

IMPLEMENTASI PENGAMANAN DATA DAN INFORMASI DENGAN METODE STEGANOGRAFI LSB DAN ALGORITMA KRIPTOGRAFI AES

Syaiful Anwar

Magister Ilmu Komputer, Universitas Budi Luhur
Jalan Ciledug Raya, Petukangan Utara, Jakarta Selatan 12260
Email :syaiful.byz@gmail.com

Abstrak

Keamanan dan kerahasiaan merupakan dua aspek penting dalam komunikasi data. Dalam menjaga keamanan dan kerahasiaan suatu pesan penting yang akan dikirimkan, pesan terlebih dahulu dienkripsi dan disembunyikan ke dalam suatu media. Media yang digunakan dalam penelitian ini adalah citra digital dengan kedalaman warna 24 bit. Penelitian ini bertujuan untuk membangun suatu sistem yang dapat digunakan dalam meningkatkan keamanan data yang berupa pesan-pesan penting dan rahasia. Algoritma kriptografi yang digunakan untuk melakukan enkripsi adalah algoritma AES dan steganografi dengan memodifikasi metode Least Significant Bit (LSB) digunakan untuk menyimpan pesan ke dalam citra. Modified LSB yang digunakan dalam penelitian ini menyisipkan bit-bit cipherteks ke dalam diagonal matriks pixel komponen warna pada citra. Sistem ini dikembangkan dengan menggunakan bahasa pemrograman C#. Pengujian yang dilakukan pada penelitian ini melihat aspek imperceptibility dan aspek recovery pada metode Modified LSB. Penelitian ini menegaskan bahwa kombinasi dari algoritma AES dan Modified LSB dapat digunakan dalam meningkatkan keamanan data.

Kata kunci:keamanan data, kriptografi, steganografi, AES, LSB.

1. Pendahuluan

Perkembangan teknologi informasi semakin memudahkan penggunaannya dalam berkomunikasi melalui bermacam-macam media. Komunikasi yang melibatkan pengiriman dan penerimaan pesan dengan memanfaatkan kemajuan teknologi informasi rentan terhadap pelaku kejahatan komputer yang memanfaatkan celah keamanan untuk mendeteksi dan memanipulasi pesan.

Keamanan dan kerahasiaan menjadi aspek yang sangat penting bagi pengguna teknologi informasi. Untuk menghindari pesan yang dikirimkan jatuh pada pihak-pihak yang tidak berkepentingan dan terjadi penyalahgunaan terhadap pesan, maka dilakukan enkripsi terhadap pesan asli dan penyisipan pesan ke

dalam suatu media dengan menerapkan ilmu kriptografi dan steganografi.

Untuk meningkatkan keamanan digunakan kombinasi antara kriptografi dan steganografi, dimana pesan rahasia dienkripsi terlebih dahulu, kemudian cipherteks disembunyikan di dalam media lain sehingga pihak-pihak yang tidak berkepentingan tidak menyadari keberadaan pesan.

Berdasarkan latar belakang masalah, proses pertukaran pesan memerlukan jaminan keamanan dan kerahasiaan. Diperlukan pengembangan teknik keamanan yang dapat memberikan proteksi lebih baik pada pesan rahasia, dan menjaga kerahasiaan pesandengan menyembunyikannya ke dalam media lain(gambar) agar keberadaan pesan rahasia tidak diketahui.

Adapun tujuan yang ingin dicapai adalah merancang sebuah aplikasi yang dapat mengenkripsi dan mendekripsi pesan teks menggunakan algoritma kriptografi AES dan juga Merancang sebuah aplikasi yang dapat menyisipkan dan mengekstrak cipherteks berupa blok-blok integer dalam media berupa citra digital menggunakan algoritma LSB.

Tinjauan Studi

Khalil Challita dan Hikmat Farhat[9] melakukan penelitian mengenai *Combining Steganografi and Cryptography : new Directions* dengan kombinasi algoritma MCO (*multiple cover object*). Membuat kesepakatan antara pengirim dan penerima pesan dalam informasi password yang digunakan sebagai kata kunci.

