Implementasi Algoritma *Ezstego* Untuk Menyembunyikan Pesan Terenkripsi Dengan *Playfair Cipher* Pada Citra GIF

William Steven, Viki Afriyandi, Kristien Margi Suryaningrum
Universitas Bunda Mulia
wstevenn21@gmail.com, vikiafriyandi00@gmail.com, kristienmargi@gmail.com

ABSTRAK

Kemajuan teknologi informasi telah memberikan dampak yang sangat luas, sehingga informasi yang dibutuhkan begitu sangat mudah didapatkan. Disamping perkembangan informasi yang begitu pesat yang banyak membantu pekerjaan manusia baik dalam hal bertukar informasi maupun menjaga pesan informasi tersebut. Kemudahan pengaksesan media komunikasi oleh semua orang tentunya memberikan dampak bagi keamanan informasi atau pesan, Informasi menjadi sangat rentan untuk diketahui, diambil dan dimanipulasi oleh pihak-pihak yang tidak bertanggung jawab. Teknik Kriptografi dan Steganografi adalah solusi yang biasa digunakan dalam pengamanan data informasi untuk menjaga kerahasiaan pesan dengan cara menyandikannya kedalam bentuk yang tidak dapat dimengerti sedangkan steganografi merupakan seni untuk menyembunyikan pesan terutama pesan yang sudah dienkripsi ke dalam berbagai media digital khususnya media gambar sehingga orang awam tidak menyadari ada sesuatu pesan di dalam media tersebut. *Playfair Cipher* merupakan salah satu Algoritma Kriptografi dengan teknik enkripsi simetris manual yang memberikan beberapa kelebihan dalam enkripsi data. Algoritma *Ezstego* menyisipkan bit-bit pesan dengan metode Least Significant Bit, Metode ini menukar setiap bit pixel dengan bit pesan yang akan disembunyikan. Tahapan penyembunyian pesan pada citra digital aman dan sulit diketahui secara kasat mata.

Kata kunci: Steganografi, *Playfair Cipher*, Algoritma *EzStego*, Kriptografi, Citra GIF, LSB.

ABSTRACT

Advances in information technology have had a very broad impact, so the information needed is very easy to obtain. Besides the rapid development of information that helps human work both in terms of exchanging information and maintaining the information message. Ease of access to communication media by all people certainly has an impact on the security of information or messages. Information becomes very vulnerable to be known, taken and manipulated by parties who are not responsible. Cryptography and Steganography Techniques are solutions commonly used in securing information data to maintain the confidentiality of messages by encoding them into an incomprehensible form while steganography is the art of hiding messages, especially messages that have been encrypted into various digital media, especially image media so that the layman is not realize there is something in the media message. Playfair Cipher is one of the Cryptographic Algorithms with manual symmetric encryption techniques that provide several advantages in data encryption. Ezstego algorithm inserts message bits with the Least Significant Bit method, this method swaps each pixel bit with the message bits to be hidden. The stages of hiding messages in digital images are safe and difficult to know by naked eye.

Keywords : *Steganography, Playfair Cipher, Ezstego Algorithm, Cryptography, GIF , LSB.*

PENDAHULUAN

Perkembangan di dunia teknologi informasi sangat pesat akhir – akhir ini berpengaruh pada segala aspek kehidupan. Salah satunya adalah pengamanan informasi yang bersifat rahasia. Kerahasiaan dari sebuah data adalah suatu aspek yang sangat penting sekarang ini dikarenakan perkembangan ilmu teknologi yang memungkinkan munculnya teknik – teknik baru yang disalahgunakan oleh pihak tertentu untuk mengancam keamanan dan kerahasiaan dari informasi tersebut.

Pengiriman dan penyimpanan dari sebuah informasi yang ingin disampaikan kepada seseorang melalui internet memerlukan suatu proses dimana dapat menjamin keamanan dan keutuhan dari informasi. Sehingga selama proses pengiriman, informasi itu harus terjaga mulai dari pengirim hingga pesan itu sampai pada penerima.

Steganografi adalah sebuah teknik penyembunyian informasi rahasia ke dalam sebuah wadah media sehingga informasi yang disembunyikan sulit untuk dikenali oleh indera manusia. Penggunaaan steganografi antara lain bertujuan untuk menyamarkan keberadaan informasi rahasia sehingga sulit dideteksi bagi orang-orang awam. Steganografi ini dilakukan pada media digital baik berbentuk citra, audio, maupun video.

