Jurnal Vocational Teknik Elektronika dan Informatika
http://ejournal.unp.ac.id/index.php/voteknika/index
Vol. 11, No. 1, Maret 2023 **P- ISSN: 2302-3295, E-ISSN : 2716-3989**

Implementasi Steganografi Menggunakan Metode *Least Significant Bit* (Lsb) dalam Pengamanan Informasi pada Citra Digital

Fitri Yanti[1*], Khairi Budayawan[2]
[1]Program Studi Pendidikan Teknik Informatika Fakultas Teknik Universitas Negeri Padang
[2]Departemen Teknik Elektronika Fakultas Teknik Universitas Negeri Padang
[*]Corresponding author e-mail: asteralesfitriyanti@gmail.com

ABSTRAK

Penggunaan teknologi sebagai sarana dalam penyampaian informasi menimbulkan dampak terutama dari segi keamanan dan kerahasiaan informasi. Ketika pengirim pesan hanya menunjukan pesan pada orang tertentu dan tidak ingin jika pesan yang dikirimkannya diketahui pihak lain, untuk terhindar dari kebocoran informasi maka dari itu teknik penyembunyian pesan sangat diperlukan. Salah satu metode yang digunakan dalam penyembunyian informasi yaitu metode *Least Significant Bit* (LSB) pada steganografi. Metode ini merupakan salah satu metode yang sering digunakan, sederhana, cepat dalam proses ekstraksi, serta memiliki daya tampung penyisipan yang cukup besar. Pesan akan disisipkan dengan cara mengganti bit terakhir pada citra penampung dengan bit pesan dengan catatan jumlah bit pada pesan tidak melebihi bit citra penampung. Berdasarkan hasil penelitian citra hasil steganografi jika dibandingkan dengan citra awal sangat sulit dibedakan dengan mata. Sehingga dapat disimpulkan bahwa teknik steganografi LSB yang dikombinasikan dengan *vigenere cipher* dinilai baik untuk penyembunyian pesan.

Kata kunci *: Steganografi, Least Significant Bit* (LSB), *Vigenere Cipher*

ABSTRACT

The use of technology as a means of delivering information has an impact, especially in terms of security and confidentiality of information. When the sender of the message only shows the message to certain people and does not want the message sent to be known to other parties, to avoid information leakage, therefore message hiding techniques are needed. One of the methods used in information hiding is the Least Significant Bit (LSB) method in steganography. This method is one method that is often used, simple, fast in the extraction process, and has a large enough insertion capacity. The message will be inserted by replacing the last bit in the container image with the message bit provided that the number of bits in the message does not exceed the container image bits. Based on the research results, the steganography result image when compared to the initial image is very difficult to distinguish by eye. So it can be concluded that the LSB steganography technique combined with the Vigenere cipher is considered good for message hiding.

Keywords: *Steganografi, Least Significant Bit* (LSB), *Vigenere cipher*

I. PENDAHULUAN

Saat ini sering terjadi serangan dan ancaman dalam penyampaian informasi. Ancaman terhadap keamanan data dapat terjadi ketika informasi yang dikirimkan tidak ditujukan kepada semua orang. Saat ini telah banyak terjadi kejahatan didunia maya, dimana informasi rahasia dengan mudahnya diambil oleh seorang peretas [1]. Untuk mencegah agar tidak terjadi kehilangan dan pemalsuan terhadap data dan informasi maka diperlukan suatu teknik yang dapat menyembunyikan serta menyandikan pesan agar tidak mudah terdeteksi oleh pihak yang tidak berwenang. Salah satu teknik yang sering dan mudah diterapkan untuk menyembunyikan pesan atau informasi rahasia adalah steganografi.

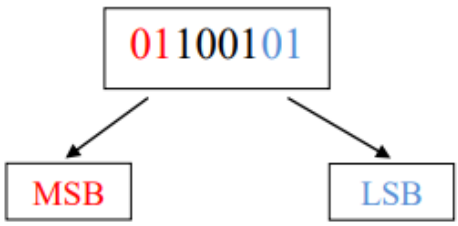
Asal kata steganografi dari bahasa Yunani *stegan* yang bermakna menyembunyikan dan *graphos* yang bermakna tulisan. Steganografi merupakan seni menyembunyikan informasi atau pesan tersembunyi

(*embedded message*) pada suatu wadah penampung (*cover object*) yang dapat berupa teks, gambar, audio, video dan lain-lain. Steganografi merupakan teknik yang bertujuan menyembunyikan pesan rahasia atau tulisan rahasia sehingga informasi rahasia tersebut tidak dapat diidentifikasi oleh orang lain (pihak ketiga) dalam artian yang dapat mengetahui pesan tersebut hanya pengirim dan penerima [2].

