IMPLEMENTASI STEGANOGRAFI LSB DENGAN ENKRIPSI VIGENERE CIPHER PADA FILE MULTIMEDIA

MAKALAH

Anik Nur Novitasari Eka .S

NIM 1710652004

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS TEKNIK

UNIVERSITAS MUHAMMADIYAH JEMBER

2017

I. PENDAHULUAN

1.1 Latar Belakang

Steganografi adalah seni dan ilmu untuk menyembunyikan pesan dalam sebuah pesan. Seni dan ilmu ini telah diterapkan sejak dahulu oleh orang Yunani kuno yang menyembunyikan pesan dengan cara membuat tato di kepala pembawa berita yang dibotaki dan menunggu sampai rambutnya tumbuh. Teknik steganografi lainnya adalah dengan menggunakan "invisible ink" (tinta yang tidak tampak). Tulisan yang ditulis dengan menggunakan invisible ink ini hanya dapat dibaca jika kertas tersebut diletakkan di atas lampu atau diarahkan ke matahari. Ketika perang dunia pertama, orang Jerman menyembunyikan pesan dalam bentuk "microdot", yaitu titik-titik yang kecil. Agen dapat membuat foto kemudian mengecilkannya sampai sekecil titik di tulisan dalam buku. Buku ini kemudian bisa dibawa-bawa tanpa ada yang curiga bahwa tanda titik di dalam tulisan di buku itu berisi pesan ataupun gambar.

Kerahasiaan pesan yang ingin disampaikan merupakan faktor utama sehingga digunakan metode steganografi. Dengan metode steganografi, pesan yang ingin disampaikan disembunyikan dalam suatu media umum sehingga diharapkan tidak akan menimbulkan kecurigaan dari pihak lain yang tidak diinginkan untuk mengetahui pesan rahasia tersebut. Oleh sebab itu metode steganografi terus digunakan dan dikembangkan sampai saat ini.

Saat ini internet sudah berkembang menjadi salah satu media yang paling populer di dunia. Karena fasilitas dan kemudahan yang dimiliki oleh internet maka internet untuk saat ini sudah menjadi barang yang tidak asing lagi. Sayangnya dengan berkembangnya internet dan aplikasi menggunakan internet semakin berkembang pula kejahatan sistem informasi. Dengan berbagai teknik banyak yang mencoba untuk mengakses informasi yang bukan haknya. Maka dari itu sejalan dengan berkembangnya media internet ini harus juga dibarengi dengan perkembangan pengamanan sistem informasi.

Atas dasar uraian diatas, maka pada penulisan tugas akhiri ini akan membahas mengenai bagaimana mengamankan suatu pesan dengan menyisipkan (menyembunyikan) kedalam pesan lainnya yaitu file multimedia dengan menggunakan algoritma LSB (Least Significant Bit) pada suatu aplikasi steganografi.

1.2 Tujuan

Tujuan dari pembuatan Makalah ini adalah untuk membuat sebuah program steganografi yang mampu menyisipkan data atau informasi berupa teks dengan menggunakan teknik LSB (Least Significant Bit) dan dengan Enkripsi Vigenere Cipher pada media dengan format JPEG dengan menggunakan bahasa pemprograman Microsoft Visual Basic 6.

1.3 Batasan Masalah

Agar Makalah ini dapat mencapai tujuan yang ingin dicapai, maka batasan masalah disusun sebagai berikut.

1. Makalah ini menitik beratkan pada pembuatan sebuah aplikasi pengamanan teks yang disisipkan dalam gambar diam dalam format berekstensi .*jpeg yang merupakan format dari Joint Photographic Experts Group (JPEG).
2. Cara penyisipan informasi ke dalam sebuah gambar menggunakan teknik Least Significant Bit (LSB).
3. Enkripsi dengan menggunakan Enkripsi Vinegere Cipher.
4. Tipe informasi yang disembunyikan adalah berkas teks.

II. LANDASAN TEORI

2.1 Steganografi

Steganografi berasal dari bahasa Yunani yaitu Steganós yang berarti menyembunyikan dan Graptos yang artinya tulisan, sehingga secara keseluruhan artinya adalah "tulisan yang disebunyikan". Secara umum steganografi merupakan ilmu yang mempelajari, meneliti, dan mengembangkan seni menyembunyikan sesuatu informasi.

Secara teori, semua file umum yang ada di dalam komputer dapat digunakan sebagai media, seperti file gambar berformat JPEG, GIF, BMP, atau di dalam musik MP3, atau bahkan di dalam sebuah film dengan format WAV atau AVI. Semua dapat dijadikan tempat bersembunyi, asalkan file tersebut memiliki bit-bit data redundan yang dapat dimodifikasi. Setelah dimodifikasi file media tersebut tidak akan banyak terganggu fungsinya dan kualitasnya tidak akan jauh berbeda dengan aslinya.

2.2 Pengertian Steganografi

Steganografi merupakan suatu ilmu atau seni dalam menyembunyikan informasi dengan memasukkan informasi tersebut ke dalam pesan lain. Dengan demikian keberadaan informasi tersebut tidak diketahui oleh orang lain. Tujuan dari steganografi adalah menyembunyikan keberadaan pesan dan dapat dianggap sebagai pelengkap dari kriptografi yang bertujuan untuk menyembunyikan isi pesan. Oleh karena itu, berbeda dengan kriptografi, dalam steganografi pesan disembunyikan sedemikian rupa sehingga pihak lain tidak dapat mengetahui adanya pesan rahasia. Pesan rahasia tidak diubah menjadi karakter aneh seperti halnya kriptografi. Pesan tersebut hanya disembunyikan ke dalam suatu media berupa gambar, teks, musik, atau media digital lainnya dan terlihat seperti pesan biasa.

