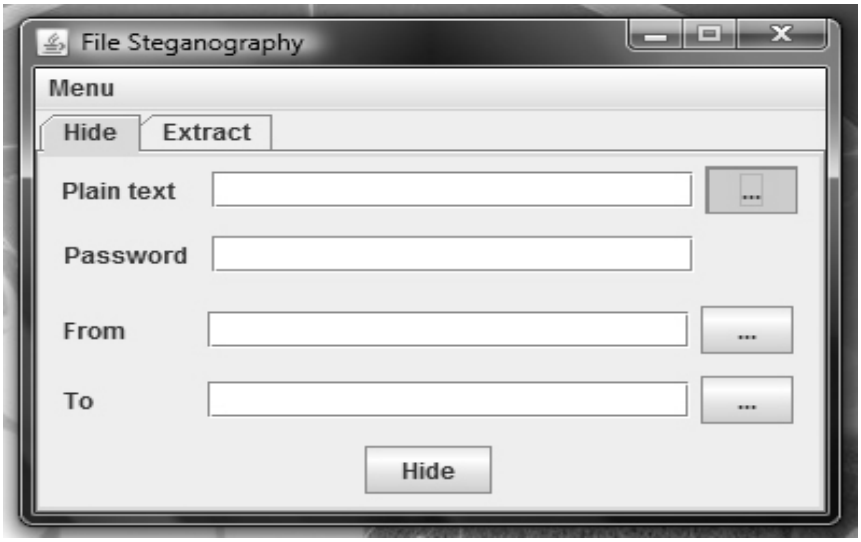
Untuk memberikan hasil keluaran dari program dalam penelitian ini diperlukan spesifikasi program seperti berikut *Processor: Intel Pentium, Core 2 Duo T5500 @1.66GHz 1.67 GHz, Memory: 2,0 Giga Bytes dan* Sistem Operasi: *Windows Vista Home Premium Service Pack* . Sedangkan pemrograman yang digunakan adalah: *Platform: Textpad 4.73, Java 2 Standard Development Kit version 1.8 dan* Bahasa Pemrograman: *Java 2 Standard Edition*

Pada saat program pertama kali dijalankan, akan muncul *form* Menu Utama dengan tab panel Hide terbuka seperti pada gambar 7. Pada *form* ini *user* dapat memilih menu *Hide,* atau *Extract.*

Gambar 7. Tampilan Layar Utama

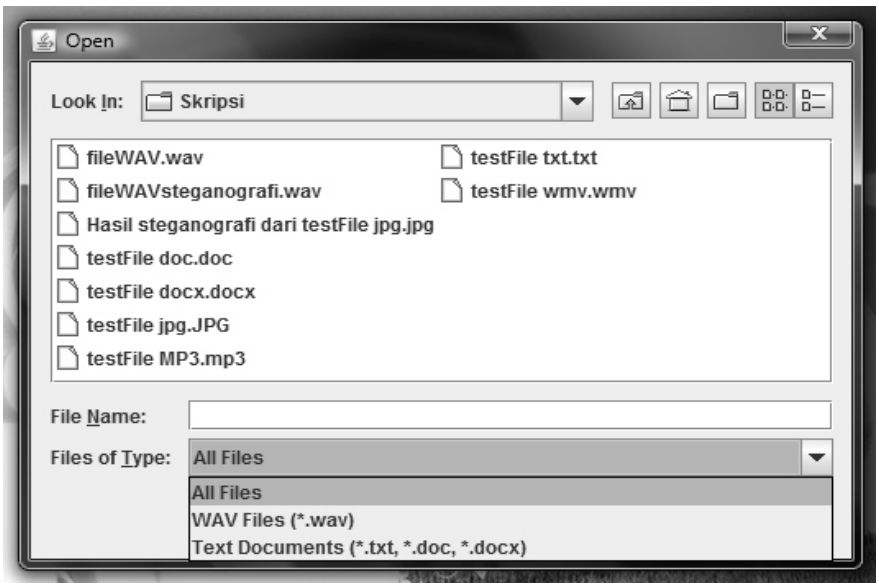
Hiding Plain Text

Jika *user* ingin melakukan *insert plain text* maka *user* harus memilih dulu *file plain text* yang akan disisipkan, dengan cara menekan tombol "..." yang sejajar dengan *textfield plain text,* seperti pada gambar 8.