Terdapat tiga kriteria steganografi yaitu : *imperceptibility*, artinya tidak dapat dipersepsikan oleh panca indra. *Fidelity*, artinya kualitas file setelah disisipkan tidak jauh berbeda dari file asli. *Recovery*, artinya file dapat dikembalikan ke bentuk semula [3].

Tujuan awal dari teknik steganografi ini bukan untuk mengamankan pesan, yaitu agar orang lain tidak mampu mendeteksi bahwa pada suatu pesan atau informasi tersebut terdapat pesan rahasia didalamnya. Teknik steganografi ini biasanya dibuat dan diimplementasikan melalui media digital.

Banyak metode atau teknik yang digunakan pada Steganografi diantaranya yaitu: *Least Significant Bit* (LSB), *Most Significant Bit* (MSB), *Metode End of File* (EOF), *Spread Spectrum*, *Discrete Cosine Transform* (DCT), *Discrete Wavelet Transform* (DWT) dan *Bit-Plane Complexity Segmentation* (BPCS). Masing-masing metode mempunyai kelebihan dan kekurangan, sehingga tidak ada metode yang sempurna dan penggunaannya tergantung kebutuhan [4]. Teknik steganografi yang sering digunakan yaitu *Least Significant Bit* (LSB).



Gambar 1. Metode LSB

Angka 0 yang berada di depan disebut *Most Significant Bit* (MSB) Maka bit LSB pada biner tersebut yaitu angka 1 yang paling kanan atau paling belakang. Ketika bit paling akhir *LSB* disisipi atau diubah dengan 0 hal tersebut tidak akan mempengaruhi tampilan warna secara jelas (tidak terlihat jelas perbedaannya). Namun jika bit yang disisipi dengan bit yang berbeda maka akan terlihat perbedaan pada citra.

Least Significant Bit (LSB) merupakan metode steganografi yang populer dan sering digunakan. Pada metode ini pesan akan disisipkan dengan cara mengganti bit terkecil (terakhir) dari *pixel* citra dengan bit pesan, karena tidak akan memberikan pengaruh atau perubahan yang signifikan terhadap citra digital [5]. Upaya yang dapat dilakukan yaitu dengan meminimalisir kemungkinan serangan dengan menggabungkan metode LSB dan algoritma kriptografi untuk penyandian pesan. Sehingga hanya yang memiliki kunci yang dapat mengetahui isi dari pesan tersebut.

Kriptografi adalah proses yang digunakan untuk mengubah teks biasa menjadi teks sandi dengan menggunakan kunci simetris yang biasa disebut sebagai enkripsi. Kriptografi bertujuan menyandikan pesan agar hanya orang-orang yang berhak mengaksesnya yang dapat mengetahui maupun menggunakan informasi tersebut. Pengamanan data dilakukan sedemikian rupa oleh pengirim agar orang lain tidak dapat mengenali data tersebut [6]. Kriptografi memanfaatkan algoritma matematika sebagai metode penyandian. Diantara metode yang sering digunakan yaitu *vigenere cipher*.

Vigenere cipher merupakan salah satu algoritma kriptografi klasik. Terdapat dua tahapan yang dilakukan dalam proses kriptografi yaitu *encrypt* (enkripsi) proses mengubah *plaintext* (pesan) menjadi kode acak dengan memanfaatkan algoritma tertentu, dan *decrypt* (dekripsi) proses membaca kode acak menggunakan kunci yang telah dibuat sebelumnya.

Berikut merupakan rumus *vigenere cipher* yang umum digunakan :

$$Ci = (Pi + Ki) \text{ Mod } 26 \quad (1)$$

$$Pi = (Ci - Ki) \text{ Mod } 26 \quad (2)$$

Pi = Pesan atau *plaintext*
Ki = kunci atau *key*
Ci = Teks hasil enkripsi atau *ciphertext*

Pada penelitian ini peneliti mengkombinasikan metode LSB dengan algoritma *vigenere cipher* modifikasi ASCII *code printable character* 32-126, berikut rumus yang digunakan:

$$C_i = ((P_i\text{-}32+K_i) \bmod 95) +32 \quad (3)$$

$$C_i = ((P_i\text{-}32+K_i) \bmod 95) +32 \quad (4)$$

Steganografi umumnya banyak diimplementasikan pada file berupa citra digital. Karena citra digital atau yang biasa disebut gambar adalah format yang paling mudah disebarluaskan melalui web dan komunitas online. Alasan lainnya kenapa format citra digital yang umum digunakan sebagai media penampung, hal ini dikarenakan ukuran dari filenya yang tidak terlalu besar selain itu terdapat banyak algoritma steganografi yang digunakan pada media penampung (*cover*) berformat citra [7]. Namun yang perlu diperhatikan yaitu ketika informasi disisipkan pada citra digital dengan format tertentu kemudian diubah ke format lain maka informasi rahasia yang terdapat pada citra tersebut akan hilang [8].