Kavita Kadam, Ashwini Koshti dan Priya Dughav [10] melakukan penelitian mengenai *Steganography Using Least Signicant Btt Algorithm* dengan kombinasi algoritma DCT (*Discrete cosine trabsformations*). Menyisipkan pesan rahasia edalam gambar yang dilindungi dengan password pribadi yang terenkripsi.

M. Anggrie Andriawanm, Solikin Dan setia Juli Irzal Ismail [11] melakukan penelitian mengenai Implementasi *steganografi* pada citra digital File gambar Bitmap (Bmp) Menggunakan java dengan penyisipan pesan ke dalam bit terendah (LSB) bitmap 24 bit. Menyembunyikan pesan rahasia dengan metode LSB untuk mengeksploitasi keterbatasan sistem penglihatan manusia.

Steganografi

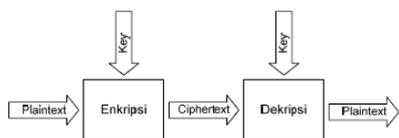
Dalam teknologi komputer pengamanan data dengan *steganografi* dapat dilakukan dengan dua cara [1]. Cara pertama melibatkan satu file saja sebagai file media atau file *carrier*. Dan cara kedua dengan cara melibatkan dua file, yaitu file yang memuat data rahasia yang akan disembunyikan dan file lain adalah file media atau *carrier*.

Ada beberapa kriteria yang harus diperhatikan dalam steganografi[6], yaitu :

- 1) *Imperceptibility*. Keberadaan pesan rahasia tidak dapat dipersepsi oleh inderawi. Misalnya, jika *coverttext* berupa citra, maka penyisipan pesan membuat citra *stegotext* sukar dibedakan oleh mata dengan citra *coverttext*-nya. Jika *coverttext* berupa audio, maka indera telinga tidak dapat mendeteksi perubahan pada audio *stegotext*-nya.
- 2) *Fidelity*. Mutu *stegomedium* tidak berubah banyak akibat penyisipan. Perubahan tersebut tidak dapat dipersepsi oleh inderawi. Misalnya, jika *coverttext* berupa citra, maka penyisipan pesan membuat citra *stegotext* sukar dibedakan oleh mata dengan citra *coverttext*-nya. Jika *coverttext* berupa audio, maka audio *stegotext* tidak rusak dan indera telinga tidak dapat mendeteksi perubahan tersebut.
- 3) *Recovery*. Pesan yang disembunyikan harus dapat diungkapkan kembali. Karena tujuan steganografi adalah data hiding, maka sewaktu-waktu pesan rahasia di dalam *stegotext* harus dapat diambil kembali untuk digunakan lebih lanjut.

Kriptografi

Kriptografi pada dasarnya terdiri dari dua proses, yaitu proses *enkripsi* dan proses *dekripsi*. Proses *enkripsi* adalah proses penyandian pesan terbuka menjadi pesan rahasia (*ciphertext*). *Ciphertext* inilah yang nantinya akan dikirimkan melalui saluran komunikasi terbuka. Pada saat *ciphertext* diterima oleh penerima pesan, maka pesan rahasia tersebut diubah lagi menjadi pesan terbuka melalui proses *dekripsi* sehingga pesan tadi dapat dibaca kembali oleh penerima pesan. Secara umum, proses *enkripsi* dan *dekripsi* dapat digambarkan sebagai berikut:



Gambar 1. Proses Enkripsi dan Dekripsi

Metode LSB (Least Significant Bit)

Metode LSB (Least Significant Bit) merupakan salah satu teknik substitusi pada steganografi. Dimana tiap bit terendah pada byte-byte media citra akan digantikan dengan bit-bit pesan yang akan disisipkan. Pada file citra 24 bit setiap pixel pada citra terdiri dari susunan tiga

warna, yaitu merah, hijau dan biru (RGB) yang masing-masing disusun oleh bilangan 8 bit (1 byte) dari 0 sampai 255 atau dengan format biner 00000000 sampai 11111111. Informasi dari warna biru berada pada bit 1 sampai bit 8, dan informasi warna hijau berada pada bit 9 sampai dengan bit 16, sedangkan informasi warna merah berada pada bit 17 sampai dengan bit 24.

