ISBN: 978-602-60280-1-3

Steganografi Gambar dengan Modifikasi Vigenere Chiper Huruf Arab (Hijaiyah) dan Teknik *Least Significant Bit* (LSB)

1st Khairunnisak
Teknik Informatika
STMIK AMIKOM Purwokerto
Purwokerto, Indoensia
knurisnaini@gmail.com

2nd Alvi
Teknik Informatika
STMIK AMIKOM Purwokerto
Purwokerto, Indonesia
alvi.sholikhatin@gmail.com

3rd Dony
Teknik Informatika
Universitas AMIKOM Yogyakarta
Yogyakarta, Indonesia
dony.a@amikom.ac.id

Abstrak—Salah satu dampak buruk era digital dengan kemudahan aksesnya adalah ancaman keamanan informasi. Oleh karena itu, upaya untuk menjaga keamanan informasi harus dijadikan prioritas utama dalam rangka pencegahan akses oleh pihak tidak bertanggung jawab. Ada beberapa metode yang dapat digunakan antara lain watermarking, steganografi, kriptografi dan digital sign. Pada beberapa kasus, kombinasi antara kriptografi dan steganografi bisa digunakan untuk memberikan perlindungan berlapis dan menjaga kerahasiaan lebih maksimal. Pada paper ini, kombinasi antara Least Significant Bit (LSB) dan algoritma Vigenere modifikasi huruf hijaiyah. Modifikasi ini dilakukan untuk meningkatkan keamanan informasi yang disembunyikan pada sebuah gambar. Tujuan utama paper ini yaitu untuk menciptakan modifikasi baru yaitu antara algoritma Vigenere dengan LSB serta diaplikasikan ke dalam steganografi gambar. Dibangun pula aplikasi encoding untuk menyembunyikan pesan ke dalam gambar. Hasil menunjukkan bahwa kombinasi algoritma Vigenere dengan LSB mampu meningkatkan keamanan pesan tanpa mengubah kualitas gambar.

Keywords— cryptography, steganography, LSB, Vigenère, Arabic character

I. PENDAHULUAN

Perkembangan teknologi di zaman globalisasi saat ini berdampak positif dan berdampak negatif di segala lini kehidupan manusia. Dampak positifnya adalah mudahnya akses berkomunikasi dengan beragam perangkat yang tersedia bahkan tanpa batas. Akibat dari fenomena tersebut dapat memicu dampak yang tidak diinginkan seperti penyalahgunaan keamanan data. Penyalahgunaan tersebut menyebabkan hacker dan cracker membuat khawatir para penggunanya sehingga komunikasi menjadi terhambat [1].

Penggunaan keamanan informasi ini ditujukan agar tidak dicuri oleh orang asing yang tidak berkepentingan dalam bertukar informasi [2]. Pengamanan pesan dapat dilakukan dengan watermarking, steganografi, kriptografi, dan tanda tangan digital. Dalam usaha melindungi sebuah hak cipta gambar digital dapat dilakukan dengan menyisipkan pesan, dimana pesan dari orang yang membuat gambar digital [3]. Salah satu cara untuk mengamankan bentuk gambar digital adalah dengan teknik steganografi. Steganografi adalah salah satu metode yang digunakan untuk memasukkan atau menyembunyikan informasi ke dalam sebuah media [4]. Teknik steganografi memiliki kelebihan yaitu pesan-pesan yang di sembunyikan tidak menarik perhatian [5].

Kriptografi adalah salah satu teknik dalam keamanan data. Kriptografi adalah seni dan ilmu untuk melindungi data dengan merubah ke dalam kode spesifik dan hanya ornag yang memilki kunci yang dapat merubahnya ke bentuk normal[6]. Kriptografi dapat diklasifikasikan menjadi dua yaitu *Shared Key Criptography and Public Key Criptography*[7]. Seni penyandian pesan dengan kriptografi yaitu dengan mengacak sebuah informasi atau pesan tersebut sedemikian rupa sehingga arti pesan tersebut tidak dapat dimengerti oleh orang lain [2].

Beberapa kondisi menggabungkan teknik steganografi dan teknik kriptografi dengan tujuan menjamin rahasia pesan yang akan disampaikan kepada pemiliknya. Salah satunya adalah penggabungan teknik *Least Significant Bit* (LSB) dan algoritma vigenere. Algoritma vigenere adalah suatu algoritma yang dirancang untuk memperbaiki kelemahan dari algoritma subtitusi tunggal, yaitu setiap karakter pada *plaintext* disubtitusikan dengan karakter yang sama. [8]. Sedangkan, *Least Significant Bit* adalah bagian yang mempunyai nilai terkecil dari sebuah susunan bit dan berada pada sisi paling kanan [9]. Jumah bit yang umum digunakan adalah 8 bit dan 32 bit. Pada penelitian ini, jumlah bit yang digunakan sebanyak 8 digit.