Tujuan dari penelitian ini yaitu mengetahui apakah metode LSB ini efektif digunakan dalam penyembunyian pesan serta sejauh mana perubahan kualitas pada citra dengan membandingkan citra *cover* dengan citra hasil steganografi. Untuk membandingkan kualitas citra hasil (*stego image*)

P-ISSN: 2302-3295

dengan citra awal (*cover image*) dilakukan perhitungan PSNR. Nilai PSNR direpresentasikan dengan angka dan satuannya adalah dB. Pengujian ini bertujuan untuk menentukan seberapa jauh citra *stego* (yang telah dilakukan penyisipan) berubah dari citra asli [9].

Untuk menghitung nilai PSNR, kita harus mengetahui nilai MSE terlebih dahulu berikut rumus menghitung MSE dan PSNR :

$$MSE = \frac{1}{MN} \sum_{X=1}^{M} \quad \sum_{Y=1}^{N} \quad (S_{xy} - C_{xy})^2 \qquad (5)$$

$$PSNR = 10 \ Log_{10} \frac{255}{MSE} \qquad (6)$$

Keterangan :

X, Y = Koordinat Pixel Citra
M, N = Ukuran Resolusi Citra
S_{xy} = Citra Stego
C_{xy} = Citra Asli

II. METODE PENELITIAN

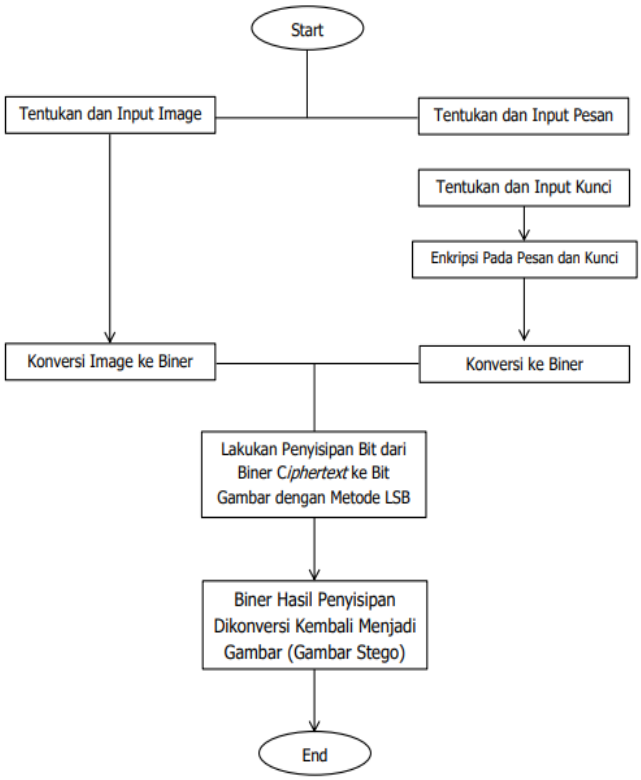
Pada penelitian ini eksperimen dilakukan pada objek berupa citra digital RGB dengan format BMP yang kemudian akan dilakukan penyandian dan penyisipan pesan teks. Implementasi akan dilakukan menggunakan software MATLAB.

Teknik pengumpulan data yang dilakukan yaitu dengan Tinjauan Pustaka (*Library Research)* dan dokumentasi yaitu membaca buku-buku, jurnal yang berkaitan dengan steganografi dan keamanan data selanjutnya dilakukan pengumpulan terhadap data-data tersebut. Terdapat dua proses umum pada steganografi yaitu proses *embedding* untuk menyembunyikan pesan dan ekstraksi untuk mengekstraksi pesan yang disembunyikan [10].

A. *Embedding*

Embedding merupakan proses penyisipan suatu file pada file penampung (*cover*) yang nantinya akan menghasilkan file *stego* (file hasil penyisipan) [11]. File yang disisipkan maupun yang disisipi pesan dapat berupa citra (gambar), audio, video, teks dan lain-lain.

