IMPLEMENTASI ALGORITMA IDEA DAN METODE END OF FILE PADA GAMBAR UNTUK MENYEMBUNYIKAN PESAN

Dyah Ayu Irawati [1], Mungki Astiningrum[2,] Elistya Rahma Dinda[3]

[1,2,3]Teknik Informatika, Teknologi Informasi, Politeknik Negeri Malang
[1]dyah.ayu@polinema.ac.id, [2] mungki.astingrum@polinema.ac.id, [3]elstdinda@gmail.com

Abstrak

Adanya media komunikasi menjadi salah satu hal penting dalam bertukar informasi. Seiring pesatnya perkembangan teknologi informasi saat ini memberikan pengaruh yang cukup besar bagi kehidupan manusia. Berbagai teknologi tak terkecuali media komunikasi dapat membantu seseorang dalam bertukar informasi. Dengan kemudahan akses yang diberikan tak dipungkiri dapat terjadi masalah keamanan dan kerahasiaan informasi. Berdasarkan masalah tersebut, diperlukan sebuah metode yang dapat membantu mengolah informasi menjadi pesan tidak terbaca yaitu kriptografi. Agar pesan yang terkriptografi tidak dapat diketahui maka diperlukan pula sebuah metode untuk menyembunyikan pesan yaitu steganografi. Dalam penelitian ini dilakukan salah satu teknik untuk mengamankan data yaitu dengan mengiplementasikan kriptografi algoritma IDEA (*International Data Encryption Algorithm*) yang dikombinasikan steganografi dengan metode End of File. Dari hasil uji coba, diketahui bahwa jika IDEA digabungkan dengan metode End of File (EOF) maka proses enkripsi, dekripsi, penyisipan dan ekstraksi pesan dapat dilakukan dengan baik. Sehingga pesan asli yang disandikan berupa teks atau karakter dengan gambar sebagai objek untuk menyembunyikan pesan dapat menutup kecurigaan dari pihak lain.

Kata kunci : Steganografi, *IDEA, End of File*

1. Pendahuluan

Komunikasi merupakan kebutuhan manusia yang sangat penting. Tanpa komunikasi kita tidak dapat berhubungan dengan orang lain dari jarak yang jauh. Maka dari itu, dibuatlah suatu sarana sebagai penyalur pada manusia untuk mengirim atau menerima informasi yaitu media komunikasi. Seiring pesatnya perkembangan teknologi informasi saat ini memberikan pengaruh yang cukup besar bagi kehidupan manusia dalam bertukar informasi.

Tidak dapat dipungkiri jika pengaksesan informasi yang mudah bagi semua penggunanya dapat memberikan dampak bagi keamanan informasi atau pesan yang menggunakan media komunikasi tersebut. Dampak yang dimaksud adalah masalah keamanan dan kerahasiaan informasi bagi penggunanya yang terhubung dengan jaringan publik misalnya internet. Tentu saja informasi penting tersebut jangan sampai jatuh ke tangan yang tidak berwenang sehingga bisa dilihat atau dimanipulasi oleh orang yang tidak berwenang tersebut sehingga dapat membahayakan kedua belah pihak maupun orang banyak.

Untuk menghindari hal tersebut, maka perlu informasi penting harus diolah ke bentuk lain guna untuk melindungi kerahasiaan dari suatu data itu sendiri. Berdasarkan masalah tersebut, diperlukan sebuah metode yang dapat membantu mengolah informasi menjadi pesan tidak terbaca yaitu kriptografi. Sedangkan steganografi menjadi salah satu alternatif untuk menambah keamanan yang telah didapat dari kriptografi, steganografi adalah teknik menyembunyikan data rahasia di dalam wadah (media) digital sehingga keberadaan data rahasia tersebut tidak diketahui oleh orang lain.

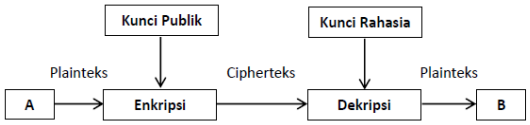
Dalam penelitian ini, digunakan alah satu teknik kriptografi yang dapat digunakan adalah algoritma IDEA, sedangkan untuk steganografinya menggunakan End of File. Diharapkan dengan penelitian ini keduanya dapat digunakan secara bersamaan untuk menjamin keamanan pesan rahasianya.