Dibandingkan dengan penelitian [13] yang juga menggunakan LSB dengan konversi tabel ASCII dan bilangan biner, perbedaan mendasar terletak pada penggunaan algoritma Vigenere modifikasi Huruf Hijaiyah. Penyisipan pesan teks ke dalam citra gambar menggunakan metode yang serupa yaitu dengan bantuan aplikasi *image encoder*. Kemudian pada penelitian [14] menggunakan algoritma Advanced Encryption Standard (AES) tanpa menggabungkan dengan metode lain, misalnya steganografi, sedangkan proses enkripsi juga menggunakan aplikasi *encoder*.

Penelitian lain [3] menunjukkan bahwa dengan adanya metode steganografi pengiriman data tidak hanya dapat menaikkan tingkat keamanannya, namun dapat meningkatkan tingkat keamanan untuk melindungi sebuah hak cipta dari sebuah gambar. Kasus lain penggabungan dua teknik tersebut dibuktikan dengan pengujian visual berupa tidak adanya perbedaan terlalu mencolok antara citra asli dan citra hasil[1]. Teknik Least Significant Bit dapat menyisipkan pesan pada sebuah gambar dan merupakan sebuah langkah yang sama dilakukan pada algoritma vigenere dalam menyisipkan pesan menjadi sebuah chiper text [4]. Penelitian yang dilakukan oleh [8] berfokus pada kekuatan maupun kelemahan sebuah citra ketika dilakukan dengan beberapa faktor pengujian bukan pada modifikasi atau kebaruan pada algoritma yang digunakan. Penelitian yang dilakukan [10] memberi bukti bahwa penggabungan teknik Least Significant Bit dengan Algoritma Vigenere

ISBN: 978-602-60280-1-3

berhasil diimplementasikan ke dalam sebuah aplikasi andorid dengan berpedoman pada tabel ASCII 7 bit sedangkan penelitian yang akan dilakukan adalah impelementasi dari modifikasi algoritma vigenere dengan teknik *Least Significant Bit* yang berpedoman pada tabel ASCII 8 bit.

Namun, kedua kombinasi teknik tersebut pada penelitian selanjutnya tidak dapat menjamin 100% bahwa pesan dapat disembunyikan dan lolos oleh hacker ataupun seorang cyrptoanalys. Oleh karena itu, adanya modifikasi dari tabel subtitusi vigenere yang diisi dengan huruf hijaiyah (arabic character) diharapkan mampu meningkatkan keamanan sebuah pesan yang disembunyikan pada sebuah gambar. Penelitian ini bermaksud untuk membuat sebuah modifikasi baru terhadap algoritma vigenere yang dikombinasikan dengan teknik *Least Significant Bit* pada sebuah steganografi gambar yang diterjemahkan dalam bentuk aplikasi.

II. METODE PENELITIAN

Langkah-langkah dalam penelitian dapat dilihat pada gambar

Fig. 1. Alur Penelitian

Penelitian ini mengkombinasikan teknik kriptografi dengan modifikasi algoritma vigenere dan teknik steganografi gambar menggunakan LSB. Kedua teknik tersebut diimplementasikan ke dalam aplikasi steganografi gambar sederhana berbasis Java.

A. *Identifikasi Masalah*

Identifikasi masalah pada penelitian ini dilakukan dengan cara mengaanalisis kekurangan-kekurangan pada penelitian sejenis terutama pada modifikasi algoritma kriptografi dan mengkombinasikannya dengan teknik steganografi.

B. *Pencarian ide dan solusi*

Pencarian ide dan solusi dilakukan dengan cara menganlisis dan mengkaji kekurangan-kekurangan penelitian sejenis. Fokus pencarian ide adalah modifikasi salah satu algoritma dalam teknik kriptografi yang dikombinasikan dengan teknik steganografi.

C. *Penerapan ide*

Penerapan ide dilakukan dengan melihat beberapa acuan antara lain, acuan tabel vigenere huruf sebagai dasar acuan dalam membuat modifikasi algoritma vigenere chiper dan tabel ASCII 8 bit sebagai dasar dalam mengetahui konversi huruf ke dalam bilangan biner untuk mempermudah dalam mengimplementasikan teknik *Least Significant Bit.*

Vigenère Chiper adalah salah satu teknik kriptografi yang diperkenalkan pertama kali oleh Giovan Batista belaso di bukunya yang berjudul "La Cifra del. Sig.Giovan Batista belaso (1553)", teknik penyandian ini yaitu dengan menggunakan tabel Vigenère dan angka sebagai substitusi huruf alfabet untuk menghasilkan *chipertext* [4]. Vigenere chiper adalah sebuah teknik kriptografi klasik yang lebih aman dari pada Caesar Chiper [6]. Huruf Hijaiyah adalah huruf dalam bahasa Arab yang berjumlah 29 huruf, termasuk huruf alif. Teknik Vigenère Chiper dengan huruf hijaiyah dan substitusinya disesuaikan dengan pelafalannya dalam huruf alfabet.