1. Konversi gambar menjadi biner.
2. Memilih dan menentukan pesan teks (*plaintext*).
3. Memilih dan menentukan kunci (*key*).
4. Melakukan enkripsi pada pesan dan kunci menggunakan metode *vigenere cipher*.
5. *Ciphertext* selanjutnya dikonversi menjadi biner.
6. Selanjutnya melakukan penyisipan biner *ciphertext* pada biner gambar dengan metode LSB, setiap bit dari *ciphertext* disisipkan pada bit terakhir dari gambar.
7. Lalu biner dari hasil dari penyisipan tersebut dikonversi kembali menjadi gambar.

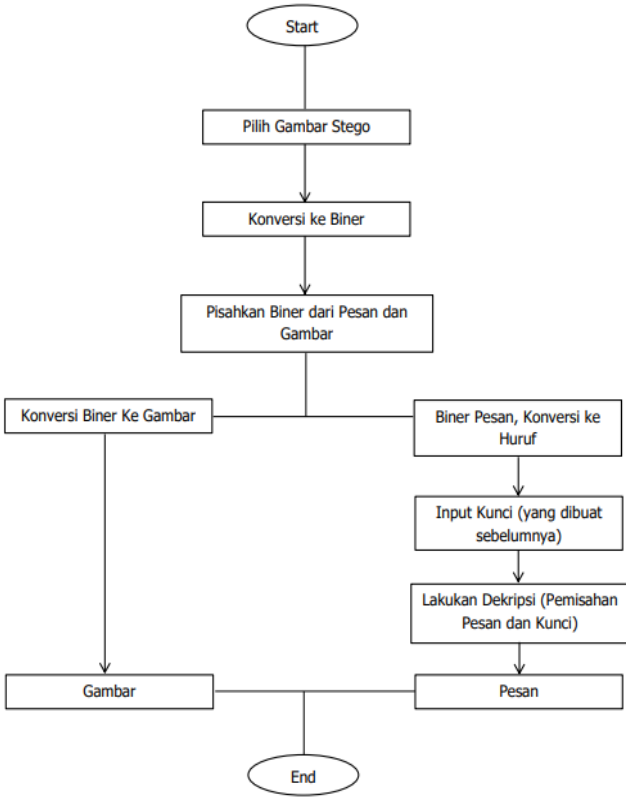


Gambar 2. Proses *embedding*

B. Ekstraksi

Merupakan proses mengembalikan dan memisahkan pesan dari *cover* menjadi bentuk semula [12].

1. Pilih dan tentukan gambar *(stego image).*
2. Lalu gambar stego dikonversi menjadi biner.
3. Melakukan pemisahan biner *ciphertext* dan biner gambar.
4. Melakukan dekripsi (pemisahan pesan dan kunci) dapat dilakukan dengan cara menginputkan kunci yang telah dibuat sebelumnya kemudian diproses dengan perintah dekripsi.
5. Biner gambar diubah menjadi desimal kemudian dipetakan menjadi gambar.
6. Biner *plaintext* dan kunci diubah menjadi bilangan desimal serta dikonversi ke bentuk huruf.



Gambar 3. Proses ekstraksi

III. HASIL DAN PEMBAHASAN

Pada proses implementasi steganografi LSB terdapat dua tahapan yang dilakukan yaitu *embedding* (penyisipan pesan) dan ekstraksi (mengembalikan pesan seperti semula) berikut merupakan proses *embedding* dan ekstraksi citra.

A. *Embedding*

1. Tentukan citra yang dijadikan sebagai penampung *(cover)* Citra penampung yang dijadikan cover image yaitu **mushroom.bmp** dengan dimensi 4x4x3 *size* 102 bytes.

Gambar 4. mushroom.bmp

2. Baca nilai desimal citra *cover*, Resolusi citra yang dijadikan *cover* pesan yaitu citra RGB dengan ukuran 4x4x3, sehingga terdapat 48 nilai desimal, terdapat 16 bit pada masing-masing warna (r*ed*, *green* dan *blue)*. Dapat dilihat sebagai berikut :

Tabel 1. Pixel RGB

Red			
234	168	166	236
126	70	86	141
152	187	233	218
242	196	187	219

Green			
246	223	222	248
196	174	179	202
193	209	238	231
242	209	205	224

Blue			
252	247	246	253
226	219	217	227
207	213	234	218
235	154	134	206

3. Konversi nilai desimal ke biner.

Tabel 2. Konversi desimal ke biner

Red			
11101010	10101000	10100110	11101100
1111110	1000110	1010110	10001101
10011000	10111011	11101001	11011010
11110010	11000100	10111011	11011011
Green			
11110110	11011111	11011110	11111000
11000100	10101110	10110011	11001010
11000001	11010001	11101110	11100111
11110010	11010001	11001101	11100000
Blue			
11111100	11110111	11110110	11111101
11100010	11011011	11011001	11100011
11001111	11010101	11101010	11011010
11101011	10011010	10000110	11001110