Kriptografi adalah sebuah teknik yang memiliki fokus dalam mengamankan data dengan mengubah data ke dalam kode – kode tertentu untuk mempersulit seseorang saat membaca data tersebut. Kriptografi ini membuat sebuah data seakan tidak memiliki sebuah informasi yang benar karena perubahan pada setiap kata diubah menjadi bentuk yang tidak dapat di baca dan tidak memiliki arti. Tentu dalam kriptografi ini banyak jenisnya, Salah satunya adalah dengan metode *Playfair Chiper*. Dalam prosesnya kriptografi memiliki dua proses yaitu proses enkripsi dan juga dekripsi. Enkripsi adalah proses mengubah suatu pesan asli (*plaintext*) menjadi suatu pesan dalam bahasa sandi (*ciphertext*) sedangkan dekripsi adalah proses mengubah pesan dalam bahasa sandi (*ciphertext*) kembali menjadi pesan asli (*plaintext*).

Dengan adanya kedua teknik tersebut data atau pesan akan semakin aman apalagi tingkat kesulitan dari masing-masing teknik membuat integritas dari informasi tersebut akan aman, apabila kedua teknik tersebut dipadukan akan membuat sebuah keamanan pada informasi rahasia yang ingin disampaikan seseorang, sehingga tingkat keamanan pada sebuah informasi yang ingin dikirimkan melalui internet akan semakin meningkat.

METODE

Tahap analisis kebutuhan dilakukan dengan menganalisa kebutuhan user, analisa perangkat lunak dan perangkat keras yang dibutuhkan dalam pengembangan sistem serta kebutuhan lain dalam pembuatan basis data. Analisis kebutuhan perangkat keras pada sistem ini yaitu laptop dengan spesifikasi sebagai berikut Acer Swift 3 , *processor intel core i5*, RAM DDR4 4 GB. Analisis kebutuhan perangkat lunak yang membantu pembuatan sistem ini yaitu Sistem Operasi *Windows* 10, *Sublime Text* 3 , Bahasa Pemrograman HTML dan *Google Chrome*.

Tahap selanjutnya yaitu mendesain sistem. Tahap ini dibuat sebelum tahap pengkodean. Tujuan dari tahap ini adalah memberikan gambaran tentang apa yang akan dikerjakan dan bagaimana tampilannya. Tahap ini memenuhi semua kebutuhan pengguna sesuai dengan hasil yang dianalisa seperti rancangan tampilan pengembangan sistem enkripsi dan dekripsi, dan membantu mendefinisikan arsitektur sistem secara keseluruhan. Dokumentasi yang dihasilkan dari tahap desain sistem ini antara lain perancangan *Data Flow Diagram (DFD)* dan Perancangan *Interface*.

Aktivitas pada tahap ini dilakukan pengkodean sistem. Penulisan kode program merupakan tahap penerjemahan desain sistem yang telah dibuat ke dalam bentuk perintah-perintah yang dimengerti komputer dengan mempergunakan bahasa pemrograman. Tahapan ini merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Sistem ini bahasa pemrograman yang dipakai adalah HTML.

Pengujian dilakukan untuk memastikan bahwa software yang dibuat telah sesuai dengan desainnya dan semua fungsi dapat dipergunakan dengan baik tanpa ada kesalahan.

Tahap ini merupakan tahap terakhir dalam metode *waterfall*. Sistem dapat di implementasikan. Pemeliharaan mencakup koreksi dari berbagai *error* yang tidak ditemukan pada tahap-tahap terdahulu, perbaikan atas

implementasi dan pengembangan unit sistem, serta pemeliharaan program. Pemeliharaan sistem dapat dilakukan oleh seorang pengembang untuk meningkatkan kualitas sistem agar jauh lebik baik.

Steganografi merupakan seni komunikasi rahasia dengan menyembunyikan pesan pada objek yang tampaknya tidak berbahaya. Keberadaan pesan steganografi adalah rahasia. Istilah Yunani ini berasal dari kata Steganos, yang berarti tertutup dan Graphia, yang berarti menulis[1]. Steganografi adalah jenis komunikasi yang tersembunyi, yang secara harfiah berarti "tulisan tertutup." Pesannya terbuka, selalu terlihat, tetapi tidak terdeteksi bahwa adanya pesan rahasia. Deskripsi lain yang popular untuk steganografi adalah Hidden in Plain Sight yang artinya tersembunyi di depan mata.
Beberapa hal yang diperlukan untuk menyembunyikan pesan yaitu[2] . :

1. *Embeding Algorithm* Algoritma ini digunakan untuk menyisipkan suatu pesan yang disembunyikan ke dalam suatu data yang akan dikirim. Proses penyisipan ini diproteksi oleh sebuah *keyword* sehingga hanya orang-orang yang mengetahui *keyword* ini yang dapat membaca pesan yang disembunyikan tersebut.
2. *Detector Function* Fungsi Detektor ini adalah untuk mengembalikan pesan-pesan yang disembunyikan tersebut.
3. *Carrier Document* Merupakan dokumen yang berfungsi sebagai media yang digunakan untuk menyisipkan informasi. Dokumen ini dapat berupa *file-file* seperti file *audio*, video atau citra (gambar).
4. *Key* Merupakan kata kunci yang ikut disisipkan kedalam dokumen berguna dan dipakai sebagai proses verifikasi sewaktu informasi akan ditampilkan atau diuraikan.
5. *Secret Message* Merupakan pesan rahasia yang akan disisipkan kedalam *carrier document*. Pesan inilah yang tidak terlihat dan terbaca orang yang tidak berkepentingan.