2. Tinjauan Pustaka

2.1 Kriptografi

Kriptografi merupakan ilmu yang mempelajari teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, antektikasi, integritas dan keabsahan data. Kriptografi juga dapat diartikan sebagai ilmu untuk menjaga kerahasiaan pesan G.M, Marakas (2006).

Kemudian, proses yang akan dibahas dalam penelitian ini meliputi 2 proses dasar pada kriptografi yaitu enkripsi dan dekripsi. Enkripsi yaitu proses mengubah data asli (plain text) menjadi pesan yang tidak dapat dibaca (chiper text), sedangkan dekripsi merupakan proses menjadikan data hasil manipulasi menjadi data asli. Pada prinsipnya, kriptografi memiliki 4 komponen utama yaitu *plain text*, yaitu pesan yang dapat dibaca. *Cipher text*, yaitu pesan acak yang tidak dapat dibaca. *Key*, yaitu kunci untuk melakukan teknik kriptografi. *Algorithm*, yaitu metode untuk melakukan enkrispi dan dekripsi.



Gambar 1. Proses Penyisipan pada Kriptografi

2.2 Steganografi

Steganografi merupakan seni komunikasi rahasia dengan menyembunyikan pada objek yang tampaknya tidak mencurigakan atau berbahaya. Steganografi bekerja dengan cara menyisipkan informasi atau pesan rahasia pada objek lain. Jika dibandingkan dengan kriptografi, steganografi mempunyai kelebihan menghindari kecurigaan pesan yang disisipkan, namun sebenarnya steganografi adalah pelengkap dari kriptografi. Contoh objek yang dapat digunakan sebagai objek penyisipan adalah gambar, audio, dan video.

Komponen yang harus dimiliki oleh steganografi yaitu sebagai berikut Harianto (2013):

- Pesan yang disembunyikan (embedded).
- Pesan yang menyembunyikan (cover text / cover object).
- Objek yang menyembunyikan pesan rahasia (stego object).
- Kunci yang digunakan untuk penyisipan (stego key).

Untuk menghasilkan steganografi yang baik ada 3 kriteria yang harus diperhatikan Munir (2017), yaitu :

- Imperceptibility.
  Keberadaaan pesan rahasia tidak bisa dikenali oleh indra manusia. Misalnya, jika convertext berupa citra maka penyisipan pesan membuat citra stegotext sukar dibedakan oleh mata dengan citra covertextnya.
- Fidelity.
  Mutu stegomedium tidak berubah banyak akibat penyisipan. Misalnya, jika convertext berupa citra maka penyisipan pesan membuat citra stegotext sukar dibedakan oleh mata dengan citra covertextnya.
- Recovery.
  Pesan yang disembunyikan harus dapat dikenali kembali. Karena tujuan steganografi adalah data hiding, maka sewaktu-waktu pesan rahasia di dalam stegotext harus dapat diambil kembali untuk digunakan.

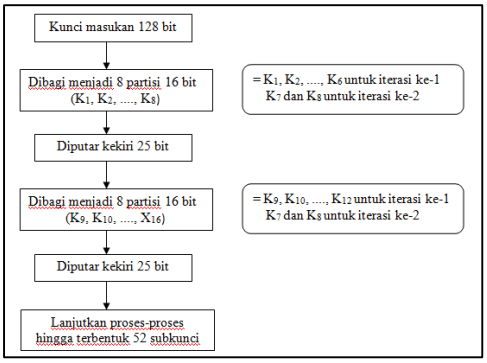
2.3 *International Data Encryption Algorithm* (IDEA)

International Data Encryption Algorithm (IDEA) merupakan salah satu algoritma simetris yang beroperasi pada sebuah blok pesan 64bit, menggunakan kunci 128bit untuk proses enkripsi dan dekripsi. Keluaran dari algoritma ini adalah blok pesan terenkripsi 64bit S.Kromodimoeljo (2009). Proses dekripsi menggunakan blok penyandian (algoritma) yang sama dengan proses enkripsi dimana kunci dekripsinya diturunkan dari kunci enkripsi. Algoritma ini menggunakan operasi campuran dari tiga operasi aljabar yang berbeda, yaitu :

- Operasi XOR, operasi ini disimbolkan dengan tanda $\oplus$.
- Operasi penjumlahan modulo 2^{16}, disimbolkan dengan tanda $\boxplus$.
- Operasi perkalian modulo $(2^{16} + 1)$, disimbolkan dengan tanda $\odot$.

