PERANCANGAN APLIKASI STEGANOGRAFI MENGGUNAKAN ALGORITMA IDEA DAN METODE EOF

Dyah Ayu Irawati(1), Elistya Dinda Rachmawati (2)
Jurusan Teknologi Informasi
Politeknik Negeri Malang
Jalan Soekarno-Hatta No. 9 Malang
e-mail : dyah.ayu@polinema.ac.id(1), elstdinda@gmail.com(2)

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 mengunakan 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. Hasil penyisipan dapat menyembunyikan plain text dengan perbandingan dari gambar sebelum dan sesudah disisipkan tidak mengalami perbedaan yang mencolok.

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

1. PENDAHULUAN

Seiring pesatnya perkembangan teknologi informasi saat ini 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. Apalagi jika informasi tersebut berada dalam suatu jaringan komputer yang terhubung atau terkoneksi dengan jaringan lain yang publik misalnya internet.

Informasi penting yang ingin dilindungi ini harus diolah dalam bentuk lain atau disandikan, guna untuk melindungi kerahasiaan dari suatu data itu sendiri. Contohnya dalam kehidupan sehari-hari yaitu terdapat pada PIN mesin ATM, password komputer, password ID game online. Berdasarkan masalah tersebut, diperlukan sebuah metode yang dapat membantu mengolah informasi menjadi pesan tidak terbaca yaitu kriptografi. Kriptografi akan merahasiakan informasi dan menyandikannya ke dalam informasi acak yang tidak dimengerti kecuali oleh penerima yang tepat. Hal ini dikarenakan kriptografi itu sendiri ialah ilmu dan seni untuk menjaga keamanan pesan atau data. Kriptografi mengubah sebuah data atau informasi menjadi pesan acak yang sesuai dengan kunci pengacaknya. Dimana kunci ini diketahui oleh pengirim dan penerima informasi saja. Tentunya pengaplikasian kriptografi akan mengamankan sebuah informasi. Namun, saat ini kriptanalisis atau kemampuan untuk memecahkan kriptografi turut berkembang pesat dan telah banyak metode kriptografi yang terpecahkan. Salah satu metode kriptografi yang dianggap sebagai algoritma block cipher yang terbaik dan teraman yang tersedia untuk publik sampai saat ini adalah metode Kriptografi IDEA (*International Data Encryption Algorithm*).

2. TINJAUAN PUSTAKA

Dalam bagian ini dijabarkan mengenai pustaka-pustaka yang diacu dalam penelitian ini, antara lain sebagai berikut :

2.1 Penelitian Pendahulu

Pustaka yang pertama yaitu penelitian oleh Wardhani (2011) berjudul "*Implementasi Metode Kriptografi Idea Pada Priority Dealer untuk Layanan Pemesanan dan Laporan Penjualan Hamdphone Berbasis Web*". Penelitian ini membangun sebuah aplikasi untuk pemesanan barang dan laporan hasil penjualan dengan dilengkapi security network yaitu menggunakan kriptografi dengan algoritma IDEA dengan menggunakan PHP. Hasil dari penelitian ini adalah dihasilkan sebuah sistem layanan penjualan dan pemesanan Handphone berbasis WEB dengan menggunakan kriptografi dengan algoritma IDEA sehingga saat pengiriman data atau pesan dari distributor server ke priority dealer terjaga keamanannya.

Pustaka selanjutnya adalah penelitian dari Zulkarnain (2013) berjudul "*Perancangan Dan Implementasi Sistem Keamanan Data Menggunakan Algoritma Kriptografi Simetri IDEA*". Penelitian ini dilakukan perancangan sebuah aplikasi perangkat lunak yang tidak hanya bisa menjaga keamanan data dengan kuat dan handal, tapi juga mudah dimengerti banyak orang menggunakan IDEA dan toolkit *Visual Basic 6.0*. Hasil dari penelitian ini adalah Melindungi data agar tidak dapat dibaca oleh orang-orang yang tidak berhak, menyisipkan atau menghapus data. Algoritma ini menyediakan keamanan yang cukup tinggi dengan tidak didasarkan atas kerahasiaan algoritmanya akan tetapi lebih ditekankan pada keamanan/kerahasian kunci yang digunakan.

"*Penerapan Steganografi Metode End Of File (EOF) Dan Enkripsi Metode Data Encryption Standard (DES) Pada Aplikasi Pengamanan Data Gambar Berbasis Java Programming*". Pada penelitian yang dilakukan oleh Anggraini (2014), dilakukan perancangan aplikasi steganografi menggunakan algoritma DES dan metode EOF dalam mendesain gambar pada Factory Outlet berbasis *Java Programming*. Hasil dari penelitian ini adalah dengan adanya aplikasi ini diharapkan dapat melindungi data rahasia perusahaan agar tidak mudah terbaca oleh orang yang tidak berkepentingan.

