Penerapan Algoritma Advanced Encryption Standard dalam Mengamankan File pada Citra dengan Metode Least Significant Bit

Modesty Sri Pebriyani Manurung
Teknik Informatika STMIK Budi Darma Medan Jl. Sisingamangaraja No. 338 - Medan
e-mail : destymanroe@gmail.com

Abstrak

Seiring dengan meningkatnya kejahatan komputer yang terus berkembang yang dilakukan oleh sebagian orang yang tidak bertanggung jawab, dimana aktivitas mereka dalam pencurian atapun manipulasi data sangat mengganggu privasi seseorang. Dengan adanya kemajauan telekomunikasi dan komputer juga memungkinkan pengguna melakukan penyimpanan file secara digital. Dalam hal ini masalah keamanan dan kerahasian file adalah suatu hal yang sangat penting, maka harus ada perlindungan terhadap file yang dirahasiakan. Sebuah teknik dalam ilmu kriptografi merupakan salah satu cara yang dapat mengamankan data dari ganguan orang lain.

Kriptografi merupakan seni dalam mengamankan pesan menjadi suatu pesan yang tidak dikenali. Advance Encryption Standard (AES) merupakan salah algoritma enkripsi kriptografi yang digunakan. Namun dengan menggunakan satu metode masih dapat menimbulkan suatu kecurigaan, untuk melengakapi agar file tersembunyi dengan aman dan tidak menimbulkan kecurigaan. Penggunaan steganografi least significant bit (LSB) menjadi salah satu pilihin yang tepat. Least significant bit merupakan suatu metode untuk menyisipkan potongan sebuah informasi rahasia dalam suatu objek media lain seperti pada gambar. Metode ini tidak menimbulkan perubahan yang besar terhadap gambar yang digunakan secara kasat mata.

Kata Kunci *: Advance Encryption Standard, Kriptografi, Least Significant Bit, Steganografi.*

Abstract

Along with the increasing number of computer crimes that are constantly developing by some irresponsible people, where their activities in theft or manipulation of data greatly disrupt someone's privacy. With the availability of telecommunications and computers, it also allows users to store files digitally. In this case, security issues and file confidentiality are very important, so there must be protection against confidential files. A technique in cryptography is one way that can secure data from other people's problems.

Cryptography is the art of securing messages into an unknown message. The Advance Encryption Standard (AES) is one of the cryptographic encryption algorithms used. But using one method can still raise a suspicion, to avoid keeping the files hidden safely and not arouse suspicion. The use of least significant bit (LSB) steganography is one of the right choices. The Least significant bit is a method for inserting a piece of confidential information in another media object as shown. This method does not cause major changes to the images used in plain view.

Keywords *: Advanced Encryption Standard, Cryptography, Least Significant Bit, Steganography.*

1. PENDAHULUAN

Keamanan telah menjadi aspek yang sangat penting dari suatu *file*. Sebuah *file* umumnya hanya ditujukan bagi segolongan tertentu. Oleh karena itu, sangat penting untuk mencegahnya jatuh ketangan pihak-pihak lain yang tidak berkepentingan. Untuk melaksanakan tujuan tersebutlah dirancang suatu sistem keamanan yang berfungsi melindungi *file*. Salah satu cara untuk mengamankan *file* adalah dengan menyembunyikan *file* ke dalam bentuk media citra. Pengamanan *file* dalam bentuk citra menjadi salah satu solusi untuk keamanan sebuah *file* yang bersifat rahasia jika *file* tersebut ingin dikirimkan.

Algoritma kriptografi yang akan digunakan untuk mengenkripsi *file* adalah algoritma *Advanced Encryption Standard* (AES). Sedangkan steganografi merupakan salah satu teknik yang digunakan dalam pengamanan informasi, yaitu dengan menyembunyikan informasi ke dalam media digital dengan metode tertentu agar tidak tampak perbedaan secara visual antara *file* asli dengan *file* yang telah disisipi informasi (*stegoimage*) sehingga tidak diketahui oleh steganalis (orang yang dapat memecahkan *stegoimage* tanpa mengetahui kunci yang ada).