Untuk memulai perhitungan algoritma IDEA yang harus dilakukan adalah :

a. Pembentukan kunci



Gambar 2. Pembentukan Kunci IDEA

Dari Gambar 2 dapat dilihat blok kunci 128 bit dipartisi menjadi 8 subkunci 16 bit yang langsung dipakai sebagai 8 subkunci pertama, dimana 6 subkunci digunakan untuk tahap pertama dan 2 subkunci berikutnya untuk iterasi ke-2. Jadi pada tahap ke-2 ini, terdapat kekurangan 4 subkunci. Kemudian blok kunci 128 bit digeser kekiri 25 bit secara memutar (rotasi) untuk dipartisi lagi menjadi 8 subkunci 16 bit berikutnya. Empat subkunci pertama digunakan untuk iterasi ke-2, melengkapi kekurangan sebelumnya. Empat subkunci berikutnya untuk iterasi ke-3. Kemudian kunci 128 bit dirotasi lagi ke kiri sebanyak 25 bit untuk mendapatkan 8 subkunci berikutnya. Proses rotasi dan partisi itu diulangi lagi sampai diperoleh 52 subkunci 16 bit.

Sub kunci untuk proses enkripsi :

Tabel 6. Sub Kunci Proses Enkripsi

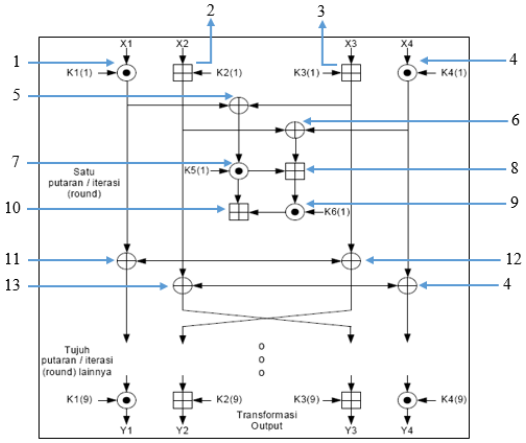
Putaran ke 1	$K_1, K_2, K_3, K_4, K_5, K_6$
Putaran ke 2	$K_7, K_8, K_9, K_{10}, K_{11}, K_{12}$
Putaran ke 3	$K_{13}, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}$
Putaran ke 4	$K_{19}, K_{20}, K_{21}, K_{22}, K_{23}, K_{24}$
Putaran ke 5	$K_{25}, K_{26}, K_{27}, K_{28}, K_{29}, K_{30}$
Putaran ke 6	$K_{31}, K_{31}, K_{33}, K_{34}, K_{35}, K_{36}$
Putaran ke 7	$K_{37}, K_{38}, K_{39}, K_{40}, K_{41}, K_{42}$
Putaran ke 8	$K_{43}, K_{44}, K_{45}, K_{46}, K_{47}, K_{48}$
Transformasi Output	$K_{49}, K_{50}, K_{51}, K_{52}$

Sedangkan sub kunci untuk proses deskripsi :

Tabel 2. Sub kunci proses deskripsi

Putaran ke 1	$(K_{49})^{-1}$; - K_{50}; - K_{51}; $(K_{52})^{-1}$; K_{47}; K_{48};
Putaran ke 2	$(K_{43})^{-1}$; - K_{44}; - K_{45}; $(K_{46})^{-1}$; K_{41}; K_{42};
Putaran ke 3	$(K_{37})^{-1}$; - K_{38}; - K_{39}; $(K_{40})^{-1}$; K_{35}; K_{36};
Putaran ke 4	$(K_{31})^{-1}$; - K_{32}; - K_{33}; $(K_{34})^{-1}$; K_{29}; K_{30};
Putaran ke 5	$(K_{25})^{-1}$; - K_{26}; - K_{27}; $(K_{28})^{-1}$; K_{23}; K_{24};
Putaran ke 6	$(K_{19})^{-1}$; - K_{20}; - K_{21}; $(K_{22})^{-1}$; K_{17}; K_{18};
Putaran ke 7	$(K_{13})^{-1}$; - K_{14}; - K_{15}; $(K_{16})^{-1}$; K_{11}; K_{12};
Putaran ke 8	$(K_{7})^{-1}$; - K_{8}; - K_{9}; $(K_{10})^{-1}$; K_{5}; K_{6};
Transformasi Output	$(K_{1})^{-1}$; - K_{2}; - K_{3}; $(K_{4})^{-1}$