Algoritma *Ezstego* menyisipkan *bit-bit* pesan pada *bit Least Significant Bit* (LSB) dari indeks palet. Akibat penyisipan tersebut, indeks palet dapat bertambah satu, tetap atau berkurang satu[3]. Oleh karena indeks palet merupakan *pointer* ke palet warna, maka indeks yang baru menunjuk ke warna berikutnya atau ke warna sebelumnya di palet yang tentu saja secara visual berbeda signifikan. Hal ini tentu menimbulkan degradasi warna yang membuat citra *stego* berbeda jauh dengan citra *cover*.
Untuk meminimalkan degradasi warna, maka langkah pertama di dalam algoritma *Ezstego* adalah mengurutkan warna-warna di dalam palet sedemikian sehingga perbedaan dua warna yang bertetangga adalah minimal. Perbedaan dua warna dapat dihitung dengan rumus jarak *Euclidean*. Misalkan warna 1 dinyatakan sebagai vektor (R1, G1, B1) dan warna 2 dinyatakan sebagai (R2, G2, B2). Jarak Eucledian kedua warna tersebut dihitung dengan rumus 1 :

$$d = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 + B_2)^2} \quad \ldots(1)$$

Jadi, proses pengurutan palet dilakukan dengan menghitung jarak antar warna di dalam palet, lalu mengurutkan palet berdasarkan jarak terkecil sedemikian sehingga akhirnya dua warna bertetangga memiliki jarak *Euclidean* yang kecil. *bit-bit* pesan disisipkan pada bit LSB indeks dari palet yang terurut secara sekuensial [3].
Algoritma *playfair cipher* merupakan bagian dari algoritma kriptografi klasik yang menggunakan teknik subtitusi. Subtitusi adalah penggantian setiap karakter plainteks dengan karakter lain. Berdasarkan jenis kuncinya algoritma *playfair cipher* merupakan algoritma simetris. Kunci yang digunakan untuk enkripsi sama dengan dekripsinya[4].
Playfair Cipher mengenkripsi pasangan huruf (digram atau digraf), bukan huruf tunggal seperti pada cipher klasik atau tradisional lainnya. Tujuannya untuk membuat analisis frekuensi menjadi sulit sebab frekuensi kemunculan huruf di dalam *ciphertext* akan menjadi datar.

Menurut Stallings[5] Sebelum melakukan enkripsi, pesan yang akan dienkripsi (*plaintext*) diatur terlebih dahulu sebagai berikut :

1. Semua spasi dan karakter yang bukan alfabet harus dihilangkan dari *plaintext* (jika ada).
2. Jika ada huruf J pada *plaintext*, maka ganti huruf tersebut dengan huruf I.
3. Pesan yang akan dienkripsi ditulis dalam pasangan huruf (bigram).

4. Jika ada huruf yang sama dalam pasangan huruf, maka sisipkan huruf X atau Z di tengahnya. Huruf yang disisipkan sebaiknya huruf X karena sangat kecil kemungkinan terdapat huruf X yang sama dalam bigram, tidak seperti huruf Z.
5. Jika jumlah huruf pada *plaintext* adalah ganjil maka pilih sebuah huruf tambahan yang dipilih oleh orang yang mengenkripsi dan tambahkan di akhir *plaintext*. Huruf tambahan dapat dipilih misalnya huruf Z atau X.

HASIL

1. PROSES ENKRIPSI ALGORITMA PLAYFAIR CIPHER

Plainteks = INSTRUMENTS
Kunci = MONARCHY

Tabel 1. *Square Key*

M	O	N	A	R
C	H	Y	B	D
E	F	G	I	K
L	P	Q	S	T
U	V	W	X	Z

Kemudian, membuat plainteks menjadi huruf berpasang-pasangan:
Plainteks = **IN ST RU ME NT SX**
Menambahkan huruf X jika plainteks yang dimasukan bernilai ganjil.

Maka Proses Enkripsi *Playfair Cipher* :

1. **IN** berada pada baris dan kolom yang berbeda pada bujursangkar kunci, maka huruf **I** di tukar dengan huruf **G**, dan huruf **N** di tukar dengan huruf **A** maka hasil enkripsi pasangan huruf **IN** = **GA**.

2. **ST** berada pada kolom yang sama pada bujursangkar kunci, maka huruf **S** di tukar dengan huruf **T**, dan huruf **T** di tukar dengan huruf **L** maka hasil enkripsi pasangan huruf **ST** = **TL**.