Dengan menggabungkan metode *Least Significant Bit* (LSB) dan algoritma *Advanced Encryption Standard* (AES) dapat meningkatkan keamanan *file*. *File* disandikan dengan menggunakan algoritma *Advanced Encryption Standard* (AES) dan disembunyikan pada media citra menggunakan metode *Least Significant Bit* (LSB). Penggunaan teknik steganografi dan kriptografi dimaksudkan untuk memberikan keamanan berlapis dalam pengamanan *file.*

2. METODOLOGI PENELITIAN

2.1. Steganografi

Steganografi merupakan seni untuk menyembunyikan pesan di dalam media digital sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu pesan di dalam media tersebut. Kata steganofrafi (*steganography*) berasall dari bahasa yunani steganos yang artinya "tersembunyi/terselubung" dan graphein "menulis" sehingga kurang lebih artinya "menulis (tulisan) terselubung" [4].

Penyembunyian data rahasia ke dalam citra digital akan mengubah kualitas citra tersebut. Hal ini tergantung pada ukuran file media penyimpanan dan ukuran file pesan yang disisipkan. Untuk itu ada beberapa hal atau criteria yang harus diperhatikan dalam penyembunyian data, yaitu :

a. *Fidelity*
Mutu citra penampung data tidak jauh berbeda. Setelah terjadi penambahan pesan rahasia, *stego-data* masih terlihat dengan baik. Pengamat tidak mengetahui kalau di dalam *stego-data* tersebut terdapat pesan rahasia.

b. *Robustness*
Pesan yang disembunyikan harus tahan (*robust*) terhadap berbagai operasi manipulasi yang dilakukan pada *stego-data*, seperti pengubahan kontras, penajaman, pemampatan, rotasi, perbesaran gambar, pemotongan *cropping*, enkripsi, dan sebagainya.

c. *Recovery*
Data yang disembunyikan harus dapat diungkap kembali (*recovery*). Karena tujuan steganografi adalah penyembunyian informasi maka sewaktu-waktu pesan rahasia di dalam *stego-data* harus dapat diambil kembali untuk digunakan lebih lanjut.

Metode LSB (*Least Significant Bit*) merupakan metode steganografi yang paling sederhana dan paling mudah diimplementasikan. Untuk menjelaskan metode ini menggunakan citra digital sebagai *covertext*. Setiap *pixel* di dalam citra berukuran 1 sampai 3 *byte*. Pada

susunan bit di dalam sebuah *byte* (1 *byte* = 8 bit), ada bit yang paling berarti (*most significant bit atau MSB*) dan bit yang paling kurang berarti (*least significant bit atau LSB*). Misalnya pada *byte* 11010010, bit 1 yang pertama yang di garis bawahi adalah bit MSB dan bit 0 yang terakhir di garis bawahi adalah bit LSB. Bit yang cocok untukk diganti dengann bit pesan adalah bit LSB, sebab modifikasi hanya mengubah nilai *byte* tersebut satu lebih tinggi atau satu lebih rendah dari nilai sebelumnya.

2.2. Kriptografi

Kriptografi berasal dari bahasa yunani, menurut bahasa dibagi menjadi dua *kripto* dan *graphia*, *kripto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Menurut teminologinya kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ketempat yang lain [1].

Encryption adalah mentransformasi data kedalam bentuk yang tidak dapat terbaca tanpa sebuah kunci tertentu. Tujuannya adalah untuk meyajinkan privasi dengan menyembunyukan informasi dari orang-orang yang tidak ditujukan, bahkan mereka yang memiliki akses ke data terenkripsi. Dekripsi merupakan kebalikan dari enkripsi, yaitu transformasi data terenkripsi kembali ke bentuknya semula.