4. Tentukan jenis pesan dan kunci, Pesan yang digunakan dalam penelitian ini berupa teks, yang menjadi pesan yang disisipkan atau *plaintext* pada contoh ini yaitu: "**hope"** yang terdiri dari 4 karakter sedangkan yang menjadi kunci yaitu : "**will"** yang terdiri dari 4 karakter.
5. Pesan dan kunci tersebut kemudian digabungkan (dienkripsi) menggunakan algoritma *vigenere cipher* modifikasi. Hasil penggabungan pesan dan kunci disebut *ciphertext.*

Tabel 3. Enkripsi dengan v*igenere cipher*

h	o	p	E
w	i	l	L
104	111	112	101
119	105	108	108
Mod 95			
96	89	93	82
`	Y	]	R

$$C_i = ((P_i\text{-}32+K_i) \bmod 95) +32 \quad (1)$$

$$C_i = ((104\text{-}32+119) \bmod 95) +32$$

$$= 191 \bmod 95$$

$$= 96 \text{ (karakter " ` ")}$$

P-ISSN: 2302-3295

Langkah yang sama berlaku untuk plaintext dan key selanjutnya. Sehingga diperoleh *ciphertext* **“`Y]R”** dari penggabungan pesan dan kunci.

6. *Ciphertext* dikonversi ke dalam biner, dan selanjutnya akan disisipkan pada citra *cover*.

Tabel 4. Biner *ciphertext*

96	89	93	82
01100000	01011001	01011101	01010010

7. Proses penukaran bit (steganografi dengan metode LSB), yaitu menyisipkan pesan pada citra *cover*, dapat dilakukan apabila jumlah biner pesan dapat ditampung pada citra *cover* berdasarkan kriteria perhitungan jumlah pixel dibagi 8 bit.

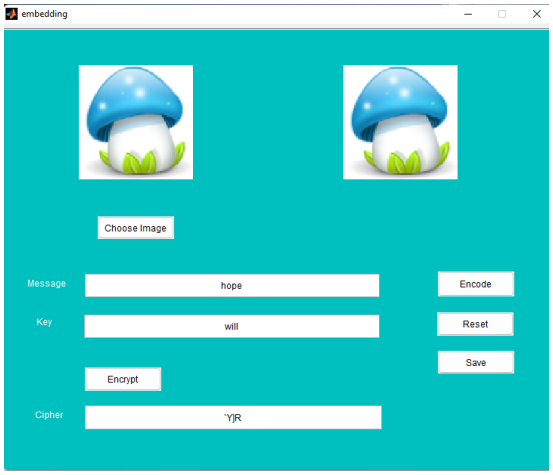
Ciphertext akan disisipkan pada biner *pixel* citra *cover* berdasarkan metode LSB menggunakan perulangan pada baris dan kolom, yaitu masing-masing bit pada *ciphertext* disisipkan secara bergiliran pada *pixel* warna *red*, *green* dan *blue*. Biner terakhir *cover* diganti dengan biner *ciphertext* 0 atau 1. Berikut merupakan hasil penyisipan *ciphertext* pada citra *cover*:

01100000 01011001 01011101 01010010

Tabel 5. Penyisipan biner *ciphertext* pada citra

Red			
1110101**0**	1010100**0**	1010011**0**	1110110**1**
1111111	1000111	1010110	1000110**1**
1001100**0**	1011101**1**	1110100**1**	11011010
11110000	11000100	10111011	11011011
Green			
1111011**1**	1101111**0**	1101111**0**	1111100**0**
1100010**0**	1010111**0**	1011001**1**	1100101**0**
1100000**1**	1101000**0**	1110111**0**	11100111
11110000	11010001	11001101	11100000
Blue			
1111110**1**	1111011**0**	1111011**0**	1111110**1**
1110001**0**	1101101**1**	1101100**1**	1110001**1**
1100111**0**	1101010**0**	11101010	11011010
11101000	10011000	10000110	11001110

8. *Ciphertext* akan disisipkan pada biner *pixel* citra *cover*, berdasarkan metode LSB yaitu masing-masing bit pada *ciphertext* disisipkan pada bit terakhir citra *cover*.



Gambar 5. Embedding pesan

9. Hasil dilakukan penyisipan, hasil dari nilai biner *cover* baru di konversi kembali ke bentuk bilangan desimal dan kemudian dipetakan menjadi citra baru yang disebut *stego image.*

Gambar 6. mushroom_stego.bmp

B. Ekstraksi

Proses ekstraksi merupakan kebalikan dari proses embedding, dimana citra yang berisi pesan akan dipisahkan kembali, yang terdiri atas *cover image* (media penampung), kunci, dan plaintext (pesan).