3. **RU** berada pada sudut bujursangkar kunci, maka huruf **R** di tukar dengan huruf **M**, dan huruf **U** di tukar dengan huruf **Z** maka hasil enkripsi pasangan huruf **RU** = **MZ**.

4. **ME** berada pada baris yang sama pada bujursangkar kunci, maka huruf **M** di tukar dengan huruf **C**, dan huruf **E** di tukar dengan huruf **L** maka hasil enkripsi pasangan huruf **ME** = **CL**.

5. **NT** berada pada baris dan kolom yang berbeda pada bujursangkar kunci, maka huruf **N** di tukar dengan huruf **R**, dan huruf **T** di tukar dengan huruf **Q** maka hasil enkripsi pasangan huruf **NT** = **RQ**.

6. **SX** berada pada baris yang sama pada bujursangkar kunci, maka huruf **S** di tukar dengan huruf **X**, dan huruf **X** di tukar dengan huruf **A** maka hasil enkripsi pasangan huruf **SX** = **XA**.

Hasil Proses Enkripsi = **GA TL MZ CL RQ XA**

2. PROSES STEGANOGRAFI ALGORITMA EZSTEGO

Langkah pertama adalah mengurutkan warna-warna di dalam palet, kemudian palet-palet yang memiliki jarak *Euclidean* terdekat akan diurutkan untuk menyisipkan *bit-bit* pesan dengan menggunakan metode *least significant bit*.
Contoh:

Gambar 1. Contoh Citra GIF

Berdasarkan Gbr. 1. Diambil nilai 5 x 5 *pixel* pertama untuk menyisipkan pesan yang sudah terenkripsi *playfair cipher* sebagai berikut:

Ciphertext : GA TL MZ CL RQ XA
Konversi Bilangan biner
01000111 01000001 01010100 01001100 01001101 01011010 01000011 01001100 01010010 01010001 01011000 01000001.

Tabel 2. Nilai RGB

R= 24 G= 29 B= 23	R= 136 G= 104 B= 68	R= 115 G= 102 B= 80	R= 41 G= 49 B= 41	R= 36 G= 41 B= 34
R= 24 G= 29 B= 23	R= 83 G= 68 B= 48	R= 202 G= 203 B= 186	R= 38 G= 33 B= 22	R= 41 G= 49 B= 41
R= 24 G= 29 B= 23	R= 124 G= 95 B= 58	R= 250 G= 236 B= 229	R= 200 G= 148 B= 112	R= 58 G= 66 B= 55
R= 24 G= 29 B= 23	R= 124 G= 95 B= 58	R= 238 G= 150 B= 141	R= 221 G= 227 B= 215	R= 145 G= 91 B= 47
R= 24 G= 29 B= 23	R= 75 G= 55 B= 38	R= 137 G= 116 B= 86	R= 201 G= 181 B= 138	R= 199 G= 170 B= 155

Langkah-langkah algoritma ezstego dalam penyembunyian pesan adalah sebagai berikut:

1. Urutkan palet berdasarkan nilai *pixel* citra.
2. Urutkan palet dari citra berdasarkan jarak *Euclidean* terdekat antar *pixel.*

$$d = \sqrt{(R_1 - R_2)^2 + (G_1 - G_2)^2 + (B_1 + B_2)^2} \quad \text{... (2)}$$

$$d_{12} = \sqrt{(24-136)^2 + (29-104)^2 + (23-68)^2}$$

$d_{12} = \sqrt{20194}$

$d_{12} = 142{,}11$

$$d_{13} = \sqrt{(128-123)^2 + (156-215)^2 + (78-79)^2}$$

$d_{13} = \sqrt{3505}$

$d_{13} = 59{,}75$

3. Masukan indeks baru pada palet yang terurut yang memiliki jarak *Euclidean* terdekat yang didapat dari langkah nomor 2, sehingga menghasilkan indeks baru.
4. Ganti *Least Significant Bit* dari indeks palet yang terurut dengan bit-bit pesan sehingga menghasilkan sebuah citra *ezstego*

Ciphertext : 01000111 01000001 01010100 01001100 01001101 01011010 01000011 01001100 01010010 01010001 01011000 01000001

Ganti *Least Significant Bit* dalam palet warna dengan nilai *bit* pesan.

Palet 1

$G_0 = 28$

Biner : 11100. Nilai biner yang paling terakhir diganti dengan bit pesan yang pertama yaitu 1 sehingga menjadi 11101 = 29. Kemudian menyisipkan bit pesan yang kedua di palet 1 G_0.