Metode LSB hanya mengubah nilai byte satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya, sehingga perubahan yang terjadi tidak begitu berarti. Lagi pula, mata manusia tidak dapat membedakan perubahan kecil yang terjadi tersebut.

Algoritma AES (Advanced Encryption Standard)

Sejak tahun 1976, Data Encryption Standard (DES) dipilih sebagai standar kriptografi yang dipakai pada pemerintahan Amerika Serikat. Namun pada tahun 1990, panjang kunci DES dianggap terlalu pendek, dan pada tahun 1998 DES berhasil dipecahkan dalam waktu 96 hari, kemudian di tahun 1999 dapat dipecahkan dalam waktu 22 hari.

Karena alasan tersebut maka kemudian diadakan kompetisi oleh NIST (National Institute of Standard and Technology) untuk mencari pengganti DES. NIST mengundang peserta dari seluruh dunia untuk berpartisipasi dengan mengajukan algoritma baru untuk menggantikan DES [3].

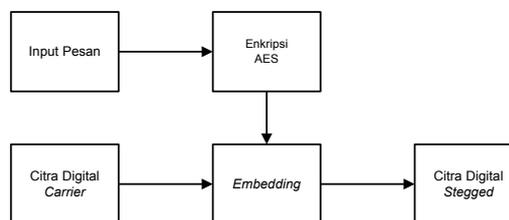
2. Pembahasan

Analisis Sistem

Secara umum proses *steganografi* pada sistem ini ada dua, yaitu proses penyisipan pesan (*embedding*) dan proses pengungkapan pesan (*ekstraksi*).

1.) Proses Embedding Pesan

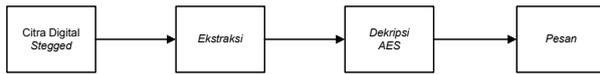
Ada dua tahap yang dilakukan dalam proses *embedding*, yaitu *enkripsi aes* yang mentransformasikan pesan asli (*plaintext*) menjadi teks acak (*ciphertext*), selanjutnya dilakukan penyisipan (*embedding*) dalam citra digital pembawa (*carrier*). Ilustrasi dari proses *embedding* dapat dilihat pada gambar 2.



Gambar 2. Proses Embedding

2.) Proses Ekstraksi Pesan

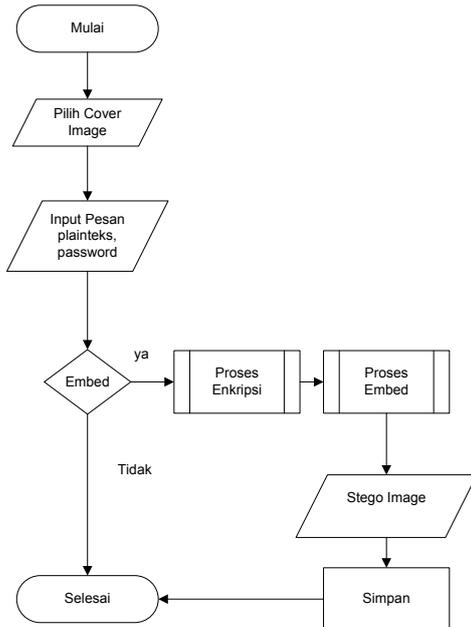
Dalam proses ekstraksi ada dua tahap yang dilakukan yaitu seperti pada gambar 3 berikut:



Gambar 3. Proses Ekstraksi

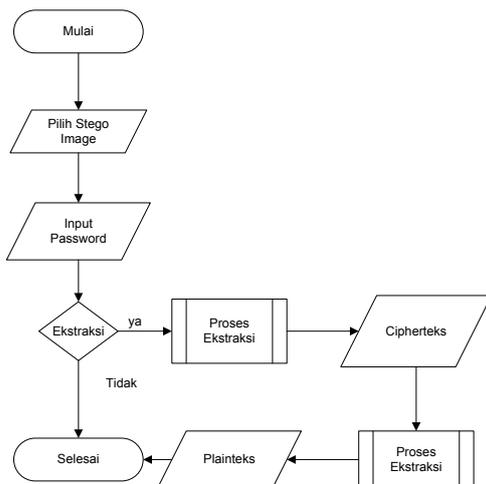
Perancangan Sistem

Untuk menyisipkan cipherteks ke dalam *cover image* penyembunyian pesan dilakukan dengan menggunakan metode *modified lsb* yang mengganti diagonal-diagonal *byte* komponen warna merah pada citra. Proses penyembunyian pesan tersebut dapat digambarkan dengan *flowchart* berikut:



Gambar 4. Flowchart Proses Embedding

Ekstraksi dilakukan untuk memisahkan cipherteks yang tersembunyi dari *stego image*. Flowchart untuk proses ekstraksi diperlihatkan pada gambar 5.



Gambar 5. Flowchart Proses ekstraksi

Implementasi Antarmuka Sistem

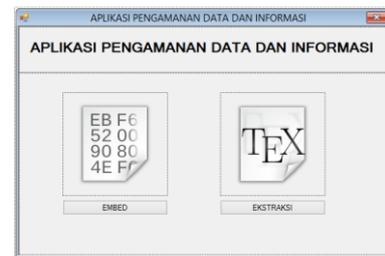
Dalam proses ini, penulis menggunakan Microsoft Visual Studio 2010 sebagai alat pengembangan untuk

menerapkan kriptografi dan steganografi dengan menggunakan C # sebagai bahasa pemrograman. Proses ini akan menerapkan coding fungsi dan mengintegrasikan fungsi ke dalam GUI (*Graphical User interface*).

Implementasi dari hasil tahapan analisis dan perancangan dapat dilihat dari tampilan antarmuka sistem sebagai berikut:

1) Tampilan Halaman Menu Utama

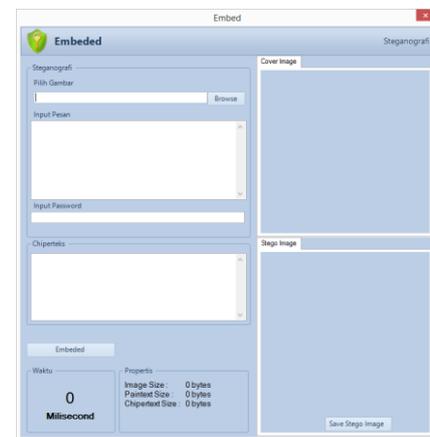
Halaman Menu utama merupakan halaman awal yang akan ditampilkan saat sistem dijalankan. Tampilan Halaman Menu utama ditunjukkan pada Gambar 6.



Gambar 6. Halaman Menu utama

2) Halaman Embed

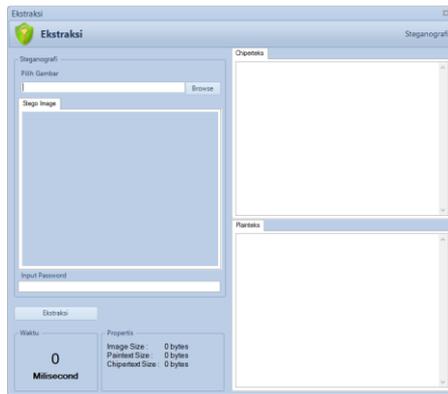
Halaman *Embed* merupakan halaman untuk proses enkripsi pesan menjadi *cipherteks* dan penyisipan bit – bit *cipherteks* dan penyisipan bit bit *cipherteks* kedalam *cover image*, proses tersebut terjadi pada pihak pengirim pesan. Pada halaman *embed* ini, *user* atau pengirim pesan akan menginputkan pesan yang akan dikirimkan kepada penerima pesan dan juga memilih *cover image*. gambar 7 berikut menampilkan halaman *embed*.