2.2 Perancangan Aplikasi

Menurut Nugroho (2004), perancangan aplikasi adalah analisa sistem, persiapan untuk merancang dan implementasi agar dapat menyelesaikan dan mengkonfigurasikan komponen-komponen perangkat lunak ke perangkat keras.

2.3 Kriptografi

Ada beberapa definisi yang telah dikemukakan di dalam berbagai literatur. Kata kriptografi berasal dari bahasa Yunani yaitu *cryptos* yang berarti rahasia dan *graphein* yang berarti tulisan. Jadi kriptografi dapat dijelaskan secara harfiah sebagai tulisan rahasia. 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. 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. Berikut ini merupakan ilustrasi sederhana dari proses kriptografi. 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.

2.4 *International Data Encryption Algorithm* (IDEA)

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. 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
   Pada 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.

Tabel 1. Tabel Kunci Enkripsi

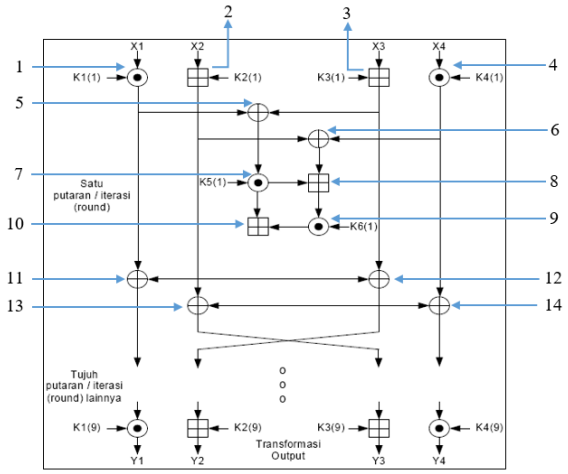
Putaran ke 1	$K_1, K_2, K_3, K_4, K_5, K_6$
Putaran ke 2	$K_7, K_8, K_9, K_{10}, K_{11}, K_{12}$
Putaran ke 3	$K_{13}, K_{14}, K_{15}, K_{16}, K_{17}, K_{18}$
Putaran ke 4	$K_{19}, K_{20}, K_{21}, K_{22}, K_{23}, K_{24}$
Putaran ke 5	$K_{25}, K_{26}, K_{27}, K_{28}, K_{29}, K_{30}$
Putaran ke 6	$K_{31}, K_{31}, K_{33}, K_{34}, K_{35}, K_{36}$
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. Tabel Kunci Dekripsi

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 1. Alur 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 subkunci enkripsi.

2.5 Steganografi

Steganografi berasal dari bahasa Yunani yaitu Steganos yang berarti menyembunyikan dan Graptos yang artinya tulisan sehingga secara keseluruhan artinya adalah tulisan yang disembunyikan. Secara umum 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.

2.6 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 *End of File* 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.

3. METODE PENELITIAN

Metodologi penelitian yang digunakan dalam perancangan aplikasi steganografi ini adalah sebagai berikut :

a. Analisa

Pada proses pengumpulan kebutuhan dilakukan analisis terhadap permasalahan, yaitu mendefinisikan kebutuhan perangkat, sistem, data atau literatur yang terkait dengan metode yang digunakan.

Pengumpulan Kebutuhan

- Kebutuhan Perangkat Sistem

  Perangkat sistem yang akan digunakan untuk penelitian ini adalah laptop dengan spesifikasi :

  Processor : Intel Core i5
  System type : 64-bit Operating System
  OS : Windows 10

Selain itu, perangkat lunak yang dibutuhkan untuk penelitian ini yaitu:

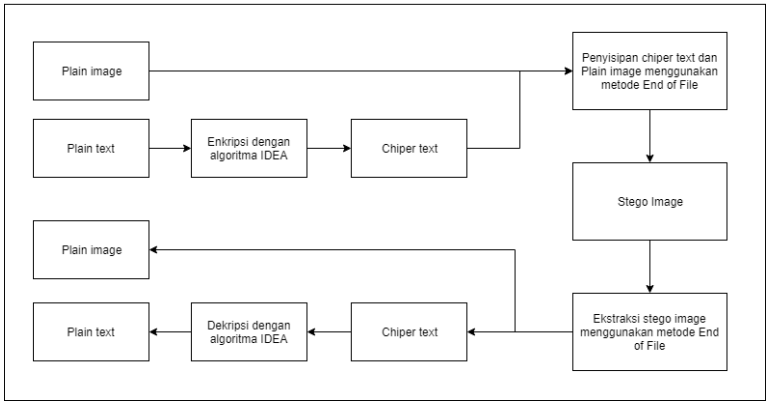
Visual Studio

- Literatur

Kebutuhan literatur yang terkait dengan proses enkripsi dan dekripsi serta proses penyisipan dan ekstraksi pada data berupa teks yang menggunakan algoritma IDEA dan metode End of File melalui jurnal dan referensi yang ada.