2.3. Advanced Encryption Standard (AES)

Advanced Encryption Standart (AES) dipublikasikan oleh NIST (National Institute of Standard and Tecnology) pada tahun 2001. *Advanced Encryption Standart* merupakan simetris block cipher untuk menggantikan DES (*Data Encryption Standard*).

Algoritma AES merupakan algoritma simetris yaitu menggunakan kunci yang sama untuk proses enkrripsi dan dekripsi. Algoritma AES memiliki tiga pilihan kunci yaitu tipe: AES-128, AES-192, dan AES-256. Masing-masing tipe menggunakan kunci internal yang berbeda yaitu *round key* untuk setiap proses putaran.

Proses putaran enkripsi AES-128 dikerjakan sebanyak 10 kali (a=10), yaitu sebagai berikut :

a. Addroundkey
b. Putaran sebanyak a-1 kali, proses yang dilakukan pada setiap putaran adalah: *SubBytes*, *ShiftRows*, *MixColumns*, dan *AddRoundKey*.
c. Final round, adalah proses untuk putaran terakhir yang meliputi SubBytes, ShiftRows, dan AddRoundKey.

3. HASIL DAN PEMBAHASAN

Plainteks Dokumen	: KEAMANANKOMPUTER
Nilai HEX	: 4B 45 41 4D 41 4E 41 4E 4B 4F 4D 50 55 54 45 52
Kunci	: MODESTY_MANURUNG
Nilai HEX	: 4D 4F 44 45 53 54 59 5F 4D 41 4E 55 52 55 4E 47

3.1. Add Round Key

Add Round Key pada dasarnya adalah mengkombinasikan chiperteks yang sudah ada dengan cipher key dengan hubungan XOR. XOR dilakukan per kolom yaitu kolom pertama chiperteks di XOR dengan kolom pertama RoundKey dan seterusnya.

State					RoundKey			
4B	41	4B	55		4D	53	4D	52
45	4E	4F	54		4F	54	41	55
41	41	4D	45	XOR	44	59	4E	4E
4D	4E	50	52		45	5F	55	47

Sehingga dihasilkan : { 06 ; 0A ; 05 ; 08 ; 12 ; 1A ;18 ; 11 ; 06 ; 0E ; 03 ; 05 ; 07 ; 01 ; 0B ; 15 }

3.2. Sub Bytes S-Box

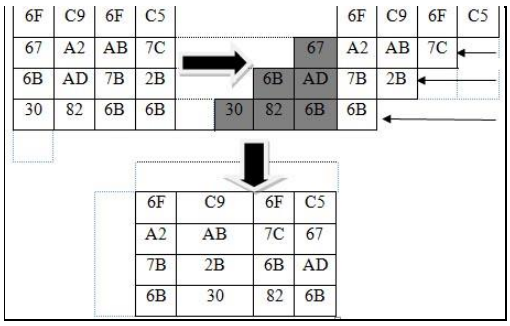
Prinsip dari *Sub Bytes* adalah menukar isi matriks/table yang ada dengan matriks/table lain yang disebut dengan S-Box. Di bawah ini adalah Tabel Sub Bytes S-Box.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7c	77	7b	F2	6b	6f	C5	30	01	67	2b	Fe	D7	Ab	76
1	Ca	82	C9	7d	Fa	59	47	F0	Ad	D4	A2	Af	9c	A4	72	C0
2	B7	Fd	93	26	36	3f	F7	Cc	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9a	07	12	80	E2	Eeb	27	B2	75
4	09	83	2c	1a	1b	6e	5a	A0	52	3b	D6	B3	29	E3	2f	84
5	53	D1	00	Ed	20	Fc	B1	5b	6a	Cb	Be	39	4a	4c	58	Cf
6	D0	Ef	Aa	Fb	43	4d	33	85	45	F9	02	7f	50	3c	9f	A8
7	51	A3	40	8f	92	9d	38	F5	Bc	B6	Da	21	10	Ff	F3	D2
8	Cd	0c	13	Ec	5f	97	44	17	C4	A7	7e	3d	64	5d	19	73
9	60	81	4f	Dc	22	2a	90	88	46	Ee	B8	14	De	5e	0b	Db
A	E0	32	3a	0a	49	06	24	5c	C2	D3	Ac	62	91	95	E4	79
B	E7	C8	37	6d	8d	D5	4e	A9	6c	56	F4	Ea	65	77a	Ae	08
C	Ba	78	25	2e	1c	A6	B4	C6	E8	Dd	74	1f	4b	Bd	8b	8a
D	70	3e	B5	66	48	03	F6	0e	61	35	57	B9	86	C1	1d	9e
E	E1	F8	98	11	69	D9	8e	94	9b	1e	87	E9	Ce	55	28	Df
F	8c	A1	89	0d	Bf	E6	42	68	41	99	2d	0f	B0	54	Bb	16