Metode steganografi paling sederhana untuk menyisipkan pesan melalui media gambar adalah dengan *Least Significant Bit* (LSB) [3]. Metode ini menggunakan bit terakhir untuk dimodifikasi sehingga tidak menghasilkan perubahan yang signifikan pada gambar. Proses menyisipkan informasi berupa bit ke dalam bit terakhir (8 bit) seperti contoh:

Pixel:
(10101111 11101001 10101000)
(10100111 01011000 11101001)
(11011000 10000111 01011001)
Pesan rahasia:
01000001
Hasil:
(10101110 11101001 10101000)
(10100110 01011000 11101000)
(11011000 10000110 01011001)

Pendekatan *LSB embedding* bisa digunakan untuk berbagai tujuan pada keamanan multimedia, dan bisa diaplikasikan ke dalam berbagai format tipe data sehingga LSB merupakan metode steganografi yang banyak dipakai hingga sekarang [11].

D. *Penerapan gabungan teknik kriptografi dan steganografi*

Teknik kriptografi dan teknik steganografi yang akan diterapkan adalah sebuah modifikasi algoritma vigenee ke dalam huruf arab (hijaiyah) yang dikombinasikan dengan teknik *Least Significant Bit* ke dalam sebuah citra digital.

E. *Penerapan keseluruhan ide ke dalam aplikasi*

Gabungan teknik kriptografi dan teknik steganografi diimplementasikan ke dalam aplikasi berbasis Java. Java merupakan salah bahasa pemrograman yang berorientasi objek, terdistribusi dan dinamis, yang mempunyai platform independen dan bisa berjalan pada sistem operasi apapun [12]. Tingkat portabilitas Java tidak hanya sebatas pada *source code*, tapi juga pada tingkat kode biner (*bytecode*). Sehingga ketika sebuah program berbasis Java telah

ISBN: 978-602-60280-1-3

dikompilasi pada komputer dengan sistem operasi Windows, ia akan tetap dapat berjalan pada komputer Macintosh tanpa perlu dikompilasi ulang (Ardhyana dan Juarna). Java Development Kit (JDK) adalah produk dari Sun Microsystem yang ditujukan untuk pengembangan java.

F. Membuat kesimpulan

Kesimpulan dibuat sebagai penegasan atas proses pengerjaan dan hasil yang diperoleh.

III. HASIL DAN PEMBAHASAN

Pada aplikasi yang dibuat menggunakan prinsip algoritma vigenere yang dimodifikasi menggunakan abjad pada huruf hijaiyah kemudian dilanjutkan dengan steganografi gambar menggunakan teknik LSB. Langkah-langkah yang harus dilakukan antara lain

A. Mengetahui tabel konversi huruf alfabet ke dalam huruf hijaiyah. Tabel konversi huruf tersebut terdapat pada gambar 2 dan 3.

Angka	:	0	1	2	3	4	5	6	7	8	9	10	11	12
Huruf alfabet	:	A	B	C	D	E	F	G	H	I	J	K	L	M
Huruf Hijaiyah	:	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م

Fig. 2. Tabel Konversi huruf A hingga M

Angka	:	13	14	15	16	17	18	19	20	21	22	23	24	25
Huruf alfabet	:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Huruf Hijaiyah	:	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز

Fig. 3. Tabel Konversi huruf N hingga X

Source code pada java, inisiasi konversi huruf abjad alfabet ke dalam huruf hijaiyah (*arabic character*) terdapat pada gambar 4.

```
 *
 * @author Nisak
 */
public class VigenereChipper {

    char mapPesanCr[] = {' ','\n','A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', '
    char mapHasilCr[] = {' ','\n','ا', 'ب', 'ث', 'د', 'ع', 'ف', 'غ', 'ه', 'ح',
```

Fig. 4. Konversi Huruf Abjad Alfabet Ke Dalam Huruf Hijaiyah

Sedangkan *source code* pada java, inisiasi huruf abjad alfabet dan tanda baca yang dapat muncul di dalam program dapat dilihat pada gambar 5.

```
 */
public class Vigene {

    private static final char[] alphnum = {'A', 'B', 'C', 'D', 'E', 'F', 'G',
        'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V',
        'X', 'Y', 'Z', '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '@',
        '#', '-', '+', '!', '$', '%', '&', '*', '(', ')', '_', '+', '=', '.',
        ';', ':', '"', '\'', '}', '{', '\\', '/', '^', '<', '>', '[', ']', '^
        '|', '`', '~'};
```