b. Perancangan Sistem

Pada Gambar 2 merupakan proses kombinasi kriptografi dan steganografi. Teks asli (plain text) yang dienkripsi terlebih dahulu dengan menggunakan IDEA sebelum disisipkan ke dalam citra/gambar/cover-image. Hasil keluaran dari proses enkripsi berupa pesan terekripsi (chiper text) yang kemudian disisipkan ke dalam citra/gambar/cover-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. Selanjutnya, chiper text didekripsi dengan menggunakan IDEA sehingga didapatkan plain text.

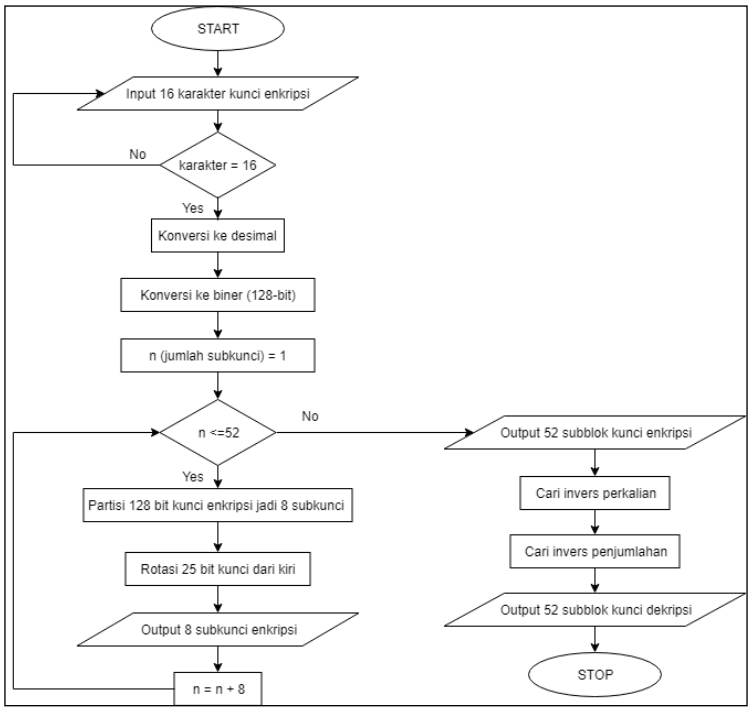


Gambar 2. Proses Kombinasi Metode IDEA dan End of File

Proses penyelesaian menggunakan IDEA memiliki 3 proses :

1. Proses pembuatan kunci

   Proses pembuatan kunci dijelaskan pada diagram alur sebagai berikut :



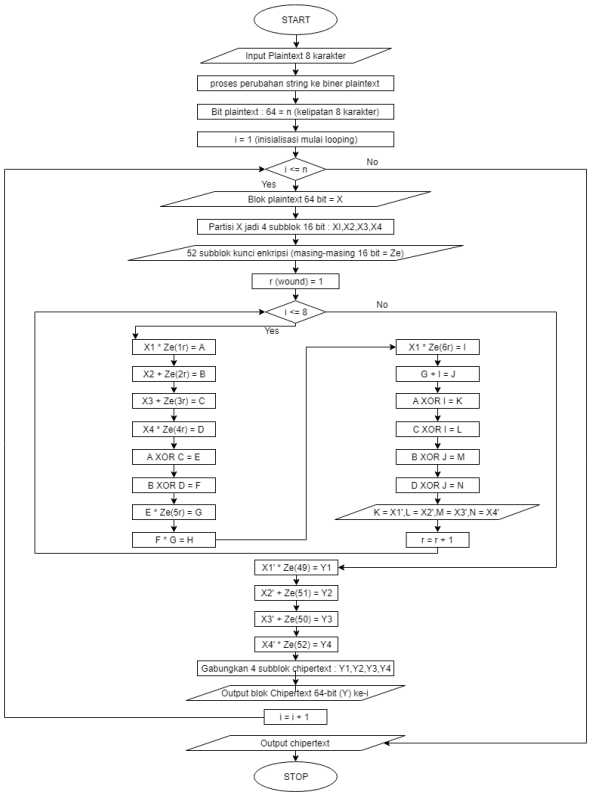
Gambar 3. Proses Pembentukan Kunci IDEA

Pada proses pembentukan kunci menenggunakan metode IDEA, dilakukan langkah awal yaitu memasukkan kunci, jumlah kunci sebanyak 16 karakter, jika jumlah kunci lebih dari 16, maka mengisi kembali