Gambar 8. Layar Tombol "..." Pada *Plain Text* Di Tekan

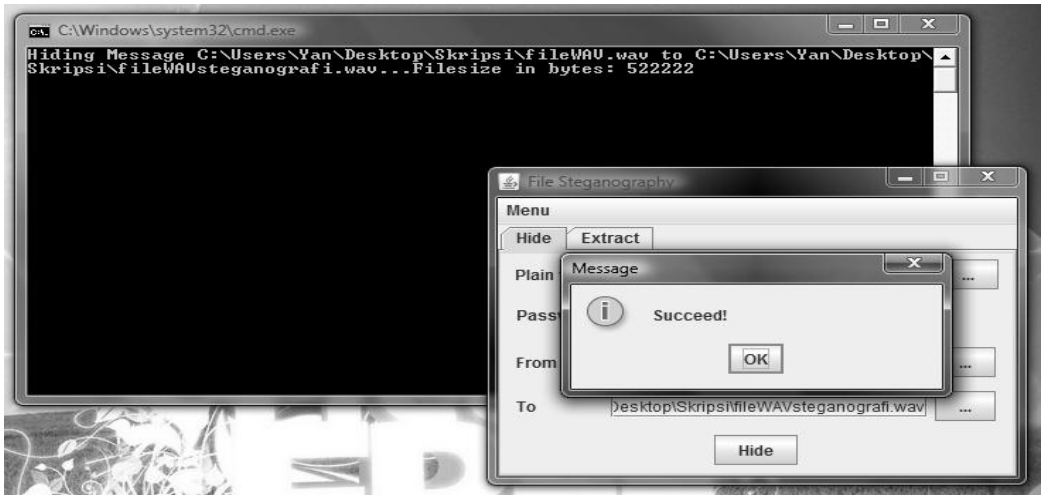
Ketika *user* menekan tombol "...", maka secara otomotis akan muncul Open File Dialog seperti pada gambar 9. *User* dapat memilih file mana yang akan menjadi *plain text* yang akan disisipkan.



Gambar 9. *Open Plain Text Dialog Box*

Setelah memilih file mana yang akan *user* sisipkan, maka *user* harus mengisikan kata kunci *(password)*, yang akan menjadi *key* pada proses enkripsi menggunakan algoritma *Rijndael AES* berupa kombinasi dari karakter *hexadecimal. Password* ini akan digunakan untuk mendapatkan kembali data yang disembunyikan di dalam *file .wav*. Selanjutnya *user* harus memilih *carrier file* berformat .wav, serta user juga harus menentukan tempat hasil dari proses steganografi ini dalam bentuk .wav juga.

Setelah *user* mengisi semua *field* yang dibutuhkan, maka untuk menjalankan proses steganografi, *user* hanya tinggal menekan tombol "Hide" dan tunggu sampai proses selesai. Hasil proses dapat dilihat pada *command windows* yang menampilkan pesan bahwa *file* tersebut sukses untuk disisipkan, seperti pada gambar 10.



Gambar 10. Informasi *File* Sukses Disisipkan

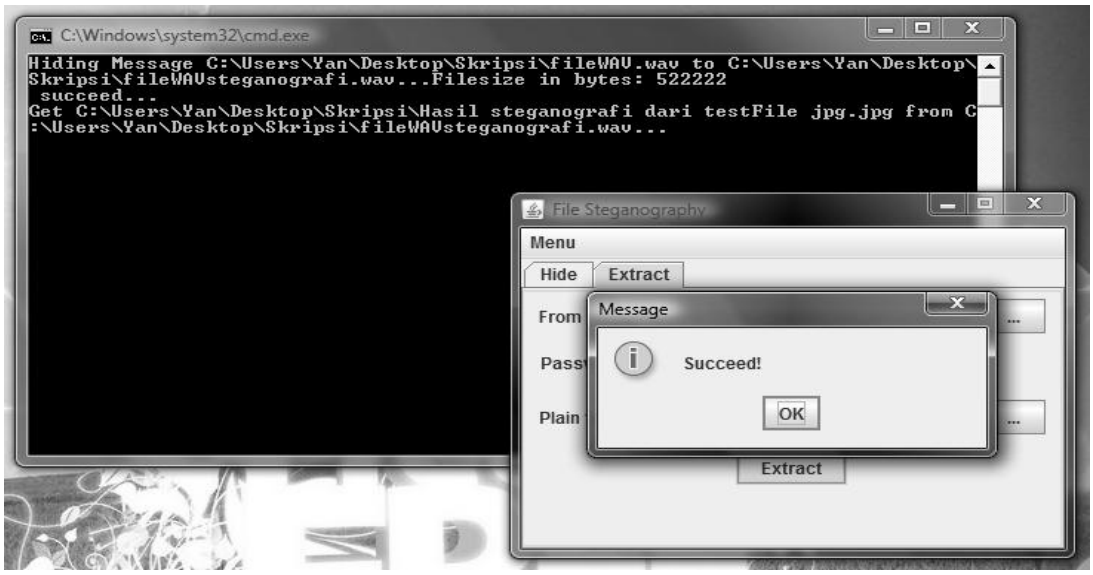
Extracting Plain Text

Jika user ingin melakukan extract plain text maka user harus memilih tab Extract terlebih dahulu, sehingga layar utama berubah tampilan seperti gambar 11.