1. Masukkan/pilih citra yang telah disisipkan pesan teks *(stego image).*

Gambar 7. mushroom_stego.bmp

2. Baca nilai *pixel stego image* selanjutnya konversi ke biner.

Tabel 6. Konversi desimal ke biner

Red
234 168 166 237
127 71 86 141
152 187 233 218
240 196 187 219
Green
247 222 222 248
196 174 179 202
193 208 238 231
240 209 205 224
Blue
253 246 246 253
226 219 217 227
206 212 234 218
232 152 134 206

Tabel 7. *Pixel stego image*

Red			
1110101**0**	1010100**0**	1010011**0**	1110110**1**

1111111	1000111	1010110	10001101
10011000	10111011	11101001	11011010
11110000	11000100	10111011	11011011

Green

11110111	11011110	11011110	11111000
11000100	10101110	10110011	11001010
11000001	11010000	11101110	11100111
11110000	11010001	11001101	11100000

Blue

11111101	11110110	11110110	11111101
11100010	11011011	11011001	11100011
11001111	11010101	11101010	11011010
11101011	10011010	10000110	11001110

3. Kemudian ambil nilai kunci dari 8 bit LSB biner citra awal *stego image* dan dikonversi ke bilangan desimal, nilai kunci dikalikan dengan 8 bit untuk mengambil nilai bit pesan.

Tabel 8. Dekripsi *vigenere cipher*

`	Y	]	R
96	89	93	82
119	105	108	108
104	111	112	101
w	i	l	l
h	o	p	e

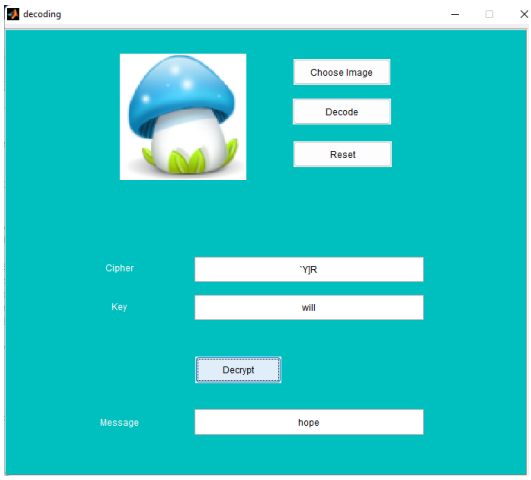
$P_i = ((C_i+32-K_i) \bmod 95) + 32$
$= ((96+32-119) \bmod 95)+32$
$= 9 \bmod 95$
$= 104$ (karakter " **h** ")

Langkah yang sama berlaku untuk plaintext dan *key* selanjutnya. Sehingga diperoleh *ciphertext* **"hope"** dari penggabungan pesan dan kunci.

4. Ambil bit LSB dari setiap elemen pixel dimulai dari bit ke-9 hingga sejumlah perkalian kunci dengan 8 bit lalu tambahkan dengan 8 bit kunci LSB, kemudian kelompokkan nilai bit-bit LSB menjadi 8 kelompok, selanjutnya konversi ke bilangan desimal.

Tabel 9. *Pixel* awal *cover*

Red			
11101010	10101000	10100110	11101100
1111110	1000110	1010110	10001101
10011000	10111011	11101001	11011010
11110010	11000100	10111011	11011011
Green			
11110110	11011111	11011110	11111000
11000100	10101110	10110011	11001010
11000001	11010001	11101110	11100111
11110010	11010001	11001101	11100000
Blue			
11111100	11110111	11110110	11111101
11100010	11011011	11011001	11100011
11001111	11010101	11101010	11011010
11101011	10011010	10000110	11001110



Tabel 10. Ekstraksi Pesan

5. Setelah diperoleh bilangan desimal dari biner pengelompokkan, konversi ke karakter, karakter yang dihasilkan tersebutlah yang menjadi pesan yang telah disembunyikan sebelumnya.

Gambar 8. mushroom.bmp

Tabel 11. Pesan atau *plaintext*

01101000	01101111	01110000	01100101
104	111	112	101
h	o	p	E

Tabel 12. Kunci atau *key*

01110111	01101001	01101100	01101100
119	105	108	108
w	i	l	L

C. Pengujian

Uji coba dilakukan pada 5 buah citra cover dengan format dan ukuran yang berbeda, berikut merupakan hasil uji cobanya:

Tabel 13. Kriteria pengamatan

Kriteria	Keterangan
Baik Sekali	Ketika tidak terdapat perbedaan antara *stego image* dengan citra asli.
Baik	Ketika terdapat perbedaan antara *stego image* dengan citra asli yang dilakukan dengan pengamatan yang teliti.
Jelek	Ketika perbedaan dapat dilihat antara *stego image* dengan citra asli.
Jelek Sekali	Perbedaan antara *stego image* dan citra asli terlihat jelas.