Teknik Steganografi yang digunakan dalam dunia moderrn sekarang ini sudah sangat beragam. Beragam mulai dari algoritma yang digunakannya sampai pada media yang digunakannya.

Beberapa contoh media penyisipan pesan rahasia yang digunakan dalam teknik Steganography antara lain adalah

1. Teks. Dalam algoritma Steganografi yang menggunakan teks sebagai media penyisipannya biasanya digunakan teknik NLP sehingga teks yang telah disisipi pesan rahasia tidak akan mencurigakan untuk orang yang melihatnya.

2. Audio. Format ini pun sering dipilih karena biasanya berkas dengan format ini berukuran relatif besar. Sehingga dapat menampung pesan rahasia dalam jumlah yang besar pula.

3. Citra. Format ini juga sering digunakan, karena format ini merupakan salah satu format file yang sering dipertukarkan dalam dunia internet. Alasan lainnya adalah banyaknya tersedia algoritma Steganografi untuk media penampung yang berupa citra.

4. Video. Format ini memang merupakan format dengan ukuran file yang relatif sangat besar namun jarang digunakan karena ukurannya yang terlalu besar sehingga mengurangi kepraktisannya dan juga kurangnya algoritma yang mendukung format ini.

2.3 Algoritma Kriptografi

Untuk melakukan kriptografi digunakan algoritma kriptografi. Algoritma kriptografi terdiri dari dua bagian, yaitu fungsi enkripsi dan dekripsi. Enkripsi adalah proses untuk mengubah plaintext menjadi ciphertext, sedangkan dekripsi adalah kebalikannya yaitu mengubah ciphertext menjadi plaintext. Terdapat dua jenis algoritma kriptografi berdasar jenis kuncinya[1], yaitu :

1. Algoritma Simetri, adalah algoritma yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Algoritma standar yang menggunakan prinsip kunci simetri antara lain OTP, DES, RC2, RC4, RC5, RC6, IDEA, Twofish, Blowfish, dan lain lain.

2. Algoritma Asimetri, adalah algoritma yang kunci untuk enkripsi dan dekripsinya jauh berbeda. Algoritma standar yang termasuk algoritma asimetri adalah ECC, LUC, RSA, EI, Gamal dan DH.

Salah satu teknik enkripsi menggunakan kunci simetri adalah teknik subtitusi, yaitu mengganti setiap karakter plaintext dengan karakter lain. Terdapat empat cara dalam menggunakan teknik subtitusi[1], yaitu :

1. Monoalphabet, dimana setiap karakter ciphertext mengganti satu macam karakter plaintext tertentu.
2. Polialphabet, dimana setiap karakter ciphertext mengganti lebih dari satu macam karakter plaintext.
3. Monograf/unilateral, dimana satu enkripsi dilakukan terhadap satu karakter plaintext.
4. Poligraf/multilateral, dimana satu enkripsi dilakukan terhadap lebih dari satu karakter plaintext.

Penilaian sebuah algoritma steganography yang baik dapat di nilai dari beberapa faktor yaitu :

1. Imperectibility. Keberadaan pesan rahasia dalam media penampung tidak dapat dideteksi oleh inderawi. Misalnya, jika covertext berupa file, maka penyisipan pesan membuat file stegotext sukar dibedakan oleh mata dengan covertext-nya. Jika covertext berupa audio (misalnya berkas file mp3, wav, midi dan sebagainya), maka indera telinga tidak dapat mendeteksi perubahan pada file stegotext-nya.
2. Fidelity. Mutu media penampung tidak berubah banyak akibat penyisipan. Perubahan itu tidak dapat dipersepsi oleh inderawi. Misalnya, jika covertext berupa file, maka penyisipan pesan dapat

membuat file stegotext sukar dibedakan oleh mata dengan file covertext-nya. Jika covertext berupa audio (misalnya berkas file mp3, wav, midi dan sebagainya), maka audio stegotext tidak rusak dan indera telinga tidak dapat mendeteksi perubahan pada file stegotext-nya.

3. Recovery. Pesan yang disembunyikan harus dapat diungkapkan kembali (reveal). Karena tujuan steganography adalah data hiding, maka sewaktuwaktu pesan rahasia di dalam stegotext harus dapt diambil kembali untuk digunakan lebih lanjut.

2.4 Teknik Penyembunyian Data

Teknik penyembunyian data ke dalam file digital dapat dilakukan dalam dua macam domain:

1. Domain Spasial/waktu (spatial/time domain). Teknik ini memodifikasi langsung nilai byte dari covertext (nilai byte dapat merepresentasikan intesitas/warna pixel atau amplitude). Metode yang tergolong ke dalam teknik ranah spasial adalah metode LSB.

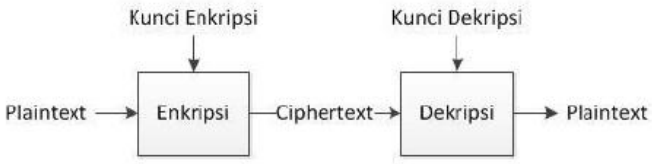
2. Domain Transform (frequency transform domain). Teknik ini memodofikasi langsung hasil transformasi frekuensi sinyal. Metode yang tergolong ke dalam teknik ini ranah frekuensi adalah spread spectrum.