b. Enkripsi



Gambar 3. Proses Enkripsi IDEA

Proses enkripsi algoritma IDEA adalah sebagai berikut, Pertama, plaintext 64 bit dibagi menjadi 4 buah sub blok dengan panjang 16 bit, yaitu X1, X2, X3, X4. Empat sub blok ini menjadi masukan bagi iterasi tahap pertama algoritma. Total terdapat 8 iterasi. Pada setiap iterasi, 4 sub blok di-XOR-kan, ditambahkan, dikalikan dengan yang lain dan dengan 6 buah subkey 16 bit. Diantara iterasi sub blok kedua dan ketiga saling dipertukarkan. Akhirnya 4 buah sub blok dikombinasikan dengan 4 subkey dalam transformasi output. Pada setiap tahapan, urutan berikut ini dikerjakan :

1. Mengalikan X1 dengan K1 mod $(2^{16} + 1)$.
2. Menambahkan X2 dengan K2 mod 2^{16} .
3. Menambahkan X3 dengan K3 mod 2^{16}.
4. Mengalikan X4 dengan K4 mod $(2^{16} + 1)$.
5. Melakukan XOR hasil dari step 1 dan 3.
6. Melakukan XOR hasil dari step 2 dan 4.
7. Mengalikan hasil step 5 dengan K5 mod $(2^{16} + 1)$.
8. Menambahkan hasil step 6 dan 7 mod 2^{16}.
9. Mengalikan hasil step 8 dengan K6 mod $(^{216} + 1)$.
10. Menambahkan hasil dari step 7 dan 9.
11. Melakukan XOR hasil dari step 1 dan 9.
12. Melakukan XOR hasil dari step 3 dan 9.
13. Melakukan XOR hasil dari step 2 dan 10.
14. Melakukan XOR hasil dari step 4 dan 10.

Langkah terakhir dari proses enkripsi IDEA yaitu keempat sub blok 16 bit yang merupakan hasil operasi 1), 2), 3), dan 4) digabung kembali menjadi blok pesan rahasia 64 bit.

c. Dekripsi

Proses dekripsi sama persis dengan proses enkripsi. Perbedaannya hanya terletak pada aturan dari subkey-nya. Urutan subkey terbalik dengan proses enkripsi dan subkey-nya di-inverse-kan. Subkey pada langkah transformasi output pada proses enkripsi di-inverse-kan dan digunakan sebagai subkey pada putaran 1 pada proses dekripsi. Subkey pada putaran 8 di-inverse-kan dan digunakan sebagai subkey pada putaran 1 dan 2 pada proses dekripsi. Demikian seterusnya.

Proses dekripsi menggunakan algoritma yang sama dengan proses enkripsi tetapi 52 buah sub blok kunci yang digunakan masing-masing merupakan hasil turunan 52 buah sub blok kunci enkripsi. Pada kasus ini akan diambil invers dari operasi penambahan oleh modulo 2^{16}(65536) dan perkalian modulo 2^{16} +1 (65537), tergantung pada operasi yang dibuat pada fase enkripsi. Setiap subkunci dekripsi adalah salah satu dari invers penambahan atau perkalian yang berkorespondensi dengan sub kunci enkripsi.

2.4 Metode *End of File* (EOF)

Metode End of File merupakan salah satu teknik yang dapat digunakan di dalam steganografi. Metode ini menggunakan cara dengan menyisipkan data pada akhir file. Metode ini juga dapat digunakan untuk menyisipkan data yang ukurannya sesuai dengan kebutuhan. Ukuran file yang telah disisipkan data,sama dengan ukuran file sebelum disisipkan data ditambah dengan ukuran data yang disisipkan ke dalam file tersebut. Proses yang terjadi dalam penyisipan pesan dengan metode EoF adalah dengan mengubah pesan menjadi kode desimal, dapatkan nilai atau letak pixel terakhir dari citra, berikan sebuah tanda pengenal start dari pesan dan tambahkan kode desimal dari pesan. Pada proses pengungkapan pesan, maka proses yang diperlukan adalah mengenali letak tanda pengenal dan mengambil nilai desimal dari pesan rahasia serta terakhir mengubah nilai desimal menjadi sebuah pesan Munir (2017).