P-ISSN: 2302-3295

Tabel 14. Implementasi pada 5 buah citra digital

No	Nama	Size	Plaintext	Key	Ciphertext	Sebelum Embedding	Setelah Embedding
1.	grapes.bmp	315 KB	Nama : Fitri Yanti	Name	=C[Gn{m,XV`On;OtcKm		
2.	lemon.bmp	199 KB	NIM : 18076022	Student ID	B>Cd`n&8'Y[*%(v		
3.	melon.bmp	477 KB	TTL : Lubuk Gedang, 16 Desember 2000	Birth	?>?t#b6hW"Ni:ZMDXZ!h s r9NVO`WNUi%%xr		
4.	kiwi.bmp	289 KB	Steganografi adalah seni dan ilmu menulis pesan tersembunyi atau menyembunyikan pesan dengan suatu cara sehingga selain si pengirim dan si penerima, tidak ada seorang pun yang mengetahui atau menyadari bahwa ada suatu pesan rahasia.	steganography	GtKOC]_OsCWRyUYGTCWo[XPZh^UceQN\g`G_^f]heXGbQVrVV[mYbH]PhYgTVR^yaZTaG:Rja[ZT[btVMUP^gWG_P[btY]CcseVCqJveZNQPVWL;UVU[]ce[Kn`MalZ[catJPncQrRVW_f^SImndQWC\h[XVe[G`blaIpYobt_IPVoUXPXNnU][QaPdIha^'NhmVJITXoJTJhJyUYGgUdQ\haaNmUceZCWQ[\C		
5.	cherry.bmp	148 KB	Segera kirimkan uang sejumlah 2.000.000.0000,00 ke No. Rek : 8851234567891012	Ancaman	5TKG`CnMXVK[MPPnYC\InUTNW[NPJncs{q~q\sq}o~q~oq}aZGn2Q{aAGZc{my'v ut"v%x'r}r!		

Berdasarkan kriteria pengamatan maka hasil yang diperoleh adalah **Baik**, karena citra sebelum dan setelah dienkripsi dan disisipkan pesan tidak mengalami perubahan jika dilihat secara visual oleh mata.

Tabel 15. Hasil perbandingan ukuran citra

No	Nama	Ukuran Citra *Cover*	Ukuran *Stego Image*
1.	grapes.bmp	315 KB	315 KB
2.	lemon.bmp	199 KB	199 KB
3.	melon.bmp	477 KB	477 KB
4.	kiwi.bmp	289 KB	289 KB
5.	cherry.bmp	148 KB	148 KB

Berdasarkan tabel perbandingan ukuran citra diatas maka dapat disimpulkan bahwa, ukuran citra dengan format awal *.bmp* setelah dilakukan penyisipan tidak mengalami perubahan.

Tabel 16. Nilai PSNR

No	Nama	PSNR *Stego Image*
1.	grapes.bmp	33.5713 dB
2.	lemon.bmp	32.6163 dB
3.	melon.bmp	31.9186 dB
4.	kiwi.bmp	30.7106 dB
5.	cherry.bmp	32.7863 dB

Berdasarkan pengujian PSNR didapatkan bahwa nilai PSNR rata-rata diatas 30 dB sehingga dapat disimpulkan kualitas citra sebelum dilakukan penyisipan tidak jauh berbeda dari citra setelah disisipkan pesan hal ini menunjukkan tidak terjadi perubahan secara signifikan terhadap citra tersebut.

IV. KESIMPULAN

Berdasarkan hasil penelitian, maka dapat disimpulkan sebagai berikut:

1. Hasil dari implementasi penyisipan dan penyembunyian pesan dengan metode steganografi LSB dan algoritma *vigenere cipher* pada citra digital dapat berjalan dengan baik. Citra yang disisipkan pesan (*stego image*) tidak mengalami perubahan yang signifikan dari citra asli.
2. Pesan yang disisipkan pada citra dapat diperoleh kembali seperti semula. Kecuali, jika pesan atau informasi yang disisipkan pada citra di ubah formatnya, dilakukan *cropped*, dan ukuran citra di *resize* maka informasi didalamnya akan rusak.