Setelah melakukan SubBytes maka hasilnya adalah :
{6F; C9; 6F; C5; 67; A2; AB; 7C; 6B; AD; 7B; 2B; 30; 82; 6B; 6B}

3.3. Shift Rows

Shift Rows seperti namanya adalah sebuah proses yang melakukan shift atau pergeseran pada setiap elemen blok/table yang dilakukan per barisnya. Yaitu baris ketiga dilakukan pergeseran 2 byte, dan baris keempat dilakukan pergeseran 3 byte. Pergeseran tersebut terlihat dalam blok adalah sebuah pergeseran tiap elemen ke kiri tergantung berapa byte tergesernya, tiap pergeseran 1 byte berarti bergeser ke kiri sebanyak satu kali.



3.4. Mix Columns

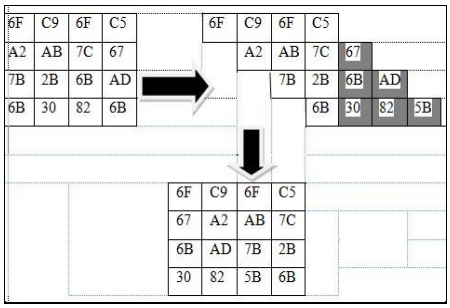
Mix Column adalah mengalikan tiap elemen dari blok chipher dengan matriks. Table sudah ditentukan dan siap pakai. Pengalian dilakukan seperti perkalian matriks biasa lalu perkalian keduannya dimasukkan ke dalam blok cipher baru.

02	01	01	03		6F	C9	6F	C5		DC	04	3A	C3
03	02	01	01		A2	AB	7C	67		05	74	1E	3B
01	03	02	01	*	7B	2B	6B	AD	=	CF	A4	CB	A5
01	01	02	03		6B	30	82	6B		13	F6	EF	53

3.5. Proses Dekripsi

3.5.1 InvShiftRows

InvShiftRows adalah transformasi *byte* yang berkebalikan dengan transformasi ShiftRows. Pada transformasi InvShiftRows, dilakukan penggeseran bit ke kanan sedangkan pada ShiftRows dilakuukan penggeseran bit ke kiri.



3.5.2. InvSubBytes

InvSubBytes juga merupakan transformasi bytes yang berkebalikan dengan transformasi SubBytes. Pada InvSubBytes, tiap elemen pada state dipetakan dengan menggunakan tabel Inverse S-Box.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	9	6a	D5	30	36	A5	38	Bf	40	A3	9E	81	F3	D7	FB
1	7c	E3	39	82	9b	2f	Ff	87	34	8e	43	44	C4	DE	E9	CB
2	54	7b	94	32	A6	C2	23	3d	Ee	4c	95	0B	42	FA	C3	4E
3	08	2e	A1	66	28	D9	24	B2	76	5b	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6c	70	48	50	Fd	Ed	B9	Da	5e	15	46	57	A7	8D	9D	84
6	90	D8	Ab	00	8c	Bc	D3	0a	F7	E4	58	05	B8	B3	45	06
7	D0	2c	1e	8f	Ca	3f	0f	02	C1	Af	BD	03	01	13	8A	6B