Menurut Paskalis A. N. proses penyisipan pesan dengan metode End of File ini dapat ditulisan sebagai berikut Munir (2006):

1. Inputkan pesan yang akan disisipkan.
2. Ubah pesan menjadi kode-kode desimal.
3. Inputkan citra grayscale yang akan disisipi pesan.
4. Dapatkan nilai derajat keabuan masing-masing piksel.
5. Tambahkan kode desimal pesan sebagai nilai derajat keabuan diakhir citra.
6. Petakan menjadi citra baru.

2.5 *Visual Basic*

Visual Basic merupakan bahasa pemrograman yang menawarkan Integrated Development Environment (IDE) visual untuk membuat program perangkat lunak/ aplikasi berbasis sistem operasi Microsoft Windows yang berbasis GUI (Grapical User Interface). Microsoft Visual Basic 6.0 hampir dapat memanfaatkan seluruh kemudahan dan kecanggihan yang dimiliki oleh sistem operasi windows yaitu menyediakan komponen-komponen yang memungkinkan untuk membuat program aplikasi yang sesuai dengan tampilan dan cara kerja windows.

Menurut Daryanto (2003:13) Visual Basic adalah salah satu development tools untuk membangun aplikasi dalam lingkungan windows. Dalam perkem-bangan aplikasi, Visual Basic menggunakan pendekatan visual untuk merancang user interface dalam bentuk form, sedangkan untuk kodingnya menggunakan dialok bahasa basic. Visual Basic telah menjadi tools yang terkenal bagi para pemula maupun para developer.

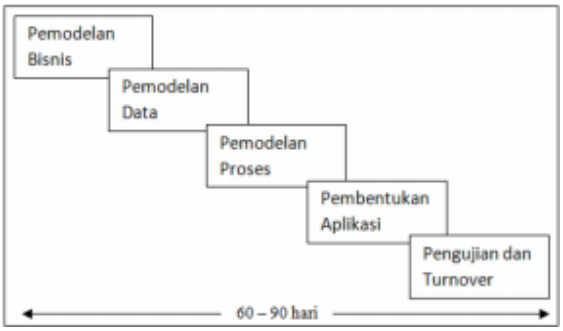
3 Metodologi Penelitian

3.1 Metode Pengumpulan Data

Dalam penelitian ini data yang digunakan merupakan data sekunder. Penulis memperoleh data dari telaah pustaka dan artikel-artikel yang penulis dapat dari pustaka yang mendukung, informasi dari internet, dan jurnal-jurnal.

3.2 Metode Pengembangan Sistem

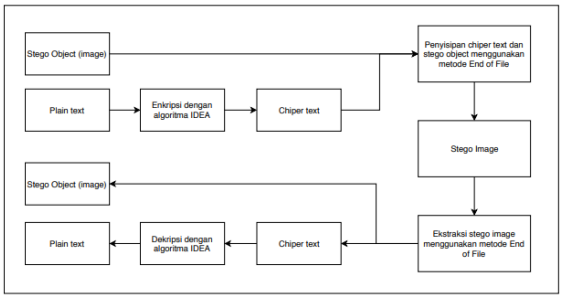
Rapid Aplication Development (RAD) adalah sebuah metode pengembangan software yang diciptakan untuk menekan waktu yang dibutuhkan untuk mendesain serta mengimplementasikan sistem informasi sehingga dihasilkan siklus pengembangan yang sangat pendek. Model RAD ini merupakan adaptasi dari model sekuensial linier dimana perkembangan yang cepat dicapai dengan menggunakan pendekatan kontruksi berbasis komponen. Sehingga, jika kebutuhan dipahami dengan baik, proses RAD memungkinkan developer menciptakan sistem fungsional yang utuh dalam periode waktu yang sangat pendek (± 60 sampai 90 hari) seperti pada gambar..



Gambar 4. Fase-fase RAD

3.3 Desain dan Perancangan Sistem

Berikut ini adalah blok diagram sistem kerja dari perangkat lunak penggabungan anatara kriptografi dengan Algoritma IDEA dan metode steganografi End of File yang penulis rancang.