Pada tugas akhir ini, akan digunakan metode LSB (Least Significant Bit) yang merupakan teknik penyembunyian data yang bekerja pada domain spasial. Tiga aspek yang berbeda yang mempengaruhi sifat sistem penyembunyian atau penyisipan pesan rahasia pada gambar adalah : kapasitas, keamanan, dan ketahanan. Kapasitas merujuk pada jumlah informasi yang dapat disembunyikan dalam medium cover. Keamanan adalah ketidakmampuan pengamat untuk mendeteksi pesan yang tersembunyi, dan ketahanan yaitu jumlah modifikasi stego medium yang dapat bertahan sebelum musuh dapat merusak pesan rahasia yang tersembunyi tersebut.

Steganografi modern hanya dapat dideteksi jika pesan rahasia diketahui kunci rahasianya. Hal ini mirip dengan prinsip Kerckhoff dalam kriptografi, yaitu memegang suatu keamanan sistem kriptografi harus mempercayakan semata-mata pada materi kuncinya. Dalam hal steganografi untuk tetap tidak terdeteksi, maka medium cover yang tidak dimodifikasi harus dijaga tetap rahasia, karena jika diperlihatkan, maka perbandingan antara medium cover dan medium stego akan mudah terungkap perbedaannya. Christian Cachin mengusulkan suatu model informasi teoritis untuk steganografi dengan mempertimbangkan masalah keamanan dari sistem steganografi terhadap pengintai yang pasif. Dalam model ini, dianggap bahwa musuh telah mengetahui sistem encoding tetapi tidak mengetahui kunci rahasianya.

Pada dasarnya komunikasi steganografi yaitu, pengirim dan penerima setuju pada suatu sistem steganografi dan membagi pakai (sharing) kunci rahasia untuk menentukan bagaimana suatu pesan dikodekan dalam file digital. Untuk mengirim suatu pesan rahasia yang tersembunyi.

Proses utama dalam kriptografi ada dua, yaitu enkripsi dan dekripsi. Enkripsi adalah proses penyembunyian pesan dengan menggunakan key tertentu. Sedangkan dekripsi adalah proses pembacaan atau ekstrasi pesan dari ciphertext. Berikut ini gambaran umum dari proses tersebut.



Gambar 2.1 Enkripsi-Dekripsi

Berikut ini penjelesana mengenai istilah dan component utama yang sering dipakai dalam kriptografi:

1. Plaintext . Plaintext adalah pesan yang akan kita kirim atau simpan dalam bentuk aslinya. Plaintext dapat dibaca secara langsung dan bermakna.
2. Ciphertext . Ciphertext adalah pesan yang sudah kita enkripsi. Ciphertext tidak dapat dibaca secara langsubg dan tidak bermakna.

3. Enkripsi. Enkripsi adalah proses penyembunyian pesan. Proses enkripsi merubah pesan plaintext menjadi ciphertext yang tidak bermakna. Pada algoritma saat ini, untuk melakukan enkripsi diperlukan suatu kunci.

4. Dekripsi. Dekripsi adalah proses mengekstraksi pesan yang ada dalam ciphertext. Proses dekripsi akan menghasilkan plaintext yang sama seperti sebelum dienkripsi. Dalam dekripsi diperlukan juga kunci.

5. Key / kunci. Key adalah suatu parameter yang digunakan untuk melakukan enkripsi maupun dekripsi. Kunci yang digunkan dapat berbentuk apapun seperti abjad, bilangan, atau bahkan dalam kriptografi modern dapat berupa bit.

Lewat notasi, proses tersebut dapat ditulis sebagai berikut:

Enkripsi

Ek(P) = C

E = fungsi enkripsi C = ciphertext

P = plaintext K = key

Dekripsi

Dk(C) = P

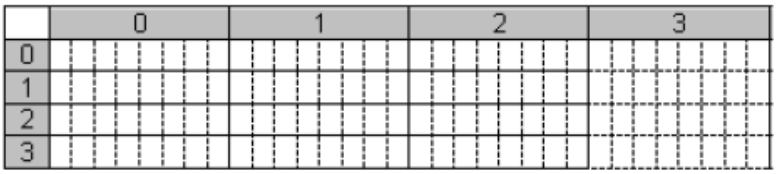
D = fungsi dekripsi C = ciphertext

P = plaintext K = key

2.5 Ukuran Teks Yang Disembunyikan

Semakin besar wadah (cover-image) yang digunakan untuk penyembunyian pesan maka semakin besar atau banyak pula jumlah karakter yang dapat disembunyikan dan semakin besar teks yang disembunyikan di dalam file, semakin besar pula kemungkinan teks tersebut rusak akibat manipulasi pada file penampung. Rumus untuk menghitung jumlah maksimal karakter yang dapat disisipkan ke gambar :

$$Max\ char = \frac{lebar\ gambar\ x\ panjang\ gambar}{8\ bit\ karakter} = \frac{200x200}{8} = 5000\ \text{char}$$



Gambar 2.2 Gambar berukuran 4x4x8b

Ukuran teks yang akan disembunyikan bergantung pada ukuran gambar yang dijadikan sebagai wadah penyimpanan. Misalnya Suatu gambar yang digunakan sebagai wadah untuk penyimpanan berukuran 200 x 200 x 8b, berarti gambar mempunyai panjang 200 piksel, lebar 200 piksel dan 8b menunjukan format pikselnya 8 bit. Gambar tersebut mempunyai 40000 piksel, karena 1 karakter terdiri dari 8 bit (ASCII) maka pesan disisipkan pada setiap 8 piksel sehingga teks maksimal yang dapat disembunyikan pada gambar adalah sebanyak 40000/8=5000 karakter.