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
8	3a	91	11	41	4f	67	Dc	Ea	97	F2	CF	CE	F0	B4	E6	73
9	96	Ac	74	22	E7	Ad	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1a	71	1d	29	C5	89	6f	B7	62	0E	AA	18	BE	1B
B	Fc	56	3e	4b	C6	D2	79	20	9a	DB	C0	FE	78	CD	5A	F4
C	1f	Dd	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7f	A9	19	B5	4a	0d	2d	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3b	4d	Ae	2a	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2b	4	7e	ba	77	D6	26	E1	69	14	63	55	21	0C	7D

3.5.3 InvAddRows

Transformasi *InvAddRows* sama dengan transformasi AddRows yaitu menggunakan operasi XOR. Akan dilakukan proses XOR antara *chipertext* dengan kunci round yang digunakan pada saat enkripsi.

06	12	06	07		4D	53	4D	52
0A	1A	0E	01		4F	54	41	55
05	18	03	0B	XOR	44	59	4E	4E
08	11	57	05		45	5F	55	47

Sehingga kembali ke bentuk aslinya atau *plaintext* { 4B; 45; 41; 4D; 41; 4E; 41; 4E; 4B; 4F; 4B; 50; 55; 54; 45; 52

Langkah-langkah untuk menyisipkan karakter ciphertext **Ü | Ï !! ┘ t ¤ ö : 30 Ë ï Ã ; ¥** S.

a. Mengubah ciphertext menjadi bilangan biner

Ü = 220 = 11011100 **┘ = 4 = 00000100**
| = 5 = 00000101 **t =116 = 01110100**
Ï = 207 = 11001111 **¤ =164 = 10100100**
!! = 19 = 00010011 **ö =246 = 11110110**

: = 58 = 00111010 **Ã = 195 = 11000011**
30 = 30= 00011110 **; =59 = 00111011**
Ë = 203 = 11001011 **¥ =165 = 10100101**

ï =239 = 11101111 S =83 = 01010011

b. Kemudian mengambil nilai biner dari tiap-tiap pixel gambar yang akan disisipkan. Dari nilai pixel-pixel gambar digunakan untuk menampung panjang karakter, yang berfungsi sebagai batas pengambilan *file* yang telah tersisipi oleh karakter-karakter, dan selanjutnya pixel pertama sampai pixel terakhir digunakan untuk menampung *file* yang telah terenkripsi, kemudian dari nilai-nilai pixel tersebut akan diubah menjadi nilai biner.

01111000	00011100	01111011	01111101	00011011	01111100	10000010	00011010
01111101	11000011	00001101	10001010	10111110	00001110	10001001	10111001
10100101	00010100	10000011	10010110	00010110	100000000	01111111	10000101
10001100	00010101	00010010	00010000	10000111	10110100	10101011	10101010
10010001	00010110	10000001	01111101	00010010	00010000	00010001	10100101
10100000	00010011	10000100	100100001	00010110	10111001	10100101	00010110
10000001	10000110	00010001	01100001	10011110	10101111	01001010	01110110
00001010	11100100	10101101	11100111	10101100	01010101	11110100	11000001
01111000	00011101	01111010	01111100	00011011	01111101	10000010	00011010
01111100	11000011	00001100	10001010	10111110	00001110	10001000	10111001
10100100	00010101	10000010	10010111	00010110	100000000	01111111	10000100
10001100	00010101	00010010	00010000	10000111	10110100	10101010	10101011
10010001	00010110	10000001	01111101	00010010	00010000	00010001	10100101
10100000	00010011	10000100	100100001	00010110	10111001	10100101	00010110
10000001	10000110	00010001	01100001	10011110	10101111	01001010	01110110
00001010	11100100	10101101	11100111	10101100	01010101	11110100	11000001

Kemudian gantikan tiap biner dari teksnya ke dalam akhir biner citra penampung, sehingga akan terlihat seperti pada tabel berikut ini.

