JISKa, Vol. 4, No. 3, JANUARI, 2020, Pp. 138 – 148
ISSN : 2527 – 5836 (print) | 2528 – 0074 (online)

Penerapan Algoritma Hill Cipher Dan Least Significant Bit (LSB) Untuk Pengamanan Pesan Pada Citra Digital

Desimeri Laoli[3], Bosker Sinaga[2], Anita Sindar[3]
Teknik Informatika STMIK Pelita Nusantara
Jl. Iskandar Muda No. 1 Medan
Email : desimeri9094laoli@gmail.com[1], sinagab8@gmail.com[2], haito_ita@yahoo.com[3]

Abstract

Nowadays people exchange information in digital media such as text, audio, video and imagery. The development of Information and Communication makes the delivery of information and data more efficient. Current developments in technology which are very significant have an impact on the community in exchanging information and communicating. Confidential hidden data can also be in the form of image, audio, text, or video. The Hill Chiper algorithm uses a matrix of size m x m as a key for encryption and decryption. One way to recover the original text is of course to guess the decryption key, so the process of guessing the decryption key must be difficult. break ciphertext into palintext without knowing which key to use. The LSB part that is converted to the value of the message to be inserted. After affixing a secret message, each pixel is rebuilt into a whole image that resembles the original image media. The Hill Cipher algorithm is used to determine the position of the plaintext encryption into a random ciphertext. 2. Testing text messages using the hill cipher algorithm successfully carried out in accordance with the flow or the steps so as to produce a ciphertext in the form of randomization of the letters of the alphabet.

Keywords : Image, Confidential Data, Cryptography, Hill Chiper Algorithm, LSB

Abstrak

Saat ini masyarakat bertukar informasi dalam media digital seperti teks, audio, video, dan citra. Perkembangan Informasi dan Komunikasi menjadikan kegiatan penyampaian informasi maupun data menjadi lebih efesien. Perkembangan teknologi saat ini yang sangat signifikan memberikan dampak bagi masyarakat dalam bertukar informasi maupun melakukan komunikasi. Data rahasia yang disembunyikan juga dapat berupa citra, audio, teks, atau video. Algoritma Hill Chiper menggunakan matriks berukuran m x m sebagai kunci untuk melakukan enkripsi dan dekripsi. Satu cara untuk mendapatkan kembali naskah asli tentunya dengan menerka kunci dekripsi, jadi proses menerka kunci dekripsi harus menjadi sesuatu yang sulit. memecahkan chiperteks menjadi palinteks tanpa mengetahui kunci yang digunakan. Bagian LSB yang diubah menjadi nilai dari pesan yang akan disisipkan. Setelah dibubuhi pesan rahasia, setiap pixel dibangun kembali menjadi gambar yang utuh menyerupai dengan media gambar semula. Algoritma Hill Cipher yang digunakan untuk menentukan posisi enkrip plaintext menjadi sebuah ciphertext yang secara acak. 2. Pengujian pesan teks menggunakan algoritma hill cipher berhasil dilakukan sesuai tepat dengan alur atau langkah-langkahnya sehingga menghasilkan cipherteks yang berupa pengacakan huruf abjad.

Kata Kunci : Citra, Data Rahasia, Kriptografi, Algoritma Hill Chiper, LSB

1. PENDAHULUAN

Keamanan dan kerahasiaan informasi yakni tolak ukur yang sangat penting dalam sistem informasi. Meningkatnya kemajuan di bidang informasi dan teknologi yang menyebabkan adanya cara-cara terbaru, yang digunakan dengan tidak bertanggung jawab oleh beberapa oknum yang menyalahgunakan fungsi keamanan akan sebuah sistem informasi. Informasi tersebar ke tangan okum lain dapat menimbulkan efek negatif untuk pemilik informasi. Secara

umum informasi dikategorikan menjadi dua, yaitu informasi yang bersifat rahasia dan informasi yang tidak bersifat rahasia. Informasi yang tidak bersifat rahasia biasanya tidak akan terlalu diperhatikan. Informasi bersifat rahasia yaitu setiap informasi yang ada didalamnya sangat berharga bagi pihak yang membutuhkan karena informasi tersebut dapat dengan mudah digandakan. Informasi bisa dalam berbentuk sebuah file ataupun string [1]. Mengacu pada permasalahan yang dibahas maka diperlukan untuk merancang sebuah sistem keamanan yang dapat melindungi data yang dianggap penting dengan penyandian data, serta membuat kunci rahasia untuk dapat membuka data tersebut yang sulit untuk di deteksi oleh pihak yang tidak berhak. Gabungan dari metode kriptografi yakni algoritma Hill Cipher untuk enkripsi pesan dengan metode steganografi yaitu Least Significant Bit (LSB) dapat menambah keamanan dalam sebuah pesan [2]. Menurut Jurnal Fresly Nandar Pabokory, Kriptografi merupakan salah satu ilmu maupun seni untuk menjaga kerahasiaan sebuah pesan dengan cara menyandikannya ke dalam bentuk yang tidak dapat dimengerti lagi maknanya. Proses enkripsi dilakukan menggunakan suatu algoritma dengan beberapa alur kerja dari proses enkripsi. Algoritma yang dipakai pada penyusunan skripsi berikut ini yaitu Hill Cipher [3]. Jurnal Jane Irma Sari, Steganografi merupakan seni untuk menyembunyikan pesan di dalam media digital sedemikian rupa sehingga orang lain tidak menyadari ada sesuatu pesan di dalam media. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya citra, audio, teks, dan video [4].