Gambar 7. Halaman Embed

3) Tampilan Halaman Ekstraksi

Halaman *ekstraksi* merupakan halaman tempat berlangsungnya proses *ekstraksi* pesan dari *stego image* dan *dekripsi cipherteks* ke *plainteks* agar dapat dibaca oleh penerima pesan. Untuk dapat mendekripsikan *cipherteks*, penerima pesan membutuhkan *password* yang sama, yang sebelumnya digunakan untuk melakukan *enkripsi* pesan oleh pengirim pesan. Gambar 8 berikut adalah tampilan halaman *ekstraksi*.



Gambar 8. Halaman Ekstraksi

Pengujian Sistem

1.) Aspek imperceptibility

Pengujian diberikan terhadap beberapa *cover image* yang telah disisipi pesan terlebih dahulu untuk membuktikan apakah algoritma *Modified LSB* telah memenuhi aspek *imperceptibility* atau tidak. Dari penyisipan pesan tersebut akan dihasilkan *stego image*. Dimana aspek *imperceptibility* akan terlihat dari perbandingan antara kedua berkas citra digital tersebut tidak dapat dilihat secara kasat mata, maka dapat ditarik kesimpulan bahwa algoritma *Modified LSB* yang digunakan untuk menyembunyikan pesan telah memenuhi aspek *imperceptibility*.

Tabel I. Tabel Hasil Pengujian pada beberapa Cover Image

No	Plainteks / size	Kunci	Cipherteks / size	Cover Image	Stego Image	Waktu (ms)
1.	gambar ini telah disisipi pesan yang harus di jaga kerahasiaan. (62 bytes)	qwerty! @#8%	EAAAAK+aRHPa 7nV6h8fJZPTlyscg grOYXamWQGBg 9v5kZnECZOUR8 4R8U8jx2ozkOIXI 89HBpc/tat4jrKT0 V0A/d3gZfKd/8iSq F7wnQy4+PbK+ (112bytes)	Property Value Image Dimensions 300 x 321 Width 300 pixels Height 321 pixels Bit depth 24 File Name 1.bmp Item type Bitmap image Folder path E:\Kuliah UBL\Kuliah Semester 2\Tu Date created 5/22/2015 20:26 Date modified 6/6/2014 06:48 Size 282 KB	Property Value Image Dimensions 300 x 321 Width 300 pixels Height 321 pixels Bit depth 24 File Name stego 1.bmp Item type Bitmap image Folder path E:\Kuliah UBL\Kuliah Semester 2\Tugas Date created 5/22/2015 20:57 Date modified 5/22/2015 20:57 Size 282 KB	21
2.	Abcdefghijklmnopq rstuvwxyz (26 bytes)	1234567 890	EAAAALY2BW m4A8j58vZ/Ds6E V8TqbiufpGC4Y2 wkHrvABK3wO9t VYWb4qhndk8+ B1KhEMQ== (72bytes)	Property Value Image Dimensions 300 x 321 Width 300 pixels Height 321 pixels Bit depth 24 File Name 2.bmp Item type Bitmap image Folder path E:\Kuliah UBL\Kuliah Semester Date created 5/22/2015 20:26 Date modified 6/6/2014 06:48 Size 282 KB	Property Value Image Dimensions 300 x 321 Width 300 pixels Height 321 pixels Bit depth 24 File Name stego 2.bmp Item type Bitmap image Folder path E:\Kuliah UBL\Kuliah Semester 2\Tugas Date created 5/22/2015 23:05 Date modified 5/22/2015 23:05 Size 282 KB	17
3.	Setelah aplikasi ini melewati proses tahap <i>coding</i> , maka tahap selanjutnya adalah tahap pengujian. (98 bytes)	asdfghjk l	EAAAAJUIDK9Z yQXMm6i2ihLG3 7mSxYffOoj19ta AR9cENUwuGM MyQqZU67XEcV zw4EwSAZ5Ptj1j atzyp9MCgzW0D qvW8ce7kEhCI9 wDXpAeDA4LdC 1KO959i2VBFP2 Z0BvADiUU96Z KbrmB3VUoOFe 2PR4dageA34ytm jbGH5DY21g (176 bytes)	Property Value Image Dimensions 420 x 240 Width 420 pixels Height 240 pixels Bit depth 24 File Name Cover Image.bmp Item type Bitmap image Folder path E:\Kuliah UBL\Kuliah S Date created 5/22/2015 15:43 Date modified 5/22/2015 15:43 Size 295 KB	Property Value Image Dimensions 420 x 240 Width 420 pixels Height 240 pixels Bit depth 24 File Name stego Cover Image.bmp Item type Bitmap image Folder path E:\Kuliah UBL\Kuliah Semester 2\Tugas Date created 5/22/2015 23:30 Date modified 5/22/2015 23:30 Size 157 KB	24