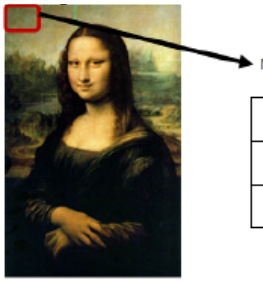
2.6 Teknik Steganografi Least Significant Bit (LSB)

Teknik Steganografi dengan menggunakan metode modifikasi Least Significant Bit (LSB) adalah teknik yang paaling sederhana, pendekatan yang sederhana untuk menyisipkan informasi di dalam suatu citra digital (medium cover). Mengkonversi suatu gambar dari format GIF atau BMP, yang merekonstruksi pesan yang sama dengan aslinya (lossless compression) ke JPEG yang lossy compression, dan ketika dilakukan kembali akan menghancurkan informasi yang tersembunyi dalam LSB.

Untuk menyembunyikan suatu gambar dalam LSB pada setiap byte dari gambar 24-bit, dapat disimpan 3 byte dalam setiap pixel. Gambar 1,024 x 768 mempunyai potensi untuk disembunyikan seluruhnya dari 2,359,296 bit (294,912 byte) pada informasi. Jika pesan tersebut dikompres untuk

disembunyikan sebelum ditempelkan, dapat menyembunyikan sejumlah besar dari informasi. Pada pandangan mata manusia, hasil stego-image akan terlihat sama dengan gambar cover.

2.6.1 Algoritma LSB Proses penyisipan (Embedding) pesan ke file Digit



Nilai Pixel Citra (RGB)

(100,120,80)	(90,75,190)	(100,80,80)
(80,180,35)	(80,122,200)	(85,120,100)
(100,120,80)	(80,122,200)	(80,122,200)

CONVERT KE BINER

Nilai Pixel Citra (Biner 8 bit)

(01100100,01111000,01010000)	(01011010,01001011,10111110)	(01100100,01010000,01010000)
(01010000,10110100,00100011)	(01010000,01111010,11001000)	(01010101,01111000,01100100)
(01100100,01111000,01010000)	(01010000,01111010,11001000)	(01100100,01111010,11001000)

PESAN YANG AKAN DISISIPKAN

ADALAH “AB”

NILAI ASCII “A” ADALAH 65,

BINERNYA 01000001

NILAI ASCII “B” ADALAH 66,

BINERNYA 01000010

EMBEDDING PESAN KE CITRA DIGITAL DENGAN MENGGANTI BIT TERAKHIR

Nilai Pixel Setelah Embedding

(01100100,01111001,01010000)	(01011010,01001010,10111110)	(01100100,01010001,01010000)
(01010001,10110100,00100010)	(01010000,01111010,11001001)	(01010100,01111000,01100100)
(01100100,01111000,01010000)	(01010000,01111010,11001000)	(01100100,01111010,11001000)

CONVERT NILAI BINER KE DESIMAL

Nilai Pixel Citra (RGB)

(100,121,80)	(90,74,190)	(100,81,80)
(81,180,34)	(80,122,201)	(84,120,100)
(100,120,80)	(80,122,200)	(80,122,200)

CONVERT NILAI DESIMAL KE PIXEL

Convert Nilai Desimal ke Pixel. maka Citra digital yang sudah disisipi pesan tidak tampak / perbedaanya jika di lihat dengan kasat mata.

2.6.2 Algoritma proses Dekripsi (Ekstraksi) Pesan dari File Digital

Nilai Pixel Citra (RGB)

(100,121,80)	(90,74,190)	(100,81,80)
(81,180,34)	(80,122,201)	(84,120,100)
(100,120,80)	(80,122,200)	(80,122,200)

CONVERT KE BINER

Nilai Pixel Dalam Biner 8 bit

(01100100,01111001,01010000)	(01011010,01001010,10111110)	(01100100,01010001,01010000)
(01010001,10110100,00100010)	(01010000,01111010,11001001)	(01010100,01111000,01100100)
(01100100,01111000,01010000)	(01010000,01111010,11001000)	(01100100,01111010,11001000)

AMBIL NILAI BIT TERAKHIR

Ambil Nilai Bit Terakhir:

01000001 = 65 = A

01000010 = 66 = B

PESAN HASIL EKSTRAKSI ADALAH “AB”

2.7 Enkripsi Vigenere Cipher

Sandi Vigenere adalah metode menyandi teks alphabet dengan menggunakan deretan sandi Caesar berdasarkan huruf-huruf pada kata kunci. Sandi Vigenere merupakan bentuk sederhana dari sandi polialfabetik. Kelebihan sandi ini dibanding sandi Caesar dan sandi mono alfabetik lainnya adalah sandi ini tidak begitu rentan terhadap metode pemecahan sandi yang disebut analisis frekuensi.

Giovan Batista Belaso menjelaskan metode ini dalam buku La cifra del. Sig. Giovan Batista Nelaso (1553) dan disempurnakan oleh diplomat Perancis Blaise de Vigenere pada tahun 1586. Pada abad ke19 banyak orang yang mengira vigenere adalah penemu sandi ini, sehingga sandi ini dikenal sebagai “sandi Vigenere”. Sandi ini dikenal dengan luas karena cara kerjanya mudah dimengerti dan dijalankan dan bagi para pemula sulit dipecahkan.