Tabel 3.Penyisipan bit pesan *ezstego*

Palet 1	Bit Pesan	Hasil
R_0 = 24 = 11000 G_0 = 29 = 11101 B_0 = 23 = 10111	0 1 0	11000 = 24 11110 = 30 10111 = 23
Palet 2	Bit Pesan	Hasil
R_0=136=10001000 G_0= 104 = 1101000 B_0 = 68 = 1000100	0 0 1	10001000=136 1101000 = 104 1000101 = 69
Palet 3	Bit Pesan	Hasil
R_0= 115 = 1110011 G_0= 102 = 1100110 B_0 = 80 = 1010000	1 1 0	1110100 = 116 1100111 = 103 1010000 = 80

Begitu seterusnya hingga semua palet terisi oleh pesan dan menghasilkan nilai RGB citra sebagai berikut.

Tabel 4. Perubahan Nilai RGB

R= 24 G= 30 B= 23	R= 136 G= 104 B= 69	R= 116 G= 103 B= 80	R= 41 G= 49 B= 41	R= 36 G= 41 B= 34
R= 24 G= 29 B= 23	R= 83 G= 68 B= 48	R= 202 G= 203 B= 186	R= 38 G= 33 B= 22	R= 41 G= 49 B= 41
R= 24 G= 29 B= 23	R= 124 G= 95 B= 58	R= 250 G= 236 B= 229	R= 200 G= 148 B= 112	R= 58 G= 66 B= 55
R= 24 G= 29 B= 23	R= 124 G= 95 B= 58	R= 238 G= 150 B= 141	R= 221 G= 227 B= 215	R= 145 G= 91 B= 47
R= 24 G= 29 B= 23	R= 75 G= 55 B= 38	R= 137 G= 116 B= 86	R= 201 G= 181 B= 138	R= 199 G= 170 B= 155

3. PROSES DEKRIPSI ALGORITMA PLAYFAIR CIPHER

Ciphertext = **GA TL MZ CL RQ XA**
Kunci = MONARCHY

Tabel 5. *Square Key*

M	**O**	**N**	**A**	**R**
C	**H**	**Y**	**B**	**D**
E	**F**	**G**	**I**	**K**
L	**P**	**Q**	**S**	**T**
U	**V**	**W**	**X**	**Z**

Kemudian, membuat plainteks menjadi huruf berpasang-pasangan:
Ciphertext = **GA TL MZ CL RQ XA**
Maka Proses Dekripsi *Playfair Cipher* :

1. **GA** berada pada baris dan kolom yang berbeda pada bujursangkar kunci, maka huruf **G** di tukar dengan huruf **I**, dan huruf **A** di tukar dengan huruf **N** maka hasil enkripsi pasangan huruf **GA** = **IN**.

2. **TL** berada pada kolom yang sama pada bujursangkar kunci, maka huruf **T** di tukar dengan huruf **S**, dan huruf **L** di tukar dengan huruf **T** maka hasil enkripsi pasangan huruf **TL** = **ST**.

3. **MZ** berada pada sudut bujursangkar kunci, maka huruf **M** di tukar dengan huruf **R**, dan huruf **Z** di tukar dengan huruf **U** maka hasil enkripsi pasangan huruf **MZ** = **RU**.

4. **CL** berada pada baris yang sama pada bujursangkar kunci, maka huruf **C** di tukar dengan huruf **M**, dan huruf **L** di tukar dengan huruf **E** maka hasil enkripsi pasangan huruf **CL** = **ME**.

5. **RQ** berada pada baris dan kolom yang berbeda pada bujursangkar kunci, maka huruf **R** di tukar dengan huruf **N**, dan huruf **Q** di tukar dengan huruf **T** maka hasil enkripsi pasangan huruf **RQ** = **NT**.

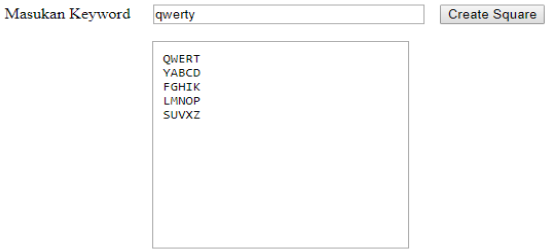
6. **XA** berada pada baris yang sama pada bujursangkar kunci, maka huruf **X** di tukar dengan huruf **S**, dan huruf **A** di tukar dengan huruf **X** maka hasil enkripsi pasangan huruf **XA** = **SX**.