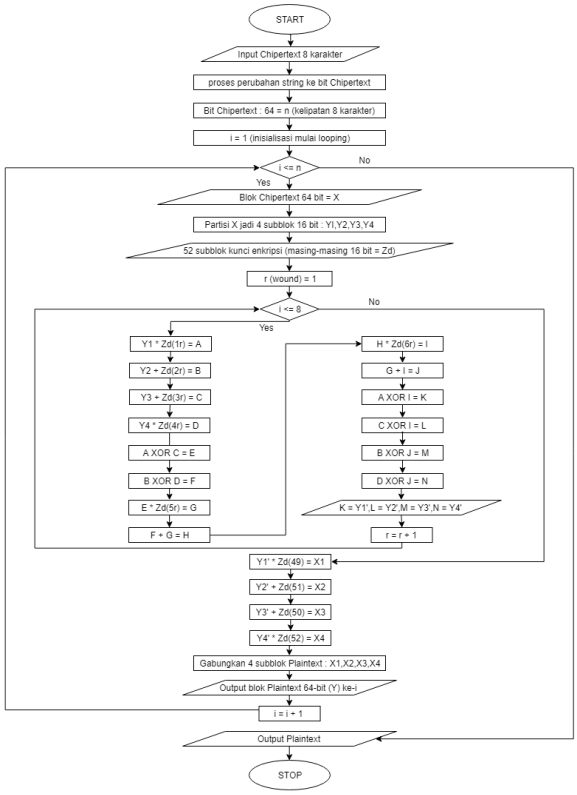
kunci hingga 16, selanjutnya mengkonversi kunci ke decimal, mengkonversi decimal ke biner, memulai sub kunci dengan n sama dengan 1. Kemudian mengecek apakah n kurang lebih sama dengan 52, jika iya dilanjutkan mempartisi kunci menjadi 8 sub kunci. Setelah itu dirotasi sebanyak 25 bit dari kiri sehingga menghasilkan output 8 subkunci enkripsi, kemudian ditambahkan dengan 8 dan dicek kembali hingga mencapai 52 sub kunci. Jika sudah mencapai 52 sub kunci enkripsi, dilanjutkan untuk mencari subkunci dekripsi dengan melakukan invers perkalian, invers pejumlahan sehingga menghasilkan 52 sub kunci dekripsi.

2. Proses enkripsi

Proses enkripsi membutuhkan plain text yang dibagi menjadi 4 bagian dan kunci enkripsi. Plain text 4 sub blok 16 bit digunakan untuk masukan putaran pertama IDEA. Dapat dilihat di gambar pada setiap putaran terdapat operasi aljabar seperti penjumlahan modulo (216), perkalian modulo (216+1), dan XOR. Keluaran dari putaran sebelumnya menjadi masukan di putaran berikutnya. Hal tersebut berulang hingga 8 putaran. Pada putaran terakhir (putaran ke 9) dilakukan transformasi keluaran menjadi Y1,Y2,Y3, dan Y4 yang digabungkan menjadi 64 bit. Semua proses tersebut berada di bawah kendali 52 sub blok kunci 16 yang terbentuk dari blok kunci 128 bit.



Gambar 4. Proses Enkripsi IDEA



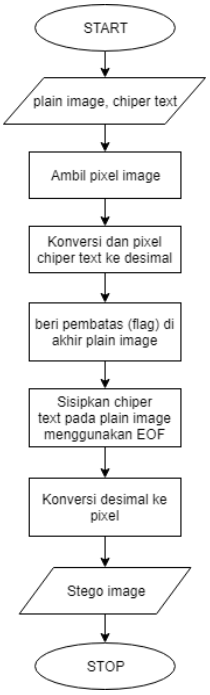
Gambar 5. Proses Dekripsi IDEA

3. Proses dekripsi

Proses dekripsi dijelaskan pada gambar 5. Proses yang dilakukan dalam dekripsi IDEA sama halnya dengan proses enkripsi IDEA. Namun, yang membedakan yaitu masukan untuk memproses menggunakan chipertext hasil proses sebelumnya (enkripsi) dan kunci yang dipakai yaitu kunci dekripsi. Kunci dekripsi merupakan kunci kebalikan dari kunci enkripsi, dengan mengurutkan sub kunci dengan di-inverse-kan.