Pada saat kejayaannya, sandi ini dijuluki le chiffre indenchiffrable (bahasa perancis: “sandi yang tak terpecahkan”). Metode pemecahan sandi ini baru ditemukan pada abadke19. Pada tahun 1854, Charles Babbage menemukan cara untuk memecahkan sandi vigenere. Metode ini dinamakan tes Kasiski karena Friedrich Kasiskilah yang pertama mempublikasikannya. Kunci pada kriptografi Vigenere adalah sebuah kata bukan sebuah huruf. Kata kunci ini akan dibuat berulang sepanjang plaintext, sehingga jumlah huruf

pada kunci akan sama dengan jumlah huruf pada plaintext. Pergeseran setiap huruf pada plaintext akan ditentukan oleh huruf pada kunci yang mempunyai posisi yang sama dengan huruf pada plaintext.

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

Gambar 2.3 Bujur Sangkar Vigenere Cipher

Algoritma enkripsi vigenere cipher :

Ci = (Pi + Ki) mod 26

Algoritma dekripsi vigenere cipher : Pi = (Ci – Ki) mod 26

Dimana :

Ci = nilai desimal karakter ciphertext ke-i

Pi = nilai desimal karakter plaintext ke-i

Ki = nilai desimal karakter kunci ke-i.

Sebagai contoh,jika plaintext adalah THEBEAUTYANDTHEBEAST dan kunci adalah ABC maka proses enkripsi yang terjadi adalah sebagai berikut :

Plaintext : THEBEAUTYANDTHEBEAST

Kunci : ABCABCABCABCABCABCAB

Chipertext : TIGBFCUUAAOFTIGBFCSU

Pada contoh di atas kata kunci ABC diulang sedemikian rupa hingga panjang kunci sama dengan panjang plainteksnya. Kemudian setelah panjang kunci sama dengan panjang plainteks, proses enkripsi dilakukan dengan melakukan menggeser setiap huruf pada plainteks sesuai dengan huruf kunci yang bersesuaian dengan huruf plainteks tersebut.

Pada contoh di atas plainteks huruf pertama adalah T akan dilakukan pergeseran huruf dengan kunci Ki=0 (kunci huruf pertama adalah A yang memiliki Ki=0) menjadi T. Huruf kedua pada plainteks adalah H akan dilakukan pergeseran huruf dengan kunci Ki=1 (kunci huruf kedua adalah B yang memiliki Ki=1) menjadi

I. Begitu seterusnya dilakukan pergeseran sesuai dengan kunci pada tiap huruf hingga semua plainteks telah terenkripsi menjadi ciphertext.

2.8 Gambar Berformat Joint Photographic Experts Group (JPEG).

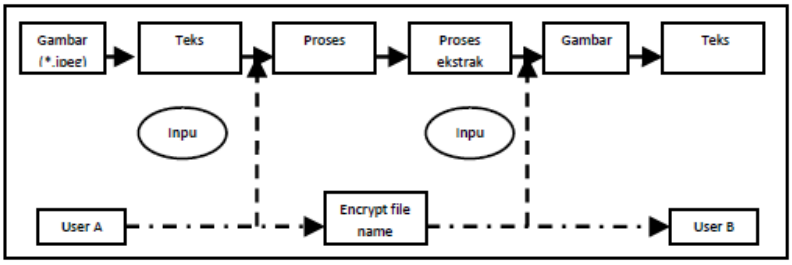
JPEG adalah suatu standar kompresi file continuoustone yang dikembangkan oleh Joint Photographic Expert Group. JPEG menggunakan teknik kompresi lossy sehingga sulit untuk proses pengeditan. JPEG cocok untuk file pemandangan (natural generated image), tidak cocok untuk file yang mengandung banyak garis, ketajaman warna, dan computer generated image.

JPEG merupakan nama teknik kompresi, sedangkan nama format filenya adalah JFIF (JPEG File Interchange Format). Tingkat kompresi yang baik untuk JPEG adalah 10:1-20:1 untuk file foto, 30:1-50:1 untuk file web, dan 60:1-100:1 untuk kualitas rendah seperti file untuk ponsel.

III. PERANCANGAN SISTEM

3.1 Metode Perancangan Sistem

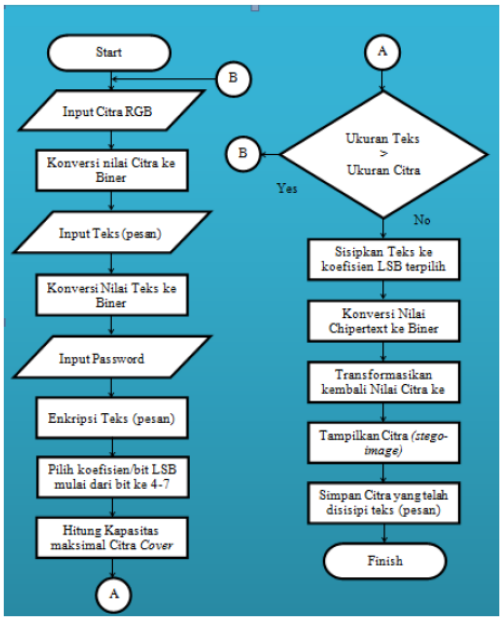
Metode perancangan sistem yang akan dipakai ialah dengan menggunakan metode diagram alir (flowchart). Diagram alir ini akan menjelaskan proses dari prosedur yang terjadi pada aplikasi dengan simbol-simbol tertentu sehingga dapat menggambarkan algoritma yang terjadi. Dengan penggunaan flowchart memungkinkan penggambaran keseluruhan dari pengambilan data awal hingga dihasilkan keluaran yang diinginkan. Blok diagram dari program implementasi secara umum dengan menggunakan Teknik Least Significant Bit yang diperlihatkan pada Gambar 3.1.