Hasil Proses Dekripsi = **INSTRUMENTSX**

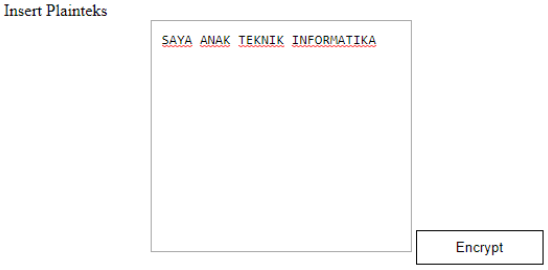
PERANCANGAN DAN IMPLEMENTASI

1. ANTARMUKA

Perancangan struktur antarmuka merupakan bagian dari sistem pakar yang digunakan sebagai alat komunikasi antara pengguna (*User*) dengan sistem. Berikut tampilan antarmuka:

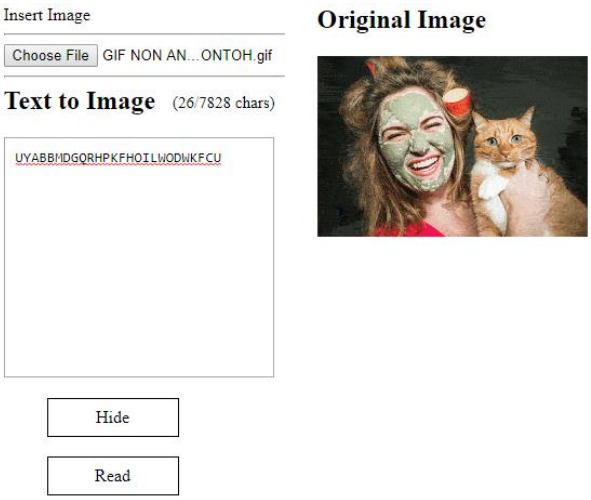


Gambar 2. Tampilan Keyword dan Square



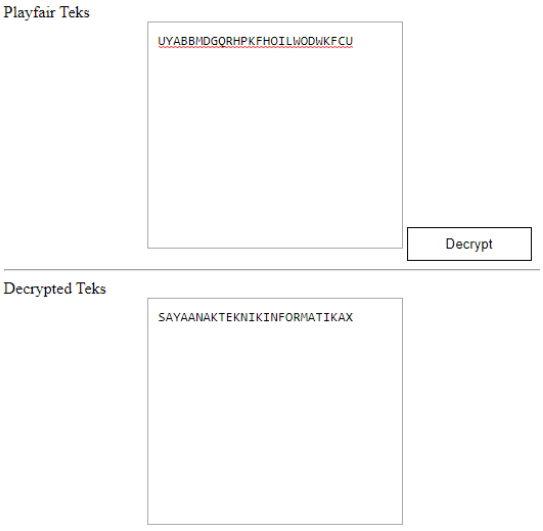
Gambar 3. Tampilan Textfield pada Plainteks

Gambar 4. Tampilan hasil *Ciphertext*



Gambar 5. Tampilan Proses Penyisipan

Gambar 6. Tampilan Hasil Steganografi Image

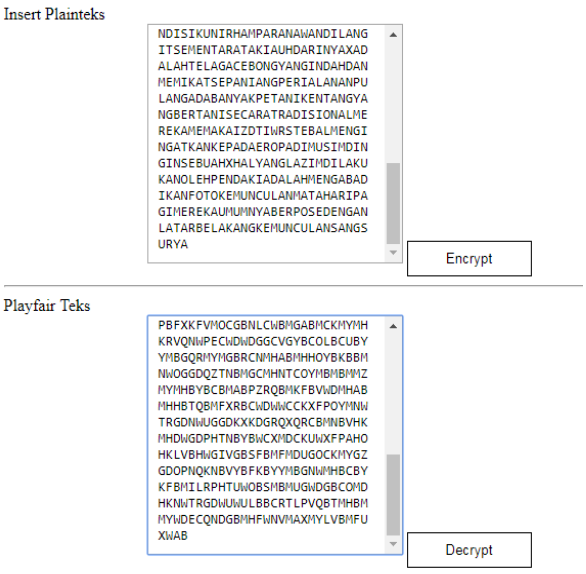


Gambar 7. Tampilan Dekripsi

PENGUJIAN PROSES ENKRIPSI

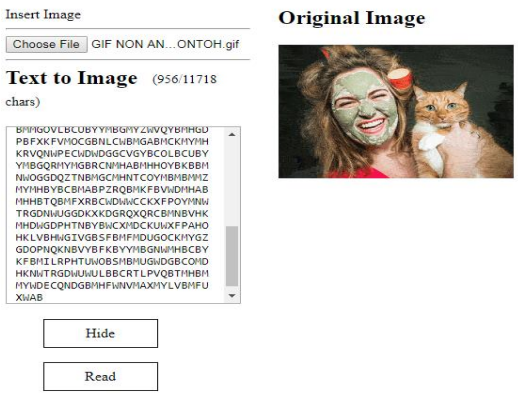
Gambar 8. Proses Pembuatan Square Key

Pada Gbr. 8. mejelaskan dimana proses membuat kunci *playfair* yang digunakan sebagai kunci untuk mengenkrip dan dekrip dari sebuah plaintext yang sudah dibuat. Kunci ini terdapat 25 buah huruf kunci Kriptografi yang terdapat dalam *Playfair Chiper*, kunci tersebut disusun di dalam bujursangkar 5x5 dengan menghilangkan huruf J dari abjad.