Fig. 5. Huruf Abjad Alfabet Dan Tanda Baca

B. Langkah kedua adalah mengetahui tabel vigenere pada huruf-huruf hijaiyah yang terdapat pada gambar 6 dan 7

	A	B	C	D	E	F	G	H	I	J	K	L
A	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل
B	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م
C	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م	ن
D	د	ع	ف	غ	ه	ح	ج	ك	ل	م	ن	ط
E	ع	ف	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ
F	ف	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق
G	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر
H	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر	س
I	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر	س	ت
J	ج	ك	ل	م	ن	ط	ظ	ق	ر	س	ت	ص
K	ك	ل	م	ن	ط	ظ	ق	ر	س	ت	ص	ض
L	ل	م	ن	ط	ظ	ق	ر	س	ت	ص	ض	و
M	م	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش
N	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي
O	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز
P	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز	ا
Q	ق	ر	س	ت	ص	ض	و	ش	ي	ز	ا	ب
R	ر	س	ت	ص	ض	و	ش	ي	ز	ا	ب	ث
S	س	ت	ص	ض	و	ش	ي	ز	ا	ب	ث	د
T	ت	ص	ض	و	ش	ي	ز	ا	ب	ث	د	ع
U	ص	ض	و	ش	ي	ز	ا	ب	ث	د	ع	ف
V	ض	و	ش	ي	ز	ا	ب	ث	د	ع	ف	غ
W	و	ش	ي	ز	ا	ب	ث	د	ع	ف	غ	ه
X	ش	ي	ز	ا	ب	ث	د	ع	ف	غ	ه	ح
Y	ي	ز	ا	ب	ث	د	ع	ف	غ	ه	ح	ج
Z	ز	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك

Fig. 6. Tabel Vigenere Huruf Hijaiyah A hingga L

	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	م	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز
B	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز	ا
C	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز	ا	ب
D	ظ	ق	ر	س	ت	ص	ض	و	ش	ي	ز	ا	ب	ث
E	ق	ر	س	ت	ص	ض	و	ش	ي	ز	ا	ب	ث	د
F	ر	س	ت	ص	ض	و	ش	ي	ز	ا	ب	ث	د	ع
G	س	ت	ص	ض	و	ش	ي	ز	ا	ب	ث	د	ع	ف
H	ت	ص	ض	و	ش	ي	ز	ا	ب	ث	د	ع	ف	غ
I	ص	ض	و	ش	ي	ز	ا	ب	ث	د	ع	ف	غ	ه
J	ض	و	ش	ي	ز	ا	ب	ث	د	ع	ف	غ	ه	ح
K	و	ش	ي	ز	ا	ب	ث	د	ع	ف	غ	ه	ح	ج
L	ش	ي	ز	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك
M	ي	ز	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل
N	ز	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م
O	ا	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م	ن
P	ب	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م	ن	ط
Q	ث	د	ع	ف	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ
R	د	ع	ف	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق
S	ع	ف	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر
T	ف	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر	س
U	غ	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر	س	ت
V	ه	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر	س	ت	ص
W	ح	ج	ك	ل	م	ن	ط	ظ	ق	ر	س	ت	ص	ض
X	ج	ك	ل	م	ن	ط	ظ	ق	ر	س	ت	ص	ض	و
Y	ك	ل	م	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش
Z	ل	م	ن	ط	ظ	ق	ر	س	ت	ص	ض	و	ش	ي

Fig. 7. Tabel Vigenere Huruf Hijaiyah M hingga Z

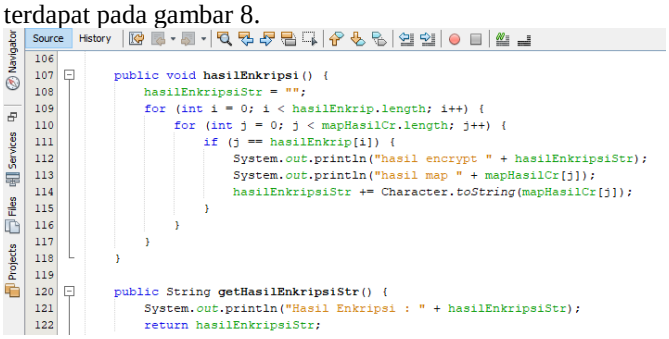
C. Menentukan plain text dan key untuk menghasilkan sebuah chipper text. Contoh tersebut terdapat pada tabel 1

TABLE I. CONTOH *PLAIN TEXT*, *KEY*, DAN *CHIPER TEXT*

Plain text	:	KHAIRUNNISAK
Key	:	NISAK
Chipper text	:	ش ظ س لا ب ه ض ف لا ث ن س

ISBN: 978-602-60280-1-3

Source code enkripsi pada Java dari contoh tabel I terdapat pada gambar 8.



```
106
107     public void hasilEnkripsi() {
108         hasilEnkripsiStr = "";
109         for (int i = 0; i < hasilEnkrip.length; i++) {
110             for (int j = 0; j < mapHasilCr.length; j++) {
111                 if (j == hasilEnkrip[i]) {
112                     System.out.println("hasil encrypt " + hasilEnkripsiStr);
113                     System.out.println("hasil map " + mapHasilCr[j]);
114                     hasilEnkripsiStr += Character.toString(mapHasilCr[j]);
115                 }
116             }
117         }
118     }
119
120     public String getHasilEnkripsiStr() {
121         System.out.println("Hasil Enkripsi : " + hasilEnkripsiStr);
122         return hasilEnkripsiStr;
```

Fig. 8. *Source Code* Proses Enkripsi

D. Langkah selanjutnya yaitu hasil Chipper text di ubah kembali ke dalam huruf abjad alfabet menggunakan tabel konversi pada gambar 1. Selain itu membutuhkan bantuan tabel ASCII seperti pada tabel II. Pembacaan hasil dari kanan ke kiri karena merupakan aturan penggunaan huruf hijaiyah namun pembacaan dalam bentuk abjad alfabet tetap dari sisi sebelah kiri. Maka hasilnya dapat dilihat pada tabel III.