Pada metode *End of File* terdapat 2 proses penyelesaian yaitu :

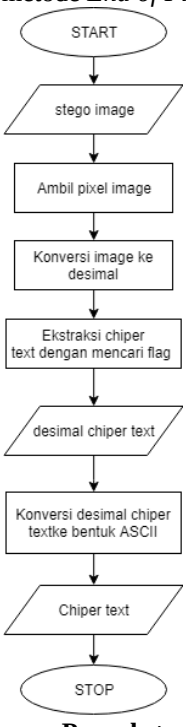
1. Penyisipan pesan ke dalam gambar (*encoder*).
   Diagram alir penyisipan teks dengan metode *End of File* dijelaskan sebagai berikut :



Gambar 6. Proses Penyisipan / *Encode*

Masukan yang diperlukan pada proses penyisipan teks ke dalam gambar yaitu *plain image* dan pesan, mengambil *pixel plain image*, kemudian mengkonverikan ke dalam bentuk decimal. Berikutnya, memberi tanda (flag) untuk melakukan proses penyisipan gambar, konversikan ke pixel kembali, dan hasilnya berupa stego image.

2. Pengekstrakan gambar untuk mengambil pesan (*decoder*).
   Diagram alir pengekstrakan stego image dengan metode *End of File* dijelaskan sebagai berikut :



Gambar 7. Proses Pengekstrakan / *Decode*

Pada proses ekstraksi stego image, dilakukan masukan stego image, kemudian mengambil pixel gambar, mengkonversikan image ke dalam bentuk decimal. Berikutnya, mengekstraksi stego image menggunakan *End of File* dengan cara mencari *flag* sehingga didapatkan desimal dari *chiper text.* Setelah itu, dikonversikan ke dalam bentuk ASCII yang menghasilkan *chiper text*.

4. HASIL DAN PEMBAHASAN

Berdasarkan analisis dan perancangan yang telah dijelaskan pada bab sebelumnya implementasi hal tersebut dijelaskan berikut ini :

Proses Enkripsi dengan IDEA

Perhitungan manual proses enkripsi metode IDEA adalah sebagai berikut :
Plain text : elistyar
Kunci : KUNCI KRIPTO 123
Plain text : elistyar
Jika dikonversikan ke biner menjadi :
01100101011011000110100101110011011101000111100101100001011100 10

TRANSFORMASI OUTPUT

1. L1 = (X11*K49) mod (2^{16}+1) = (1000011011010010 * 1001000011010010) mod (2^{16}+1) = 0110110000000000
2. L2 = (X13+K50) mod (2^{16}) = (1000010001010010 + 0100100000010010) mod (2^{16}) = 1100110001100100
3. L3 = (X12+K51) mod (2^{16}) = (0101111010101110 + 1101010010010010) mod (2^{16}) = 0011001101000000
4. L4 = (X14*K52) mod (2^{16}+1) = (0111001100011010 * 0101010000010101) mod (2^{16}+1) = 1101001101010101

Hasil : 01101100 00000000 11001100 01100100 00110011 01000000 11010011 01010101
Chipertext : l Ìd3@ÓU

Proses Dekripsi dengan IDEA

Perhitungan manual proses enkripsi metode IDEA adalah sebagai berikut :
- Chiper text : l Ìd3@ÓU
- Kunci : KUNCI KRIPTO 123

TRANSFORMASI OUTPUT

1. L1 = (X11*K49) mod (2^{16}+1) = (0011001100000100 * 0111111001100111) mod (2^{16}+1) = 0110010101101100
2. L2 = (X13+K50) mod (2^{16}) = (1011011110110110 + 1011000110111101) mod (2^{16}) = 0110100101110011
3. L3 = (X12+K51) mod (2^{16}) = (1011110110011001 + 1011011011100000) mod (2^{16}) = 0111010001111001
4. L4 = (X14*K52) mod (2^{16}+1) = (0111001100011010 * 0111010101101111) mod (2^{16}+1) = 0110000101110010

Hasil : 01100101011011000110100101110011011101000111100101100001011100 10
Plaintext : elistyar

Encode dengan Metode EOF

Terdapat suatu citra RGB 10x7 yang memiliki nilai setiap pixel seperti pada Tabel 3.