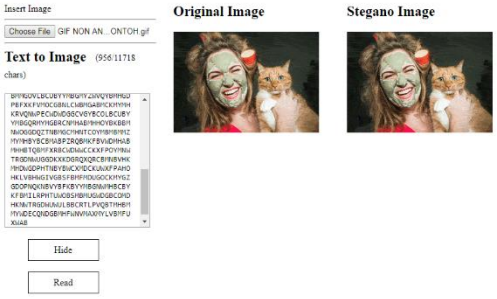
Gambar 9. Pengujian Proses Enkripsi

Pada Gbr. 9. menjelaskan dimana setelah membuat kunci *square user* memasukkan *plaintext* dan setelah di *input user* menekan *button encrypt* untuk mengubah *text* tersebut ke bentuk *chippertext*. Pesan yang dienkripsi diatur terlebih dahulu dengan mengganti huruf j (jika ada) dengan huruf i, kemudian pesan ditulis dalam pasangan huruf (bigram/huruf berpasangan dua-dua).



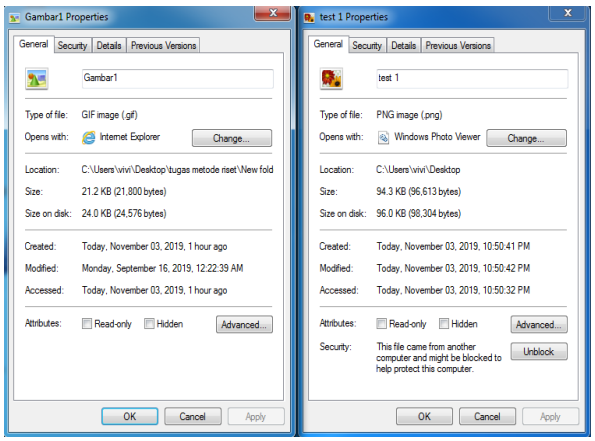
Gambar 10. Pengujian Penyisipan *Plaintext*

Pada *textfield* terdapat cipherteks yang akan disisipkan kedalam gambar. Gbr. 10. bernama Gambar1.gif setelah tombol *Hide* ditekan akan menghasilkan seperti berikut :



Gambar 11. Hasil Gambar Steganografi

Pada Gbr. 11. terlihat menghasilkan sebuah citra baru akan tetapi sulit untuk melihat perbedaan terhadap kedua gambar tersebut. Akan terlihat berbeda jika melihat perbandingan antara *Original Image* dan *Stegano Image.*

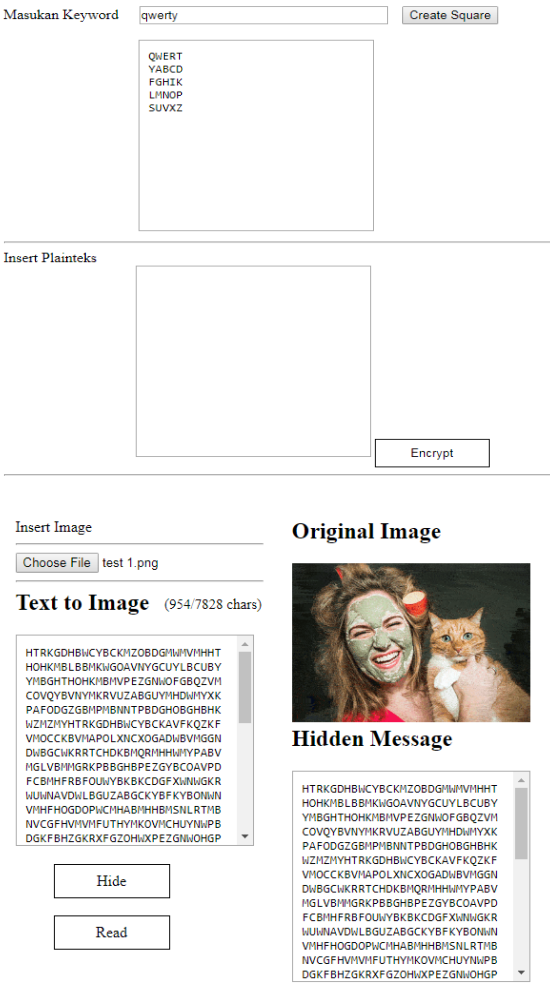


Gambar 12. Pengujian Penyisipan *Chipertext*

Pada Gbr. 12. menjelaskan keberhasilan proses penyisipan *chippertext* ke dalam gambar.