Gambar 3.1 Rancangan implementasi program secara umum.

3.1.1 Diagram alir program penyisipan teks ke dalam gambar

Diagram alir penyisipan teks ke dalam gambar dengan menggunakan Teknik Least Significant Bit pada perangkat lunak steganografi dapat dijabarkan dalam diagram alir pada Gambar 3.2.

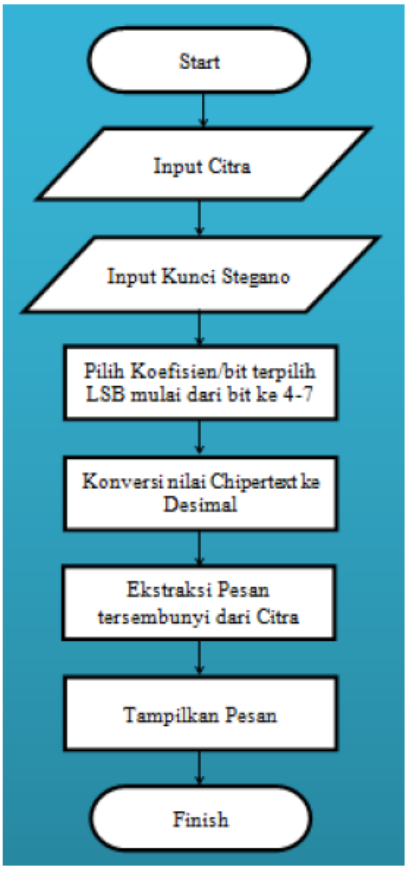


Gambar 3.2 Flowchart penyisipan (enkripsi) teks ke dalam gambar

Pada proses penyisipan (enkripsi) teks ke dalam gambar, gambar yang telah dipilih, nilai pixel file (cover-image) akan dikonversi ke biner (8-bit). Setelah itu memasukan atau memilih file teks (plaintext) yang akan disisipkan dan dilanjutkan dengan membangkitkan pseudo-number Vigenere Cipher. Setelah proses embedding selesai maka sistem akan menghitung daya tampung maksimal file, jika plaintext melebihi kapasitas kemampuan dari cover-image maka sistem akan menolak untuk melanjutkan dan meminta untuk mengurangi jumlah pesan. Jika cover-image mampu menampung maka proses melanjutkan mengkonversi nilai biner chiper-text (stego-image) ke nilai desimal dan dilanjutkan konversi nilai desimal ke nilai pixel dan akhirnya menyimpan (save) file cover-image yang sudah disisipi pesan (chiper-text).

3.1.2 Diagram alir program ekstraksi teks yang terdapat pada gambar

Diagram alir pengekstrakan teks yang ada pada gambar dapat dijelaskan pada Gambar 3.3.



Gambar 3.3 Flowchart untuk proses pengekstrakan teks yang ada dalam gambar

Pada proses ekstraksi *(dekrpsi)* teks yang terkandung di dalam *stego-image* dipilih dan nilai pixel citra *(stego-image)* akan dikonversi ke biner (8-bit) dan dilanjutkan dengan membangkitkan *pseudo-number* Vigenere Cipher. selanjutnya mengambil nilai bit terakhir (LSB) di tiap pixelnya, dan nilai biner dikonversi dari biner ke desimal dan pesan (*plain-text)* ditampilkan.

.

IV. HASIL PENELITIAN DAN PENGUJIAN

4.1 Batasan Implementasi

Aplikasi “Steganografi LSB dengan Enkripsi Vinegere Cipher Pada Citra Jpeg” ini dibuat dengan menggunakan bahasa pemrograman Visual Basic 6.0. Tahap implementasi merupakan tahap yang akan membangun sebuah sistem berdasarkan atas analisis kebutuhan sistem yang telah dirancang sehingga akan dihasilkan sistem yang dapat menghasilkan tujuan yang akan dicapai.

Sebelum program diterapkan dan diimplementasikan, maka program harus free error (bebas kesalahan). Kesalahan program yang mungkin terjadi antara lain kesalahan penulisan bahasa, kesalahan waktu proses, atau kesalahan logikal. Setelah program bebas dari kesalahan, program di tes dengan memasukkan data yang akan diolah.

4.2 Implementasi Antarmuka

4.2.1 Tampilan Menu

pembuatan perangkat lunak ini terdiri dari beberapa tahapan yaitu tahap pemrograman visual, tahap penulisan kode, dan tahap debugging. lunak yang dibuat dengan meng-compile program. Metode pengujian yang digunakan adalah trial and error, dimana setiap langkah yang menghasilkan output diteliti kembali keabsahannya, sehingga hasil yang didapatkan benar-benar dengan metode yang digunakan.