01111000	00011100	01111011	01111101	00011011	01111100	10000011	00011011
01111101	11000010	00001101	10001010	10111110	00001110	10001000	10111000
10100101	00010101	10000011	10010111	00010110	100000000	01111111	10000101
10001101	00010101	00010010	00010000	10000111	10110100	10101010	10101010
10010000	00010110	10000000	01111101	00010010	00010000	00010000	10100100
10100000	00010010	10000101	100100000	00010111	10111001	10100101	00010110
10000000	10000110	00010001	01100000	10011110	10101111	01001010	01110111
00001010	11100101	10101101	11100110	10101101	01010101	11110101	11000001
01111000	00011101	01111010	01111101	00011011	01111101	10000010	00011010
01111100	11000011	00001101	10001011	10111111	00001110	10001000	10111000
10100101	00010101	10000010	10010111	00010110	100000000	01111111	10000101
10001101	00010101	00010011	00010001	10000110	10110101	10101011	10101011
10010001	00010111	10000000	01111100	00010010	00010000	00010001	10100101
10100001	00010011	10000100	100100001	00010111	10111001	10100100	00010110
10000001	10000110	00010001	01100000	10011110	10101111	01001010	01110111
00001011	11100101	10101100	11100110	10101101	01010100	11110101	11000000

3.6. Algoritma Enkripsi

Input : Plainteks, Kunci { Teks Asli 16 Bytes, Kunci AES }
Output : Cipherteks { Teks sandi 16 Bytes}
Proses : (Nr,w) Enkripsi Kunci (K) {Nr : Jumlah Ronde, w : larik bytes kunci ronde }
Byte cipher [4, Nb]
Cipherteks = Plainteks
Penambahan kunci (Cipher,w [0…3])
For round = 1 Nr -1
  Subtitusi byte (Cipherteks) { Dokumen menjadi hexa}
  Pengeseran kunci (Cipherteks)
  Percampuran kolom (Cipherteks)
  Penambahan kunci (Cipherteks, w + round * Nb)
End for
  Subtitusi byte (Cipherteks)
  Pengeseran kunci (Cipherteks)
Penambahan kunci (Cipherteks, w + Nr * Nb)

Output :
Cipherteks

3.7. Algoritma Dekripsi

Input : Cipherteks, Kunci {Teks Sandi 16 bytes, Kunci AES}
Output : Plainteks {Teks Asli 16 bytes}
Proses : (Nr,w) Ekspansi Kunci (K) {Nr : Jumlah Ronde, w : Larik bytes kunci ronde }
Byte plainteks [4, Nb]
Plainteks = Cipherteks
Penambahan kunci (plainteks, dw + Nr * Nb)
For round = Nr 1 to 1
Pengeseran kunci (Plainteks)
Subtitusi byte (Plainteks)
Percampuran kolom (Plainteks)
Penambahan kunci (Plainteks, dw + round * Nb)
End for
Pengeseran kunci (Plainteks)
Subtitusi byte (Plainteks)
Penambahan kunci (Plainteks, dw)
Output :
Plainteks

3.8. Algoritma Penyisipan File Dokumen (Embedded)

Input : Masukkan data citra digital dengan file ekstensi bmp.
X = Lebar Citra
Y= Tinggi Citra
Proses : Jika X < 8 maka algoritma selesai (tidak dapat menyisipkan dokumen)
Jika X ≥ 8 maka lanjutkan langkah berikutnya.
Masukkan dokumen
P = panjang karakter dokumen
Jika P > Y maka ulangi
Jika P ≤ Y maka lanjutkan langkah berikutnya.
Untuk nilai i = 1 sampai i = 8,
Untuk nilai j = 1 sampai j = P :
i : = i + 1
j : = j + 1
Output :
Data Citra Gambar