TABLE II. TABEL ASCII 8 BIT

Decimal	*Binary*	*Value*
65	01000001	A
66	01000010	B
67	01000011	C
68	01000100	D
69	01000101	E
70	01000110	F
71	01000111	G
72	01001000	H
73	01001001	I
74	01001010	J
75	01001011	K
76	01001101	L
77	01001101	M
78	01001110	N
79	01001111	O
80	01010000	P
81	01010001	Q
82	01010010	R

TABLE III. TABEL ASCII 8 BIT (LANJUTAN)

Decimal	*Binary*	*Value*
83	01010011	S
84	01010100	T
85	01010101	U
86	01010110	V
87	01010111	W
88	01011000	X
89	01011001	Y
90	01011010	Z

TABLE IV. HASIL PEMBACAAN CHIPER TEXT DAN DIKONVERSI KE DALAM BIT

Chipper Text	:	ش ظ س لا ب ه ض ف لا ث ن س
Huruf Alfabet	:	X-P-S
		L-B-H
		V-F-L
		C-N-S
Binary Text	:	01011000-01010000-01010011
		01001100-01000010-01001000
		01010110-01000110-01001100
		01000011-01001110-01010011

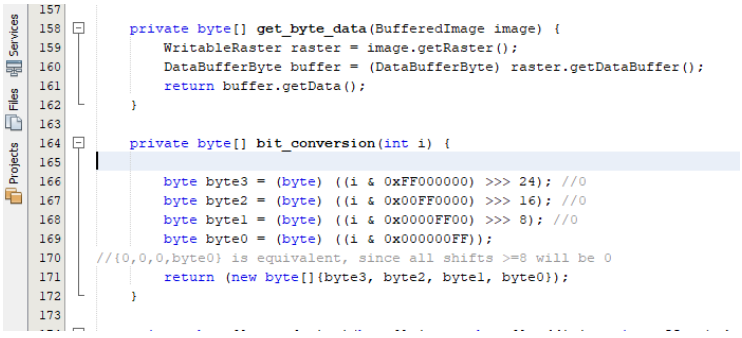
Catatan: Kunci pada enkripsi ini adalah NISAK yang berjumlah 5 digit. Pada tabel ASCII, digit ke-5 dikonversikan menjadi huruf E yang menghasilkan binnary text 01000101.

Hasil LSB terdapat pada tabel IV.

TABLE V. HASIL TEKNIK LSB

Bit LSB	:	0101100**0**-01010001-0101001**0**
		0100110**0**-0100001**0**-0100100**1**
		0101011**0**-0100011**1**-01001100
		01000011-01001110-01010011
Huruf Alfabet (Teknik LSB)	:	X-**Q**-**R**
		L-B-**I**
		V-**G**-L
		C-N-S

Source code pada Java terdapat pada gambar 9.



```
157
158     private byte[] get_byte_data(BufferedImage image) {
159         WritableRaster raster = image.getRaster();
160         DataBufferByte buffer = (DataBufferByte) raster.getDataBuffer();
161         return buffer.getData();
162     }
163
164     private byte[] bit_conversion(int i) {
165
166         byte byte3 = (byte) ((i & 0xFF000000) >>> 24); //0
167         byte byte2 = (byte) ((i & 0x00FF0000) >>> 16); //0
168         byte byte1 = (byte) ((i & 0x0000FF00) >>> 8); //0
169         byte byte0 = (byte) ((i & 0x000000FF));
170   //{0,0,0,byte0} is equivalent, since all shifts >=8 will be 0
171         return (new byte[]{byte3, byte2, byte1, byte0});
172     }
173
```

Fig. 9. Teknik LSB

ISBN: 978-602-60280-1-3

E. Pada langkah ini adalah membuat plain text yang dimasukkan dalam file berekstensi .txt. Selain .txt, Ekstensi file yang dapat menjadi file pesan antara lain .doc .xls. Contoh pada gambar 10 dan gambar 11 terdapat file ekstensi .txt sebagai isi pesan

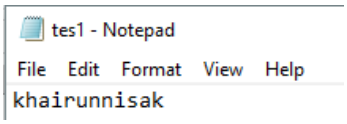


Fig. 10. Contoh isi pesan

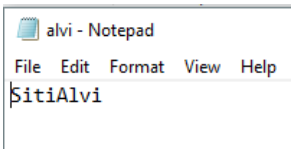


Fig. 11. Contoh isi pesan kedua

F. Pada bagian encode, memilih gambar yang menjadi tempat disembuyikannya pesan. Lalu, mengisi kolom file pesan dengan mencari file yang sebelumnya sudah dibuat daan disimpan pada gambar dan memasukkan kunci pesan.