A. Proses Enkripsi :

1. Memilih gambar yang akan dijadikan media penyisipan.
2. Memasukkan atau mengetikkan teks yang akan disisipkan.
3. Memasukkan password
4. Memproses dan menyimpan file yang telah dilakukan proses steganografi.

B. Proses Dekripsi:

1. Memilih gambar yang akan diekstraksi yang sudah mengandung pesan.
2. Memasukkan kunci enkripsi (password).
3. Mengekstraksi file gambar (gambar stego) yang mengandung pesan proses steganografi sehingga menampilkan pesan yang tersembunyi di dalam gambar.

4.3 Pengujian

Pengujian merupakan tahap yang utama dalam pembuatan suatu aplikasi perangkat lunak. Hasil pengujian yang didapat, akan dijadikan sebagai tolak ukur dalam proses pengembangan selanjutnya. Pengujian ini dilakukan untuk mengetahui hasil yang didapat dari perangkat lunak yang telah dibuat. Aplikasi dibuat menggunakan perangkat lunak Microsoft Visual Basic 6.0 dan denganenggunakan Sistem Operasi Microsoft Windows 7 Home Premium.

Pelaksanaan Pengujian dengan Materi yang akan diujikan pada aplikasi Steganografi ini adalah sebagai berikut:

1. Penyisipan (enkripsi) Pesan.

Proses ini meliputi:

a. Input File (cover-image).

b. Input pesan.

c. Input Kunci Enkripsi (password).

2. Ekstraksi (dekripsi) Pesan.

Proses ini meliputi:

a. Input File (stego-image).

b. Input Kunci Enkripsi (password).

c. Ekstraksi (dekripsi) pesan.

3. Error

Akan dilakukan pengujian apakah error akan keluar apabila pengguna salah memasukkan sebuah data atau ukuran teks terlalu besar untuk disisipi ke dalam cover-image.

4.4 Analisis Hasil

Pada tahap ini akan dijelaskan analisis hasil kinerja dari aplikasi Steganografi LSB ini. Sebagai berikut:

4.4.1 Analisis Enkripsi Pesan

Pada proses penyisipan (enkripsi) teks ke dalam gambar, gambar yang telah dipilih, nilai pixel file(cover-image) akan dikonversi ke biner (8-bit). Setelah itu memasukan atau memilih file teks (plaintext) yang akan disisipkan dan dilanjutkan dengan membangkitkan pseudo-number Vigenere Cipher. Setelah proses embedding selesai maka sistem akan menghitung daya tampung maksimal file, jika plaintext melebihi kapasitas kemampuan dari cover-image maka sistem akan menolak untuk melanjutkan dan meminta untuk mengurangi jumlah pesan. Jika cover-image mampu menampung maka proses melanjutkan mengkonversi nilai biner chiper-text (stego-image) ke nilai desimal dan dilanjutkan konversi nilai desimal ke nilai pixel dan akhirnya menyimpan (save) file cover-image yang sudah disisipi pesan (chiper-text).

Sebagai contoh pengujian sebagai berikut :

Enkripsi Pesan.

Plaintext : “ATTACK”

Kunci : “NOW”

Nilai Citra Digital True Color 24-Bit RGB *(Cover-image)*

(100,121,80)	(90,74,190)	(100,81,80)
(81,180,34)	(80,122,201)	(84,120,100)
(100,120,80)	(80,122,200)	(80,122,200)
(100,121,80)	(90,74,190)	(100,81,80)
(81,180,34)	(80,122,201)	(84,120,100)
(100,120,80)	(80,122,200)	(80,122,200)

Enkripsi Vigenere Cipher :

Plaintext	A	T	T	A	C	K
Kunci	N	O	W	N	O	W
Ciphertext	N	H	P	N	Q	G

Konversi nilai piksel berkas citra digital 24-Bit RGB ke biner :

(01100100,01111000, 01010000)	(01011010,01001011, 10111110)	(01100100,01010000, 01010000)
(01010000,10110100, 00100011)	(01010000,01111010, 11001000)	(01010101,01111000, 01100100)
(01100100,01111000, 01010000)	(01010000,01111010, 11001000)	(01100100,01111010, 11001000)
(01100100,01111000, 01010000)	(01011010,01001011, 10111110)	(01100100,01010000, 01010000)
(01010000,10110100, 00100011)	(01010000,01111010, 11001000)	(01010101,01111000, 01100100)
(01100100,01111000, 01010000)	(01010000,01111010, 11001000)	(01100100,01111010, 11001000)

Konversi biner ke nilai piksel citra :

Nilai Citra Digital 24-Bit RGB yang sudah disisipi pesan.

(100, 121, 80)	(90, 75, 191)	(101, 80, 80)
(81, 180, 34)	(81, 122, 200)	(84, 120, 101)
(100, 121, 120)	(120, 122, 200)	(100, 123, 200)
(100, 121, 81)	(91, 74, 190)	(101, 80, 81)
(80 180, 34)	(81, 122, 201)	(84, 120, 100)
(101, 121, 81)	(120, 122, 200)	(100, 122, 200)

Konversi ciphertext ke biner :

1. Nilai ASCII huruf “N” : 78, Biner : 0100 1110
2. Nilai ASCII huruf “H” : 72, Biner : 0100 1000
3. Nilai ASCII huruf “P” : 80, Biner : 0101 0000
4. Nilai ASCII huruf “N” : 78, Biner : 0100 1110
5. Nilai ASCII huruf “Q” : 81, Biner : 0101 0001
6. Nilai ASCII huruf “G” : 71, Biner : 0100 0111