Tabel 3. Tabel pixel sebelum disisipkan pesan dengan metode End of File

R=77 G=82 B=43	R=65 G=65 B=89	R=167 G=168 B=70	R=171 G=147 B=153	R=160 G=130 B=141	R=60 G=30 B=41	R=60 G=70 B=141	R=157 G=172 B=182	R=186 G=190 B=189	R=84 G=96 B=79
R=92 G=55 B=35	R=76 G=76 B=90	R=64 G=57 B=158	R=181 G=70 B=179	R=204 G=203 B=199	R=24 G=93 B=109	R=204 G=95 B=99	R=202 G=14 B=71	R=167 G=182 B=184	R=167 G=182 B=184
R=76 G=90 B=82	R=69 G=73 B=83	R=163 G=63 B=171	R=186 G=190 B=189	R=177 G=184 B=162	R=17 G=101 B=56	R=77 G=84 B=162	R=202 G=14 B=71	R=157 G=172 B=182	R=186 G=190 B=189
R=95 G=102 B=112	R=86 G=98 B=114	R=157 G=172 B=182	R=167 G=182 B=184	R=174 G=186 B=187	R=74 G=86 B=18	R=174 G=186 B=187	R=84 G=96 B=79	R=202 G=14 B=71	R=69 G=73 B=83
R=74 G=84 B=99	R=121 G=14 B=152	R=173 G=87 B=194	R=184 G=96 B=21	R=202 G=14 B=71	R=22 G=46 B=73	R=100 G=214 B=227	R=74 G=84 B=99	R=167 G=182 B=184	R=204 G=95 B=99
R=74 G=84 B=99	R=21 G=134 B=100	R=173 G=96 B=94	R=184 G=196 B=107	R=22 G=46 B=79	R=104 G=77 B=99	R=202 G=89 B=99	R=74 G=84 B=99	R=157 G=172 B=182	R=24 G=93 B=109

R=74 G=94 B=93	R=121 G=134 B=92	R=173 G=90 B=194	R=84 G=96 B=79	R=92 G=100 B=81	R=52 G=76 B=91	R=202 G=214 B=227	R=204 G=95 B=99	R=181 G=70 B=179	R=17 G=101 B=56

Citra RGB tersebut akan disisipkan pesan "l Ìd3@ÓU" yang memiliki nilai desimal dalam kode ASCII yaitu "108 0 204 100 51 64 211 85". Pesan akan ditambahkan sebagai nilai akhir secara horizontal pada pixel citra RGB. Pada awal dan akhir pesan diberi penanda yang memiliki nilai desimal dalam kode ASCII yaitu "128, 101, 72, 101, 83, 140, 128, 150, 76, 111, 13, 54". Untuk menyisipkan pesan (*chiper text*) pada gambar dilakukan penambahan satu baris pixel paling bawah. Selanjutnya, proses penyisipan dilakukan penambahan penanda nilai desimal sebanyak 12, penambahan pesan (*chiper text*) sebanyak 8. Kemudian untuk menutupi kecurigaan ditambahkan pixel atasnya menjadi tabel 4.

Tabel 4. Tabel pixel sesudah disisipkan pesan dengan metode End of File

R=77 G=82 B=43	R=65 G=65 B=89	R=167 G=168 B=70	R=171 G=147 B=153	R=160 G=130 B=141	R=60 G=30 B=41	R=60 G=70 B=141	R=157 G=172 B=182	R=186 G=190 B=189	R=84 G=96 B=79
R=92 G=55 B=35	R=76 G=76 B=90	R=64 G=57 B=158	R=181 G=70 B=179	R=204 G=203 B=199	R=24 G=93 B=109	R=204 G=95 B=99	R=202 G=14 B=71	R=167 G=182 B=184	R=167 G=182 B=184
R=76 G=90 B=82	R=69 G=73 B=83	R=163 G=63 B=171	R=186 G=190 B=189	R=177 G=184 B=162	R=17 G=101 B=56	R=77 G=84 B=162	R=202 G=14 B=71	R=157 G=172 B=182	R=186 G=190 B=189
R=95 G=102 B=112	R=86 G=98 B=114	R=157 G=172 B=182	R=167 G=182 B=184	R=174 G=186 B=187	R=74 G=86 B=18	R=174 G=186 B=187	R=84 G=96 B=79	R=202 G=14 B=71	R=69 G=73 B=83
R=74 G=84 B=99	R=121 G=14 B=152	R=173 G=87 B=194	R=184 G=96 B=21	R=202 G=14 B=71	R=22 G=46 B=73	R=100 G=214 B=227	R=74 G=84 B=99	R=167 G=182 B=184	R=204 G=95 B=99
R=74 G=84 B=99	R=21 G=134 B=100	R=173 G=96 B=94	R=184 G=196 B=107	R=22 G=46 B=79	R=104 G=77 B=99	R=202 G=89 B=99	R=74 G=84 B=99	R=157 G=172 B=182	R=24 G=93 B=109
R=74 G=94 B=93	R=121 G=134 B=92	R=173 G=90 B=194	R=84 G=96 B=79	R=92 G=100 B=81	R=52 G=76 B=91	R=202 G=214 B=227	R=204 G=95 B=99	R=181 G=70 B=179	R=17 G=101 B=56
R=128 G=101 B=72	R=101 G=83 B=140	R=128 G=150 B=76	R=111 G=13 B=54	R=108 G=0 B=204	R=100 G=51 B=64	R=211 G=85 B=227	R=204 G=95 B=99	R=181 G=70 B=179	R=17 G=101 B=56