Dari beberapa pengujian yang telah dilakukan terlihat bahwa *cover image* dengan *stego image* telah berisi *cipherteks* secara kasat mata terlihat sama dan tidak terlihat perbedaan sedikitpun, tidak hanya itu ukuran citra dan juga dimensi citra sebelum dan sesudah disisipkan pesan tidak mengalami perubahan. Hal ini disebabkan perubahan byte-byte diagonal komponen warna pada citra hanya akan menghasilkan perubahan 1 byte lebih tinggi atau lebih rendah, pergantian tersebut tidak akan menampilkan perubahan yang berarti pada *stego image*. Dengan demikian metode *Modified LSB* telah memenuhi aspek *imperceptibility* sebagai salah satu kriteria algoritma *steganografi* yang baik.

2.) *Aspek Recovery*

Aspek *recovery* menyatakan bahwa pesan yang disembunyikan dalam *stego object* harus dapat diungkap kembali. Untuk mengatur keberhasilan aspek *recovery* dalam algoritma *modified lsb*, dapat dilihat dari kesesuaian *cipherteks* yang berhasil diekstraksi dari *stego image* dengan *cipherteks* hasil dari proses enkripsi.

Hasil pengujian terhadap aspek *recovery* untuk contoh masukan *stego image* hasil embedding pada pengujian sebelumnya ditunjukkan tabel II.

Tabel II. Tabel hasil pengujian pada beberapa *stego image*

No	Stego Image	Cipherteks sebelum	Cipherteks sesudah	Password	Plainteks hasil	Waktu (ms)
1.		EAAAAK+aRHPa7nV 6h8fJZPTlyscggrOYXa mWQGBg9v5kZnECZ OUR84R8U8jx2ozkOI XI89HBpc/tat4jrKT0V 0A/d3gZfKd/8iSqF7wn Qy4+Pb+ (112 bytes)	EAAAAK+aRHPa7 nV6h8fJZPTlyscggr OYXamWQGBg9v5 kZnECZOUR84R8 U8jx2ozkOIXI89HB pc/tat4jrKT0V0A/d 3gZfKd/8iSqF7wnQ y4+PbK+ (112 bytes)	qwerty!@# \$%	gambar ini telah disisipi pesan yang harus di jaga kerahasiaan. (62 bytes)	22
2.		EAAAALY2BWm4A 8j58vZ/Ds6EV8Tqbiuf pGC4Y2wkHrvABK3 wO9tVYWb4qhndk8+ B1KhEMQ== (72 bytes)	EAAAALY2BWm4A 8j58vZ/Ds6EV8Tqb iufpGC4Y2wkHrvA BK3wO9tVYWb4qh ndk8+B1KhEMQ= = (72 bytes)	123456789 0	Abcdefghijklm nopqrstuvwxyz (26 bytes)	18
3.		EAAA AJUIDK9ZyQ XMm6i2ihLG37mSxY ffOoj19taAR9cENUw uGMMYqQZU67XEc Vzw4EwSAZ5Ptj1jat yp9MCgzW0DqvW8c e7kEhCI9wDXpAeDA 4LdC1KO959i2VBFP 2Z0BvADiUU96ZKbr rnB3VUoOFe2PR4dag cA34ytmjbGH5DY21 g (176 bytes)	EAAA AJUIDK9Zy QXMm6i2ihLG37 mSxYffOoj19taAR 9cENUwuGMMYq qZU67XEcVzw4E wSAZ5Ptj1jatyp9 MCgzW0DqvW8ce 7kEhCI9wDXpAe DA4LdC1KO959i2 VBFP2Z0BvADiU U96ZKbrnB3VUo OFe2PR4dagA34y tmjbGH5DY21g (176 bytes)	asdfghjkl	Setelah aplikasi ini melewati proses tahap coding, maka tahap selanjutnya adalah tahap pengujian. (98 bytes)	19