Gambar 11. Tampilan Layar *Extract*

Untuk melakukan proses *extract plain text*, mula-mula *user* harus memilih *file .wav* yang berisi *plain text* dengan memilih tombol "..." pada baris *From* . Ketika *user* telah memilih *file .wav*, maka akan tampil informasi mengenai *file .wav* yang dipilih. Setelah itu dilanjutkan dengan memasukkan *password* yang sesuai dengan *password* yang ditulis pada saat proses enkripsi.

Kemudian *user* harus memasukkan nama file yang diinginkan untuk menampung data steganografi yang di*extract.* Setelah itu, user harus menekan tombol Extract, maka program aplikasi akan melakukan proses dekripsi menggunakan algoritma *Rijdael AES,* serta melakukan proses ekstraksi dari data *file .wav.* Pada saat proses ekstraksi sukses, akan muncul pesan sukses pada command windows seperti yang terlihat pada gambar 12.



Gambar12. Informasi *File* Sukses Diekstraksi

Program dijalankan dengan memakai inputan *file .wav* sebesar 36.4 *MB* (38.178.592 *Bytes) dan beberapa jenis file* data diperoleh seperti berikut ini.

Tabel 2. Hasil Percobaan

Jenis *File*	***Extension File***	**Besar *File***	**Dapat di-*extract***
Teks	*.txt*	5.100 *Bytes*	Ya
Teks	*.doc*	48.640 *Bytes*	Ya
Teks	*.docx*	23.849 *Bytes*	Ya
Gambar	*.jpg*	522.222 *Bytes*	Ya
Musik	*.mp3*	3.791.284 *Bytes*	Ya
Video	*.wmv*	2.797.732 *Bytes*	Ya

Perbandingan besar data plainteks maksimum dengan besar data medium adalah 1:8. Perbandingan besar hasil ekstraksi data untuk teks, gambar, musik, dan video dengan besar data medium adalah 1:8.

4. Kesimpulan

Hasil percobaan program aplikasi steganografi data file audio menggunakan metode *Rijndael Advanced Encryption Standard (EAS) y*ang dibuat dapat mengenkripsikan data dan menyisipkannya ke dalam *file* .wav dengan presentase keberhasilan program dalam menyisipkan dan mengambil kembali data yang disisipkan adalah 100%. Berdasarkan percobaan, program aplikasi ini dapat menyisipkan data dengan perbandingan besar maksimal 1: 8 dari *file* .wav yang menjadi *carrier dan u*kuran besar dari *file* yang diesktrak adalah 1: 8 dari besar ukuran *file* .wav yang menjadi *carrier*.

Daftar Pustaka

[1] Abraham, DG dan Dolan, GM dan Double, GP dan Stevens, JV (2001). IBM Systems Journal v 30 no 2.
[2] Bennet, Simon, McRobb, Steve, Farmer, Ray (2005). Object-Oriented System Analysis and Design Using UML, Third Edition, Sine Nomine.

[3] Booch G, Rumbaught J, Jacobson I (2001). The Unified Modelling Language User Guide, Addison Wesley, Longman Inc., USA.
[4] Biham, E (2004). Journal of Cryptography v 7.
[5] Davies, DW dan Murphy, S (2005). Pairs and Triplets of DES S-Boxes, Journal of Cryptology version 8.
[6] Hansfeld,Nils. The Cryptography Tutorial. http://www.antilles.k12.vi.us/math/ Akses: November 2009.
[7] Lethbridge, Timothy C., Laganiere, Robert (2002). Object-Oriented Software Engineering: Practical Software Development Using UML and Java. McGraw-Hill, New York.
[8] Stallings, William (2006). Cryptography and Network Security Third Edition, Pearson International Edition. Canada U.S.A.
[9] Trappe, Wade (2002). Introduction to Cryptography with Coding Theory. Lawrence C Washington, New York.