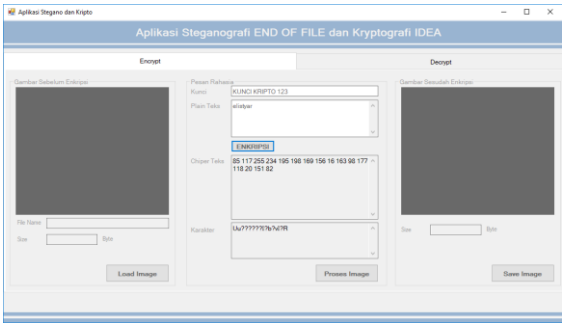


Gambar 5. Blok Diagram Kriptografi IDEA dan End of File

Pada Gambar 5 merupakan proses kombinasi kriptografi IDEA dan steganografi End of File. Teks asli (plain text) diubah menjadi biner untuk dienkripsi terlebih dahulu dengan menggunakan IDEA sebelum disisipkan ke dalam citra/gambar/cover-image. Hasil keluaran dari proses enkripsi berupa pesan terekripsi (chiper text). Kemudian chiper text diubah menjadi bilangan desimal yang akan disisipkan ke dalam citra/gambar/plain-image menggunakan metode End of File. Keluaran yang muncul setelah proses penyisipan dengan End of File yaitu gambar yang telah disisipi (stego image). Untuk mengembalikan menjadi teks asli, dilakukan ekstraksi stego image menggunakan metode yang sama yaitu End of File yang menghasilkan cover image dan chiper text secara terpisah. Selanjutnya, chiper text didekripsi dengan menggunakan IDEA sehingga didapatkan plain text.

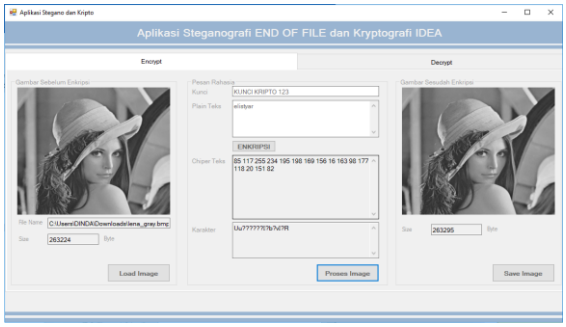
4 Pengujian Sistem

4.1 Pengujian Input Output Program



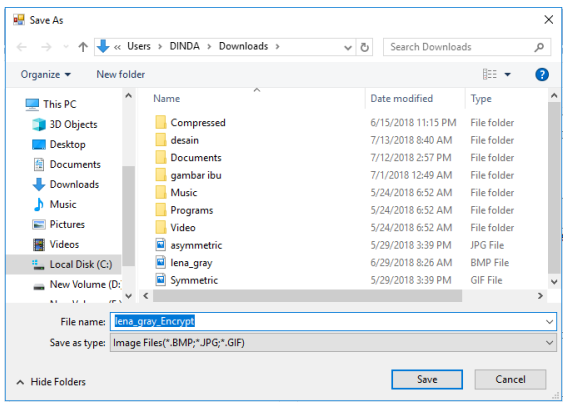
Gambar 6. Proses enkripsi menggunakan IDEA dan input plain image

Gambar di atas menjelaskan cara menginputkan plain text dan kunci untuk memulai.enkripsi dengan menggunakan IDEA.



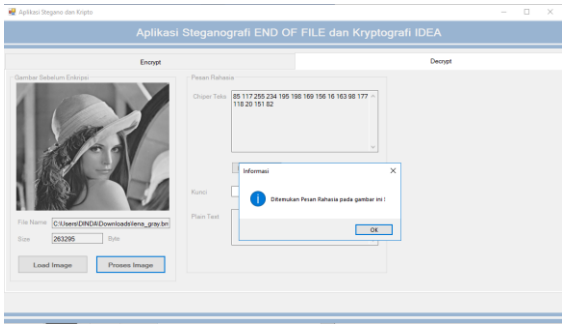
Gambar 7. Proses Stegano dengan EOF

Setelah dilakukan enkripsi maka hasil keluaran dari proses tersebut berupa chiper text berupa karakter dan desimal (85 117 255 234 195 198 169 156 16 163 98 177 118 20 151 82). Input plain image berupa format bmp. Keluaran dari proses stegano menggunakan EOF adalah stego image. Untuk perbandingan size plain image sebelum disisipkan yaitu 263224 byte, sedangkan, size plain image sesudah disisipkan (stego image) adalah 263295 byte.