Pengujian

Dari uji coba yang telah dilakukan terhadap 8 kata plain text dengan panjang karakter bervariasi, dalam melakukan proses enkripsi hasil yang diperoleh adalah perubahan pesan ke dalam bentuk yang tidak dapat dibaca atau diartikan (chiper text) seperti pada tabel 5 sebagai berikut :

Tabel 5. Proses Enkripsi dan penyisipan pesan

No	Plain Image	Size	Dimensi	Plain text	Chiper Text	Stego Image	Size
1	barbara_grey.bmp	263.224*Bytes*	512x512	idea	)è)‹ ¿5ó	barbara_grey_Encrypt.bmp	327.562 *Bytes*
2	man.bmp	1049.654 *Bytes*	1024x1024	TextAsli	`Bß?™!Km	man_Encrypt.bmp	1316.171 *Bytes*
3	rocks.bmp	131.382 *Bytes*	256x171	elistyar	l Ìd3@ÓU	rocks _Encrypt.bmp	143.034 *Bytes*
4	pepper.bmp	786.486 *Bytes*	512x512	informatika	î1"Õw`™ öNp•"ù 9	pepper_Encrypt.bmp	811.720 *Bytes*
5	cat.jpg	116.029 *Bytes*	600x400	POLTEKTI	¢ë«×2Ò+	cat_Encrypt.jpg	220.919 *Bytes*
6	tower.jpg	1174.073 *Bytes*	2560x1440	+#2&?F4c	;£çÊ¬àE	tower_Encrypt.jpg	3562.567 *Bytes*
7	dahlia.jpg	25.558 *Bytes*	280x300	eof	OÎÊ±©ÜžÛ	dahlia_Encrypt.jpg	225.283 *Bytes*
8	bumi.jpg	1228.983 *Bytes*	2560x1440	Percobaan menggunakan IDEA dan end of file.	¸¼:£%$¥Đú Þú®ÌA;÷ UÅ7 Emo kT;["*‹ ¾ VS …äÛ³Ñ<	bumi_Encrypt.jpg	3458.924 *Bytes*

Dari hasil proses enkripsi dan penyisipan yang ditunjukkan tabel 5 diperoleh hasil yaitu tidak ada perbedaan yang mencolok dengan plain image sebelum proses penyisipan. Jika diamati, baik dari jenis file yang

berbeda format dan warna baik RGB maupun Greyscale pada plain image tidak mengganggu jalannya proses enkripsi. Tetapi perbedaan bisa dilihat adalah besar ukuran dan dimensi dari plain image sebelum dan sesudah disisipi mengalami perubahan yaitu pada plain image yang sudah disisipi (stego image) akan bertambah ukuran dan dimensinya. Tabel 6 berisi hasil dekripsi yang dilakukan.

Tabel 6 Proses dekripsi dan ekstraksi pesan

No	Stego Image	Size stego	Size unstego	Chiper Text	Plain text	Banyak karakter
1	barbara_grey_Encrypt.bmp	327.562 *Bytes*	263.224*Bytes*	l Ìd3@ÓU	idea	4
2	man_Encrypt.bmp	1316.171 *Bytes*	1049.654 *Bytes*	`Bß?™!Km	TextAsli	8
3	rocks _Encrypt.bmp	143.034 *Bytes*	131.382 *Bytes*	Çgd_ À D	elistyar	8
4	pepper_Encrypt.bmp	811.720 *Bytes*	786.486 *Bytes*	%9ÿ‰{ ×	informatika	11
5	cat_Encrypt.jpg	220.919 *Bytes*	116.029 *Bytes*	¢ë«×2Ò+	POLTEKTI	8
6	tower_Encrypt.jpg	3562.567 *Bytes*	1174.073 *Bytes*	;£çÊ¬àE	+#2&?F4c	8
7	dahlia_Encrypt.jpg	225.283 *Bytes*	25.558 *Bytes*	OÎÊ±©ÜžÛ	eof	3
8	bumi_Encrypt.jpg	3458.924 *Bytes*	1228.983 *Bytes*	¸¼:£%$¥Đú Þú®ÌA;÷ UÅ7 Emo kT;[“*‹ ¾ VS …äÙ³Ñ<	Percobaan menggunakan IDEA dan end of file.	43

Dari hasil dekripsi dan ekstraksi tersebut didapatkan hasil yaitu panjang karakter sama seperti *plain text*. Begitu juga *size* unstego (*plain image*) dapat kembali seperti semula. Semakin besar ukuran dari *plain image* dan *plain text* maka waktu yang diperlukan untuk proses penyisipan juga membutuhkan waktu lebih banyak. Oleh karena itu, banyaknya karakter dan besar ukuran *plain image* dapat mempengaruhi waktu penyisipan.