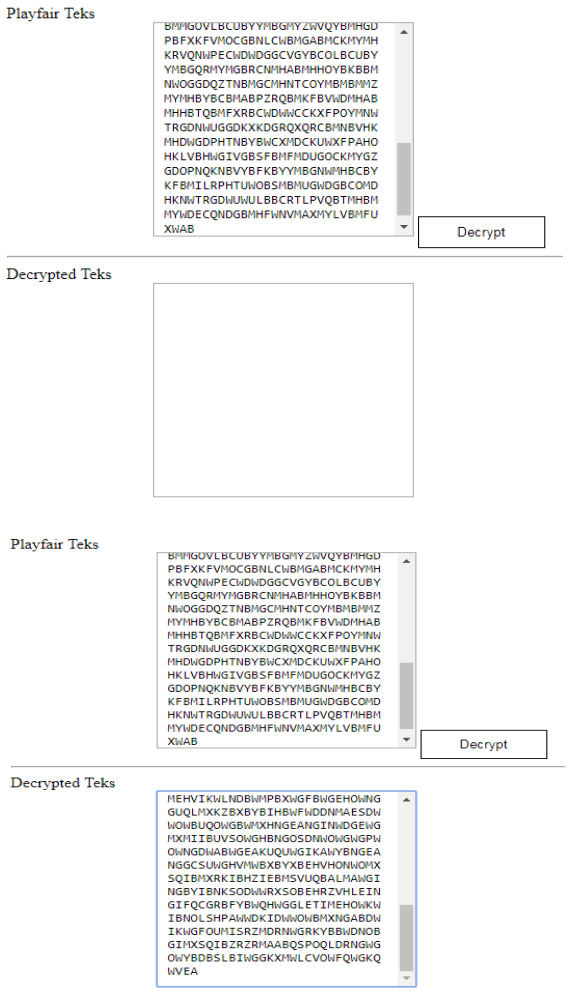
Disini dapat dilihat dari perbedaan kedua gambar dari *original image* dan *stegano image*, terdapat garis-garis putih dalam *stegano image* itu dikarenakan adanya penyisipan *text* ke dalam *pixel* sehingga gambar mengalami perubahan. Dapat dilihat juga dari perbedaan *size* dari image yang asli dengan yang sudah disisipkan pesan, *size image* yang sudah di sisipkan pesan bertambah dari yang 21.2 KB menjadi 94.3 KB.

PENGUJIAN PROSES DEKRIPSI



Gambar 13. Proses Pemisahan *Plaintext*

Pada Gbr. 13. menjelaskan ketikan *user* sudah membuat kunci *square* yang sama, *user* memilih *image* yang tadi sudah disisipkan teks *chippertext*. Selanjutnya *user* menekan *button read* sehingga *chippetext* muncul pada *hidden message*.



Gambar 14. Proses Dekripsi

Pada Gbr. 14. menjelaskan setelah muncul pesan *chippertext* tersebut, *copy* seluruh teks tersebut ke kotak *playfair text* tersebut, kemudian setelah menekan *button decrypt chippertext* akan berubah menjadi *plaintext* awal sehingga dapat di baca kembali.

SIMPULAN

Dari penelitian dan pengujian yang dilakukan maka dapat disimpulkan sebagai berikut:

1. Program ini mampu menyembunyikan pesan ke dalam sebuah citra berbentuk GIF *non-animated* dengan menggunakan algoritma *EzStego*.
2. Program ini hanya bisa menggunakan plainteks tanpa spesial karakter untuk dienkripsi.
3. Hasil dari enkripsi dan dekripsi akan menghilangkan spasi yang ada walaupun ketika memasukan plainteks menggunakan spasi akan tetapi sesuai dengan *rules playfair cipher* bahwa segala bentuk spesial karakter akan dihilangkan.
4. Program ini akan menghasilkan gambar dalam bentuk PNG walaupun gambar original nya berbentuk GIF akan tetapi karena PNG lebih fleksibel dan efisien karena mudah diimprovisasi oleh program

DAFTAR PUSTAKA

[1]. Cox, I., Miller, M., Bloom, J., & Fridrich, J. &. (2008). *Digital Watermarking and Steganography 2nd Ed.* Morgan Kaufmann., MA.

[2]. Sembiring, S., 2013. Perancangan Aplikasi Steganografi Untuk Menyisipkan Pesan Teks Pada Gambar Dengan Metode End Of File. Jurnal Pelita Informatika Budi Darma, Volume : IV , Nomor : 2.

[3]. Defiari, S. Abdul Sani, H. Rivalri Kristianto. (2018). Implementasi Algoritma EzStego Untuk Menyembunyikan Pesan Terenkripsi Data Encryption Standard (DES) Pada Citra GIF. Jurnal Pelita Informatika, Vol 17, No. 4, 422-429. ISSN: 2301-9425.

[4]. Kurniawan, Y., 2004. Kriptografi Keamanan Internet Dan Jaringan Komputer. Bandung: Informatika Bandung.

[5]. Stallings, W., 2010. Cryptography and Network Security: Principles and Practice., 5th edition, Prentice Hall.