3.9. Algoritma Pengambilan File Dokumen (Extraction)

Input : Masukkan data citra digital dengan file ekstensi bmp.
X = Lebar Citra
Y= Tinggi Citra
Proses : Jika X < 8 maka algoritma selesai (tidak ada dokumen)
Jika X ≥ 8 maka lanjutkan langkah berikutnya.
Dokumen = ‘’
Lakukan untuk nilai j = 1. Sampai j = Y :
Kj = (R(1, j) mod 2) * 2^7 + (R(2, j) mod 2) * 2^6 +
(R(3, j) mod 2) * 2^5 + (R(4, j) mod 2) * 2^4 +
(R(5, j) mod 2) * 2^3 + (R(6, j) mod 2) * 2^2 +
(R(7, j) mod 2) * 2^1 + (R(8, j) mod 2) * 2^0

Jika Kj ≠ 0 maka dokumen = dokumen + CHR (Kj)
Jika Kj = 0 maka dokument tetap
CHR (Kj) = karakter ASCII ke – Kj
Kj = Karakter dalam decimal
Output :
Data Dokumen Asli

4. KESIMPULAN

Setelah melalui proses penyelesaian penelitian ini, maka penulis menarik beberapa kesimpulan sebagai berikut :

a. Teknik yang dilakukan dalam mengamankan *file* terenkripsi yaitu dengan cara menerapkan metode *Least Significan Bit* (LSB) pada citra gambar sehingga *file* terenkripsi tersebut dapat disisipkan ke dalam citra gambar .
b. Untuk proses enkripsi dan dekripsi *Advanced Encryption Standard* (AES), plaintext ditransformasikan secara berulang kali selama beberapa putaran. Banyaknya transformasi putaran (Nr) tergantung dari nilai Nk dan Nb. Nk yaitu panjang kunci dibagi 32, sedangkan Nb yaitu panjang blok dibagi 32 dan untuk proses dekripsi.
c. Mengubah *file* yang terenkripsi ke dalam bilangan biner, kemudian Penyisipan *file* yang terenkripsi ke dalam citra digital dilakukan dengan metode *Least Significant Bit* (LSB) yang akan mengganti bit-bit pesan rahasia pada bit terakhir tiap komponen warna piksel citra. Satu komponen warna citra hanya disisipkan satu bit pesan yang bernilai 0 atau 1 sehingga ukuran citra tidak berubah.
d. Memasukkan metode *Least Significant Bit* (LSB) ke dalam program *visual basic.net* 2008 agar *file* yang terenkripsi dapat disembunyikan kedalam citra gambar.

DAFTAR PUSTAKA

[1] Rinaldi Munir. Kriptografi, Informatika, Bandung, 2006.
[2] Dony Ariyus. Kriptografi Keamanan Data Dan Komunikasi, Graha Ilmu, Yogyakarta, 2006.
[3] Abdul Kadir & Adhi Susanto. Teori Dan Aplikasi Pengolahan Citra, Andi, Yogyakarta, 2013
[4] T.Sutoyo, Edy mulyanto, Dr.Vincent Suhartono, Oky Dwi Nurhayati, MT.Wijanarto, Teori Pengolahan Citra Digital, Andi Yogyakarta ; UDINUS Semarang, 2009
[5] Heri Sismoro, Pengantar Logika Informatika Algoritma Dan Pemograman Komputer, Andi, Yogyakarta, 2005
[6] C.Widyo Hermawan. Visual Basic 2008, Andi, Yogyakarta, 2009
[7] Yuni Sugiarti. Analisa dan Perancangan UML Generate Vb.6, Graha Ilmu, Yogyakarta, 2013
[8] Indrajani. Database Design, PT. Elex Media Komputindo Jakarta, 2015