V. SARAN

Berdasarkan hasil yang diperoleh, terdapat beberapa hal yang penulis sarankan untuk peneliti selanjutnya yaitu:

1. Pada penelitian ini penulis hanya menggunakan citra digital (gambar) sebagai media penampung dan teks sebagai pesan, diharapkan penelitian selanjutnya dapat menggunakan media audio, video dan lain-lain sebagai media penampung maupun pesan yang akan disisipkan.
2. Pada penelitian ini penulis sudah mengkombinasikan metode steganografi LSB dengan algoritma kriptografi *vigenere cipher* menggunakan software Matlab. Diharapkan pada penelitian berikutnya dapat menggunakan aplikasi yang bisa diinstal pada android serta mengkombinasikan metode steganografi dengan algoritma kriptografi yang lebih aman dalam penyandian seperti algoritma dengan 2 kunci seperti algoritma RSA, sehingga lebih baik bagi keamanan pesan atau informasi.

DAFTAR PUSTAKA

[1] Tri Handayani, Tri Yuliati, dan S. Patimah, "Implementasi Steganografi Dengan Metode End Of File (EOF) Untuk Menyisipkan Pesan Teks Pada Gambar," *J. Fasilkom*, vol. 11, no. 3, hal. 143–149, 2021, doi: 10.37859/jf.v11i3.3124.

[2] C. Nugroho, "Steganografi Pada Pengiriman Teks Pesan Gambar dengan Metode Least Significant Bit & Steghide," *J. Ilmu Siber*, vol. 1, no. 3, hal. 44–47, 2022, [Daring]. Tersedia pada: https://jurnal.unsia.ac.id/index.php/jis/article/view/54%0Ahttps://jurnal.unsia.ac.id/index.php/jis/article/download/54/40

[3] N. Nurmaesah *et al.*, "Aplikasi Steganografi Untuk Menyisipkan Pesan DALAM MEDIA IMAGE," *J. TAM (Technology Accept. Model.*, vol. 8, no. 1, hal. 13–17, 2017.

[4] A. P. Ratnasari dan F. A. Dwiyanto, "Metode Steganografi Citra Digital," *Sains, Apl. Komputasi dan Teknol. Inf.*, vol. 2, no. 2, hal. 52, 2020, doi: 10.30872/jsakti.v2i2.3300.

[5] S. Supardi, A. A. Alkodri, dan B. Isnanto, "Teknik Steganografi Penyembunyian Pesan Text Rahasia Pada Citra Digital Dengan Metode Least Significant Bit," *J. Sisfotek Glob.*, vol. 11, no. 1, hal. 70, 2021, doi: 10.38101/sisfotek.v11i1.351.

[6] A. Rohmanu, "METODE ALGORITMA DES DAN METODE END OF FILE Ajar Rohmanu," *J. Inform.*, vol. 2, no. 1, hal. 1–11, 2017.

[7] A. A. Permana *et al.*, "Implementasi Steganografi File Citra Digital Menggunakan Metode Least Significant Bit Implementation of Digital Image File Steganography Using," vol. 11, no. 01, hal. 62–72, 2022.

[8] P. Painem, "Implementasi Steganografi Metode Discrete Cosine Transform (Dct) Dan Kompresi Metode Huffman Untuk Mengamankan Dokumen Surat Keputusan Pada Yblc," *Telemat. Mkom*, vol. 8, no. 2, hal. 121–126, 2016.

[9] T. E. Putri, M. R. Al Fauzan, dan P. A. Sejati, "Perbaikan Algoritma Steganografi Teknik Least Significant Bits Untuk Aplikasi Keamanan Data," *J. Online Phys.*, vol. 3, no. 1, hal. 27–32, 2018, doi: 10.22437/jop.v3i1.5343.

[10] M. F. Syawal, D. C. Fikriansyah, dan N. Agani, "Implementasi Teknik Steganografi Menggunakan Algoritma Vigenere Cipher Dan Metode LSB," *J. TICOM*, vol. 4, no. 3, hal. 91–99, 2016.

[11] N. F. Hasan, C. N. Dengen, dan D. Ariyus, "Analisis Histogram Steganografi Least Significant Bit Pada Citra Grayscale," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 11, no. 1, hal. 20–29, 2020, doi: 10.31849/digitalzone.v11i1.3413.

[12] B. W. Rauf, "Kombinasi Steganografi Bit Matching dan Kriptografi Playfair Cipher, Hill Cipher dan Blowfish," *J. Teknol. Inf.*, vol. 4, no. 2, hal. 228–233, 2020, doi: 10.36294/jurti.v4i2.1346.

P-ISSN: 2302-3295