G. Selanjutnya pilih encoding image, maka muncul pemberitahuan bahwa gambar berhasil disisipi pesan rahasia dengan nama file gambar berbeda yang berada disebelah kanan. interface enkripsi sekaligus steganografi gambar ditunjukkan pada gambar 12 dan gambar 13.

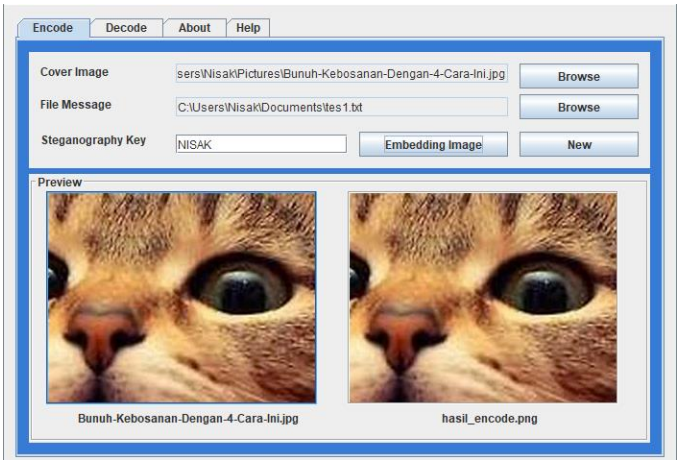


Fig. 12. *Interface* Enkripsi sekaligus Steganografi gambar pertama

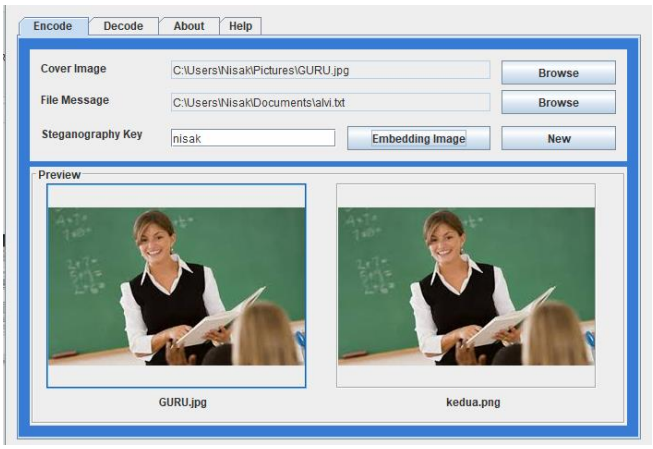


Fig. 13. *Interface* Enkripsi sekaligus Steganografi gambar kedua

H. Langkah kedelapan adalah dekripsi dari hasil chipper text. Langkah-langkah yang harus diperhatikan adalah

1) Mengetahui *binnary text* LSB dari hasil *chipper text*
2) Mengetahui *binnary text* dari kunci pesan
3) Mengubah ke bentuk semula hasil LSB berpedoman dengan *binnary text* kunci pesan
4) Mengubah *binnary text* menjadi huruf abjad alfabet atau dapat mengkonversinya ke dalam huruf hijaiyah
5) Mengkonversi isi pesan tersebut dengan kunci pesan untuk menghasilkan *plain text*.

Source code dekripsi pesan terdapat pada gambar 14

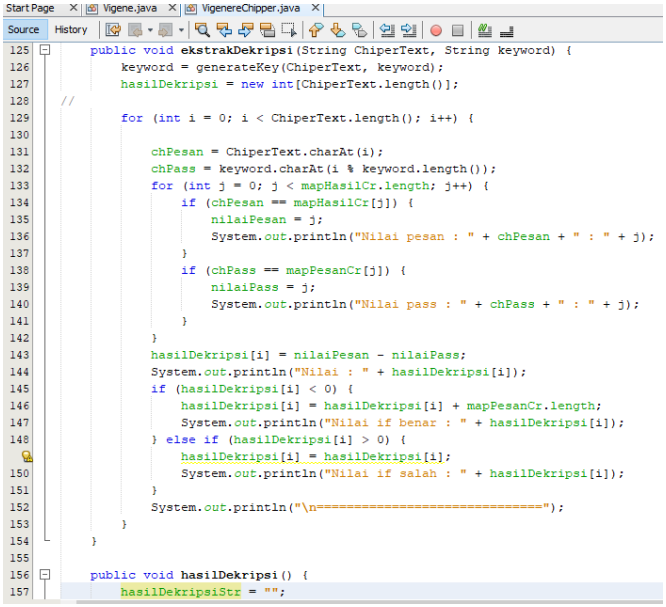


Fig. 14. *Source code* dekripsi pesan

Sedangkan *interface decoding image* dan dekripsi pesan terdapat pada gambar 15 untuk gambar pertama dan gambar 16 gambar kedua.