Dari beberapa pengujian yang dilakukan dapat dilihat bahwa hasil ekstraksi dari *stego image* menghasilkan *cipherteks* yang sesuai dengan *cipherteks* hasil enkripsi, dan untuk proses dengan kunci yang tepat akan menghasilkan *plainteks* sama seperti yang diinputkan semula. Dengan demikian, algoritma *modified lsb* yang digunakan dalam penelitian ini telah memenuhi aspek *recovery*.

3. Kesimpulan

Dari hasil percobaan yang telah dilakukan, maka dapat disimpulkan bahwa implementasi algoritma kriptografi aes dan steganografi dengan metode lbs cukup berhasil.

- 1) Pengujian terhadap beberapa sample membuktikan bahwa metode Modified LSB memenuhi aspek *imperceptibility*, dimana keberadaan pesan rahasia pada citra digital sulit untuk dipersepsi oleh inderawi.

Hal ini karena perubahan yang terjadi tidak begitu berarti dan tidak menghasilkan perbedaan yang mencolok terhadap stego object. Pengujian terhadap aspek recovery menunjukkan bahwa cipherteks dapat diekstraksi dengan tepat menggunakan metode Modified LSB

- 2) Pengimplementasian teknik kriptografi AES dan steganografi dengan metode LBS pada data citra RGB berhasil dan berjalan dengan baik. Semakin besar ukuran pesan semakin lama proses enkripsi dan embed.

Daftar Pustaka

- [1.] Ariyus, D.. "Kriptografi Keamanan Data dan Komunikasi". Yogyakarta: Graha Ilmu. 2006
- [2.] Ariyus, D. Keamanan Multimedia. Yogyakarta : Penerbit Andi 2007.
- [3.] Ariyus, D. "Pengantar Ilmu Kriptografi Teori, Analisis, dan Implementasi". Yogyakarta: ANDI Offset 2008.
- [4.] Chandraleka, H. "Mengamankan Data Pribadi ala Agen Rahasia". Jakarta: Elex Media Komputindo 2009.
- [5.] Daemen, J. & Rijmen, V. 1999. AES Proposal: Rijndael. <http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf> (18 Mei 2015).
- [6.] Munir, R. "Kriptografi. Informatika" Bandung : Bandung 2006.
- [7.] Nechvatal J. et al. "Report on the Development of the Advanced Encryption Standard (AES)". Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Administration U.S. Department of Commerce 2000.
- [8.] Edition. New Jersey: "Pearson Education. Stallings, W.. Cryptography and Network Security Principles and Practice. Third 2003"
- [9.] Khalil Challita, Hikmat Farhat, "Combining Steganografi and Cryptography : new Directions dengan kombinasi algoritma MCO (multiple cover object)", 2012
- [10.] Kavita Kadam, Ashwini Koshti dan Priya Dunghav, "Steganography Using Least Significant Bit Algorithm dengan kombinasi algoritma DCT (Discrete cosine transformations)", 2012
- [11.] M. Anggrie Andriawanm, Solikin Dan setia Juli Irzal Ismail, "Implementasi steganografi pada citra digital File gambar Bitmap (Bmp) Menggunakan java dengan penyisipan pesan ke dalam bit terendah (LSB) bitmap 24 bit".

Biodata Penulis

Syaiful Anwar, memperoleh gelar Sarjana Komputer (S.Kom), Jurusan Teknik Informatika Universitas Budi Luhur Jakarta, lulus tahun 2004. Saat ini menjadi Mahasiswa Pascasarjana di Universitas Budi Luhur Jakarta.