Nilai biner ciphertext disisipkan kedalam bit terakhir (LSB) berkas citra :

(01100100,01111001, 01010000)	(01011010,01001011, 10111111)	(01100101,01010000, 01010000)
(01010001,10110100, 00100010)	(01010001,01111010, 11001000)	(01010100,01111000, 01100101)
(01100100,01111001, 01010000)	(01010000,01111010, 11001000)	(01100100,01111011, 11001000)
(01100100,01111001, 01010001)	(01011011,01001010, 10111110)	(01100101,01010000, 01010001)
(01010000,10110100, 00100010)	(01010001,01111010, 11001001)	(01010100,01111000, 01100100)
(01100101,01111001, 01010001)	(01010000,01111010, 11001000)	(01100100,01111010,110 01000)

4.4.2 Analisis Dekripsi Pesan

Pada proses ekstraksi *(dekrpsi)* teks yang terkandung di dalam *stego-image* dipilih dan nilai pixel citra *(stego-image)* akan dikonversi ke biner (8-bit) dan dilanjutkan dengan membangkitkan *pseudo-number* Vigenere Cipher. selanjutnya mengambil nilai bit terakhir (LSB) di tiap pixelnya, dan nilai biner dikonversi dari biner ke desimal dan pesan (*plain-text)* ditampilkan. Hasil pendekripsian dengan password benar sebagai berikut:

Dekripsi Pesan.

Cipherteks : “NHPNQG”

Kunci : “NOW”

Nilai File Digital True Color 24-Bit RGB yang mengandung pesan *(stego-image).*

(100, 121, 80)	**(90, 75, 191)**	**(101, 80, 80)**
(81, 180, 34)	**(81, 122, 200)**	**(84, 120, 101)**
(100, 121, 120)	**(120, 122, 200)**	**(100, 123, 200)**
(100, 121, 81)	**(91, 74, 190)**	**(101, 80, 81)**
(80 180, 34)	**(81, 122, 201)**	**(84, 120, 100)**
(101, 121, 81)	**(120, 122, 200)**	**(100, 122, 200)**

Konversi nilai piksel berkas file stego-image ke biner :

(01100100,01111001,01010000)	(01011010,01001011,10111111)	(01100101,01010000,01010000)
(01010001,10110100,00100010)	(01010001,01111010,11001000)	(01010100,01111000,01100101)
(01100100,01111001,01010000)	(01010000,01111010,11001000)	(01100100,01111011,11001000)
(01100100,01111001,01010001)	(01011011,01001010,10111110)	(01100101,01010000,01010001)
(01010000,10110100,00100010)	(01010001,01111010,11001001)	(01010100,01111000,01100100)
(01100101,01111001,01010001)	(01010000,01111010,11001000)	(01100100,01111010,11001000)

Ambil nilai bit terakhir tiap byte (LSB) dan susun menjadi tiap 8-bit :
01001110 01001000 01010000
01001110 01010001 01000111

Konversi Nilai Biner Ciphertext ke Desimal dan Alfabet :

0100 1110	0100 1000	0101 0000	0100 1110	0101 0001	0100 0111
78	72	80	78	81	71
N	H	P	N	Q	G

Hasil dari Dekripsi Vigenere Cipher :

Ciphertext	N	H	P	N	Q	G
Kunci	N	O	W	N	O	W
Plaintext	A	T	T	A	C	K

Gambar 4.2 Hasil plaintext dengan kunci yang benar.

4.4.3 Pesan *Error* / Tidak bisa dibaca.

Berikut hasil jika penerima salah memasukkan password:

Cipherteks : “NHPNQG”

Kunci : “WON”

Plainteks : “8T]8CT”

V. Kesimpulan dan Saran.

5.1 Kesimpulan

Kesimpulan yang dapat diambil dari penulisan laporan tugas akhir ini adalah sebagai berikut :

1. Implementasi algoritma LSB (*Least Significant Bit*) dengan enkripsi Vinegere Cipher dapat digunakan cukup baik untuk menyembunyikan pesan di dalam pesan sebuah berkas file multimedia sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu di dalam pesan tersebut.
2. Pada proses ekstraksi, pesan atau informasi yang disisispkan pada file multimedia uji dalam aplikasi Steganografi ini, dapat diperoleh kembali secara utuh atau dengan kata lain pesan yang disisipkan sebelum proses penyisipan dan setelah proses ekstraksi sama tanpa ada perubahan kecuali pengguna memasukkan *password* yang salah.

5.2 Saran

Saran dari penulisan laporan tugas akhir ini adalah sebagai berikut :

1. Algoritma pada implementasi steganografi ini kurang kuat meskipun untuk memecahkannya cukup sulit, tetapi jika kelemahan utamanya ditemukan maka sangat mudah untuk memecahkannya.
2. Kedepannya diharapkan dapat dikembangkan suatu aplikasi Steganografi dengan metode lain yang lebih variatif dan lebih seperti baik seperti algoritma Data Enkripsi Encryption Standard (DES), Triple Data Encryption Standard (3DES), RC4, Anvanced Encryption Standard (AES) dan lain-lain agar pesan tersembunyi menjadi sangat sulit terdeteksi dan ukuran serta kualitas file yang dihasikan tidak jauh berbeda dengan kualitas file sebelumnya.

Sumber Makalah

http://www.elektro.undip.ac.id/el_kpta/wp-content/uploads/2012/05/L2F306054_MTA.pdf