Berdasarkan hasil dari data pada tabel 6, diperoleh hasil setelah melakukan proses dekripsi dan ekstraksi menggunakan Aplikasi Steganografi End of File dan Kriptografi IDEA. Hasil yang didapatkan dari 8 kali pengujian menunjukkan bahwa pengujian yang dilakukan berhasil.

5. KESIMPULAN

Berdasarkan pembahasan pada bab-bab sebelumnya maka diperoleh kesimpulan pada penelitian ini adalah :

- Langkah-langkah dalam mengamankan data pada aplikasi ini yaitu mengekripsikan plain text menggunakan IDEA yang mengeluarkan output chiper text, selanjutnya dilakukan penyisipan antara chiper text dan plain image menggunakan metode End of File dan hasilnya berupa stego image yang dapat menutup kecurigaan dari pihak lain. Besar ukuran image sebelum maupun sesudah disisipi mengalami perubahan, namun tidak mengubah warna image itu sendiri yang menambah ukuran file.
- Hasil penyisipan dapat menyembunyikan plain text tanpa diketahui secara kasat mata oleh manusia. Dengan kata lain perbandingan dari gambar sebelum dan sesudah disisipkan tidak mengalami perbedaan yang mencolok.
- Pada metode End of File, besar ukuran image baik itu yang kecil maupun yang besar tidak mengganggu proses penyisipan dan ekstraksi yang dapat mengubah image tersebut.

DAFTAR PUSTAKA

Adi Nugroho. 2004, Konsep Pengembangan Sistem Basis Data, Informatika, Bandung

Agustina, Dewi Try. "Perancangan dan Implementasi Sistem Keamanan Data Menggunakan Algoritma Kriptografi Simetri IDEA", Progtam Studi Ilmu Komputer Departemen Ilmu Komputer Universitas Sumatra Utara, 2008.

Anggraini, Yayuk, and Sakti, Dolly. 2014. "Penerapan Steganografi Metode End Of File (EOF) Dan Enkripsi Metode Data Encryption Standard (DES) Pada Aplikasi Pengamanan Data Gambar Berbasis Java Programming", in Konferensi Nasional Sistem Informasi 2014, STMIK Dipanegara Makassar, 27 Februari – 01 Maret 2014, pp.1743-1753.

Arfiyah. 2013. "Perbandingan Teknik Steganografi Dengan Metode First-Of-File, End-Of-File Dan Kombinasi First-Offile Dan End-Of-File Pada File Bitmap". Program Studi S1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.

Hanan, Abdul. 2013. "Metode Enkripsi dan Dekripsi Data Menggunakan Kriptografi IDEA", Program Studi Teknik Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer, STMIK U'BUDIYAH INDONESIA, Banda Aceh,Januari.

Harianto Antonio. 2013."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.

Munir, Rinaldi. 2006. Kriptografi. Bandung: Informatika.

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

Sumandri. 2017. "Studi Model Algoritma Kriptografi Klasik dan Modern", Universitas Negeri Yogyakarta, Program Studi Pascasarjana Pendidikan Matematika, Seminar Matematika dan Pendidikan Matematika UNY, pp. 265-272.

Paskalis Andrianus Nani , "Penerapan Enkripsi Algoritma Blowfish Pada Proses Steganografi EOF," Universitas Katolik Widya Mandira, Kupang.

Wardani, Kholidya Yuli., Hadi, M. Zen S., dan Yuliana, Mike Aqsas. 2011. "Implementasi Metode Kriptografi Idea Pada Priority Dealer untuk Layanan Pemesanan dan Laporan Penjualan Hamdphone Berbasis Web" Artikel Politeknik Elektronika Negeri Surabaya, Institut Teknologi Sepuluh Nopember.

Zulkarnain, Iskandar. 2013. "Perancangan Dan Implementasi Sistem Keamanan Data Menggunakan Algoritma Kriptografi Simetri Idea" SAINTIKOM Vol. 12, No. 2, pp.79-94.