ISBN: 978-602-60280-1-3

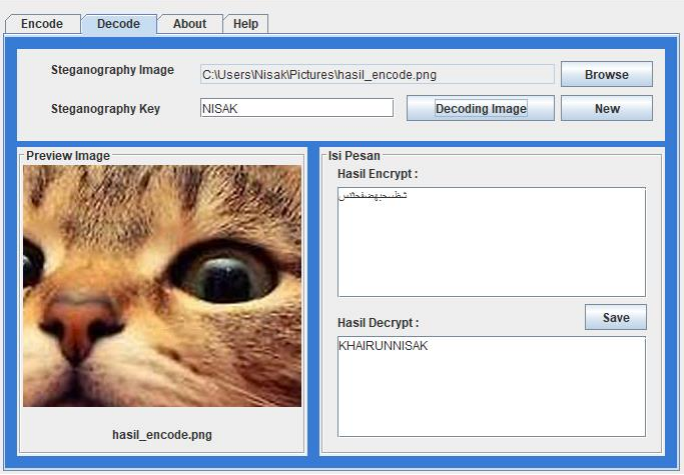


Fig. 15. *Interface decoding image* dan dekripsi pesan

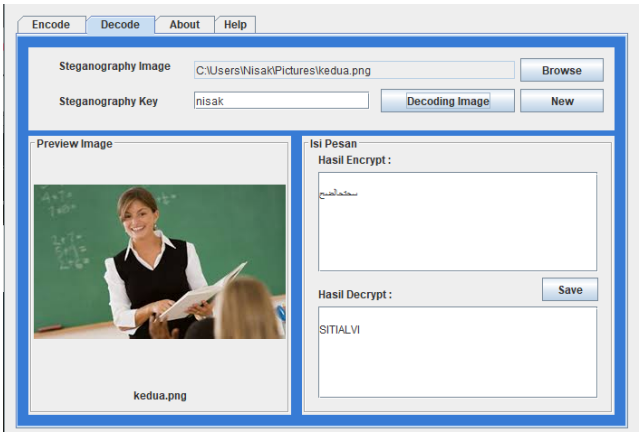


Fig. 16. *Interface decoding image* dan dekripsi pesan gambar kedua

Perbandingan jumlah bit gambar asli dan gambar berisi pesan rahasia ada pada gambar 17 dan 18 untuk gambar pertama dan gambar 19 serta 20 untuk gambar kedua.

Fig. 17. Gambar asli pertama dengan jumlah bit 48 KB

Fig. 18. Hasil *encode* gambar pertama dengan jumlah bit 885 KB

Fig. 19. Gambar asli kedua dengan jumlah bit 8 KB

Fig. 20. Hasil *encode* gambar kedua dengan jumlah bit 72 KB

Dari hasil kedua gambar tersebut perbedaan antara gambar asli dan gambar yang telah disisipi pesan yang dapat dilihat pada daftar metadata di tabel V. Informasi metadata dilihat menggunakan bantuan apliaksi IrfanView.

TABLE VI. HASIL META DATA

List of details	Contoh Gambar			
	Gambar 1		Gambar 2	
	Asli	Hasil Enkripsi	Asli	Hasil Enkripsi
Nama	Bunuh kebosanan dengan 4 cara ini	Hasil_encode	GURU	kedua
Ukuran dalam byte	48,2 kB	885 kB	5,69 kB	69,1 kB
Ukuran dalam pixel	1090x490 pixel	1090x490 pixel	294x171 pixel	294x171 pixel
Number of unique color	88682	88721	14242	14265
Current directory index	36/88	53/88	51/88	60/88
Loaded in (millisc)	125	15	0	0
Bit depth	24	24	24	24

ISBN: 978-602-60280-1-3

Format	.jpg	.png	.jpg	.png

Pada penelitian ini penulis mengembangkan *source code* aplikasi milik Karya Gunawan yang di akses melalui http://jagungodak.web.id/2015/01/26/steganografi-menggunakan-least-significant-bit-lsb-pada-java/ dengan beberapa modifikasi di dalamnya sesuai kebutuhan antara lain,

a) Adanya modifikasi tabel vigenere huruf menjadi tabel vigenere huruf hijaiyah dengan tabel konversi yang terdapat pada gambar 6 dan 7.
b) Adanya tabel bantuan yang dibuat oleh penulis pada gambar 2 dan 3 untuk memudahkan konversi huruf hijaiyah ke huruf alfabet dan di implementasikan ke dalam baris kode program.
c) Pada aplikasi yang dikembangkan oleh penulis, terdapat aturan tersendiri sebelum text diubah menggunakan teknik LSB yaitu jumlah digit karakter kunci menjadi penentu acuan digit binnary yang akan di masukkan ke dalam karakter *plain text* yang ada. Langkah tersebut di jelaskan pada langkah D.