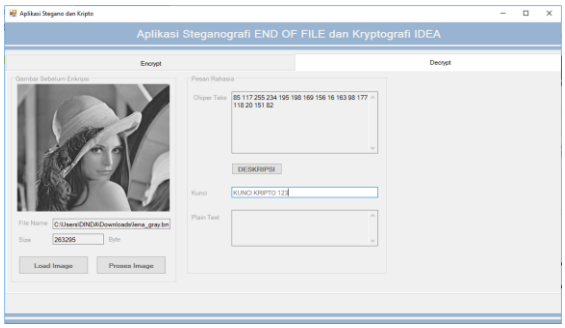
Gambar 8. Penyimpanan Stego Image

Hasil stego image dapat disimpan dengan nama baru yaitu (nama file)_Encrypt.



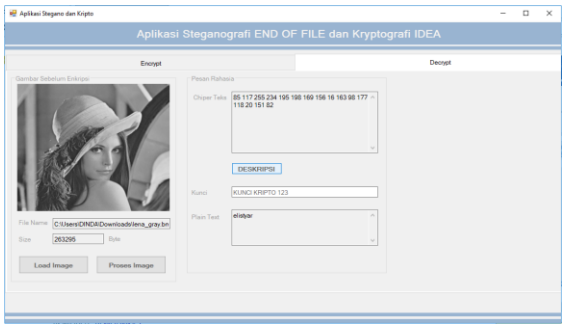
Gambar 9. Proses Ekstraksi dengn EOF

Untuk mengambil kembali chiper text yang ada pada stego image yang tidak diedit untuk dilakukan ekstraksi menggunakan End of File. Setelah dicek maka keluar notifikasi bahwa terdapat pada stego image tersebut.



Gambar 10. Hasil proses Ekstraksi EOF

Hasil yg chiper text yang disisipkan : 85 117 255 234 195 198 169 156 16 163 98 177 118 20 151 82. Kemudian menginputkan kunci untuk memproses chiper text agar mendapatkan plain text dengan dekripsi menggunakan IDEA.



Gambar 11. Proses Dekripsi menggunakan IDEA

Gambar di atas merupakan hasil keluaran dari proses dekripsi menggunakan IDEA.

5 Kesimpulan

Dari hasil perancangan dan pengujian sistem dalam aplikasi Steganografi End of File dan Kriptografi IDEA dapat disimpulkan bahwa :

- Dari hasil percobaan yang telah dilakukan membuktikan bahwa aplikasi dapat

- mengacak dan menyembunyikan file dengan aman dan tidak menimbulkan kecurigaan pada pihak lain. Pada file hasil kriptografi dan steganografi tidak menimbulkan efek yang dapat merusak ataupun mengganggu kinerja file sebelumnya.
- Proses enkripsi IDEA dan steganografi End of File dapat melakukan penyisipan dengan baik karena ukuran stego image yang tidak berbeda secara kasat mata.
- Dengan menggunakan metode End of File maka pesan yang ingin disisi
- Perubahan ukuran file citra yang disisipkan tergantung dari besarnya pesan yang disisipkan.

Daftar Pustaka:

G.M, Marakas, System Analysis Design: an Active Approach. New York: Mc.Graw---Hill, 2006.

Harianto, Antonio, "Studi Perbandingan Enkripsi Steganografi Dengan Menggunakan Metode Least Significant Bit Dan End Of File," Program Studi Teknik Informatika Jurusan Teknik Elektro Fakultas Teknik Universitas Tanjungpura, 2013.

Munir, Rinaldi, Diktat Kuliah Studi Teknik Informatika, Institut Teknologi Bandung. Bandung, 2006.

Munir, Rinaldi, Kriptografi. Bandung: INFORMATIKA, 2007.

Munir, Rinaldi, Pengolahan Citra Digital. Bandung: INFORMATIKA, 2007.

S.Kromodimoeljo, *Teori dan Aplikasi Kriptografi*. SPK IT Consulting, 2009, p. 458.