IV. KESIMPULAN

Penyisipan pesan atau informasi dengan mengkombinasikan teknik kriptografi modifikasi algoritma vigenere huruf hijaiyah dengan salah satu teknik steganogarfi *Least Significant Bit* yang diimplementasikan ke dalam aplikasi berbasis Java, maka dapat ditarik kesimipulan

1. Aplikasi dapat melakukan proses penyembunyian dengan tingkat keamanan ganda karena terdapat dua lagkah keamanan yaitu enkripsi dan steganogarfi gambar.
2. Penyembunyian pesan melalui gambar tidak merubah kualitas gambar secara kasat mata, hanya terdapat perubahan ukuran pada gambar asli dan gambar berisi pesan rahasia.
3. Gambar yang disisipi pesan rahasia memiliki ukuran yang lebih besar dari gambar asli. Ukuran file semakin besar tergantung pada pesan rahasia yang disembunyikan dalam sebuah gambar.
4. Proses dekripsi pesan dapat dengan mudah dijalankan dan dilakukan dalam satu proses dengan perubahan isi pesan ke dalam huruf hijaiyah (arab).

ACKNOWLEDGMENT

Puji syukur kehadiran Allah swt. atas limpahan rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan paper ini dengan baik. Penghargaan dan ucapan terima kasih kami berikan kepada Dony Ariyus, M.Kom selaku dosen mata kuliah *Cyber Security*, Mery Cahyani, dan Yoga Willy Utomo teman kami yang turut mendukung proses pengembangan aplikasi steganografi gambar ini.

REFERENCES

[1] Y. D. Widiannanto and E. Z. Astuti, "Impelementasi Teknik Steganografi Metode Least Significant Bit dengan Algoritma Kriptografi Vigenere pada Citra," 2 July 2018. [Online]. Available: http://mahasiswa.dinus.ac.id.

[2] A. E. Handoyo, D. R. I. M. Setiadi, E. H. Rachmawanto, C. A. Sari and Ajib, "Teknik Penyembunyian dan Enkripsi Pesan pada Citra Digital dengan Kombinasi Metode LSB dan RSA," *Teknologi dan Sistem Komputer*, vol. VI, no. 1, pp. 37-43, 2018.

[3] I. G. A. P. Dewangga, T. W. Purboyo and R. A. Nugrahaeni, "A New Approach of Data Hiding in BMP Image Using LSB Steganography and Caesar Vigenere Chiper Cryptography," *International Journal of Applied Engineering Research*, vol. XII, no. 21, pp. 10626-10636, 2017.

[4] D. A. Z, T. W. Purboyo and R. A. Nugrahaeni, "Implementation of Secure Steganography on Jpeg Image Using LSB Method," *International Journal of Applied Engineering Research*, vol. XIII, no. 1, pp. 442-448, 2018.

[5] Irfan, "Penyembunyian Informasi (steganography) Gambar menggunakan metode LSB," *Rekayasa Teknologi*, vol. V, no. 1, 2015.

[6] I. Saputra, Mesran, N. A. Hasibuan and R. Rahim, "Vigenere Cipher Algorithm with Grayscale Image Key Generator for Secure Text File," *International Journal of Engineering Research & Technology (IJERT)*, vol. VI, no. 1, pp. 266-269, 2017.

[7] G. C. Kessler, "An Overview of Cryptography," 19 June 2018. [Online]. Available: http://www.garykessler.net/library/crypto.html.

[8] D. Darisman, P. Sokibi and M. Asfi, "Aplikasi Steganografi untuk Penyembunyian Data ke Dalam Citra Digital dengan Kombinasi Metode Least Significant Bit dan Algoritma Chiper," *DIGIT*, vol. IV, no. 2, pp. 240-257, 2014.

[9] A. A. Fitri, M. Mulya and Alfarissi, "Steganografi pada Citra Digital Berwarna 32-Bit menggunakan Least Significant Bit," in *Annual Research Seminar Universitas Sriwijaya*, Palembang, 2016.

[10] P. Hernawandra, Supriyadi and U. T. Lenggana, "Aplikasi Steganografi Menggunakan LSB 4 Bit Sisipan dengan Kombinasi Algoritme Substitusi dan Vigenere Berbasis Android," *Teknologi dan Sistem Komputer*, vol. VI, no. 2, pp. 44-50, 2018.

[11] A. K. Singh, J. Singh and D. H. V. Singh, "Steganography in Images Using LSB Technique," *International Journal of Latest Trends in Engineering and Technology (IJLTET)*, vol. V, no. 1, pp. 426-430, 2015.

[12] A. Rohmani, "Implementasi Kriptografi dan Steganografi dengan Metode Algoritma DES dan Metode End of File," *Informatika SMANTIK*, vol. I, no. 2, 2017.

[13] I. Santiko, "Implementasi Model Steganografi Dalam Mengelola Kerahasiaan Informasi dengan Metode LSB," in *CITISEE (Conference on Information Technology, Information System and Electrical Engineering)*, Yogyakarta, 2016.

ISBN: 978-602-60280-1-3

[14] M. Imron, I. Ardiansyah and D. Suhartono, "Implementasi Pengamanan Data Koperasi Menggunakan Algoritma Advanced Encryption Standard (AES)," in *CITISEE (Conference on Information Technology, Information System and Electrical Engineering)*, Yogyakarta, 2016.