2. METODE PENELITIAN

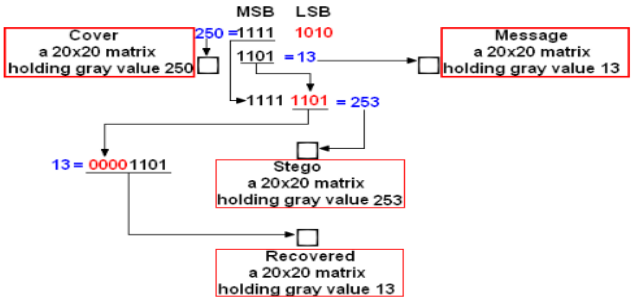
Teknik kriptografi ini diciptakan dengan maksud untuk dapat menciptakan cipher (kode) yang tidak dapat dipecahkan menggunakan teknik analisis frekuensi. Hill Cipher tidak mengganti setiap abjad yang sama pada plaintext dengan abjad lainnya yang sama pada ciphertext karena menggunakan perkalian matriks pada dasar enkripsi dan dekripsinya. Oleh karena itu Hill Cipher termasuk dalam salah satu kriptosistem polialfabetik. Cipher ini ditemukan pada tahun 1929 oleh Lester S. Hill. Teknik enkripsi yang digunakan adalah enkripsi simetris kunci dekripsi sama dengan kunci enkripsi. Untuk *public key cryptography,* diperlukan teknik enkripsi asimetris kunci dekripsi tidak sama dengan kunci enkripsi [5]. Enkripsi, dekripsi dan pembuatan kunci untuk teknik enkripsi asimetris memerlukan komputasi yang lebih intensif dibandingkan enkripsi simetris, karena enkripsi asimetris menggunakan bilangan-bilangan yang sangat besar. Walaupun enkripsi asimetris lebih “mahal" dibandingkan enkripsi simetris, *public key cryptography* sangat berguna untuk *key management* dan *digital signature.* suatu proses enkripsi yang baik menghasilkan naskah acak yang memerlukan waktu yang lama (contohnya satu juta tahun) untuk didekripsi oleh seseorang yang tidak mempunyai kunci dekripsi [6]. Semakin banyak proses yang diperlukan berarti juga semakin lama waktu yang dibutuhkan, maka semakin kuat algoritma tesebut dan semakin aman digunakan untuk menyandikan pesan [7]. Dasar dari teknik Hill Cipher adalah aritmatika modulo terhadap matriks. Dalam penerapannya, Hill Cipher menggunakan teknik perkalian matriks dan teknik invers terhadap matriks. Kunci pada Hill Cipher adalah matriks n x n dengan n merupakan ukuran blok [8]. Matriks K yang menjadi kunci ini harus merupakan matriks yang invertible, yaitu memiliki inverse K-1 sehingga kunci harus memiliki invers karena matriks K- adalah kunci yang digunakan untuk melakukan dekripsi [9]. Proses enkripsi pada Hill Cipher dilakukan per blok plaintext. Ukuran blok tersebut sama dengan ukuran matriks kunci. Sebelum membagi teks menjadi deretan blok-blok, plaintext terlebih dahulu dikonversi menjadi angka, masing-masing sehingga A=0, B=1, hingga Z=25. Secara matematis, proses enkripsi pada Hill Cipher adalah: C = K . P; dengan C = Ciphertext K = Kunci P = Plaintext.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

Proses dekripsi pada Hill Cipher:
C = K . P
K^{-1} . C = K^{-1} . K . P
K^{-1} . C =I . P
P = K^{-1} . C

Menjadi persamaan proses deskripsi: P= K^{-1} . C untuk menentukan K^{-1}: ; $\frac{1}{det\,det\,K} mod\ 26 = x$ atau (det K * x mod 26 = 1).

Least Significant Bit adalah salah satu metode untuk menyembunyikan pesan dalam media digital dengan cara menyisipkan pesan tersebut pada satu bit paling kanan ke pixel file obyek. Dalam menyisipkan data pesan ke dalam berkas citra digital dengan menggunakan metode *Least Significan Bit* (LSB) *Modification* [10]. Modul proses output data akan memisah kembali antara file citra digital dan data pesan rahasia pada suatu stegano image serta melakukan dekripsi pesan menjadi *ciphertext* [11].

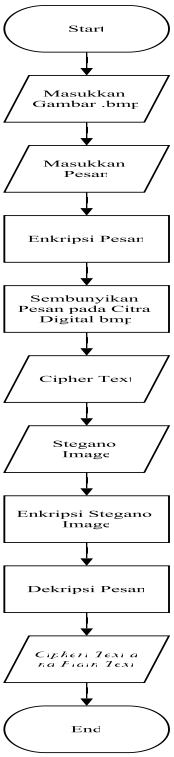


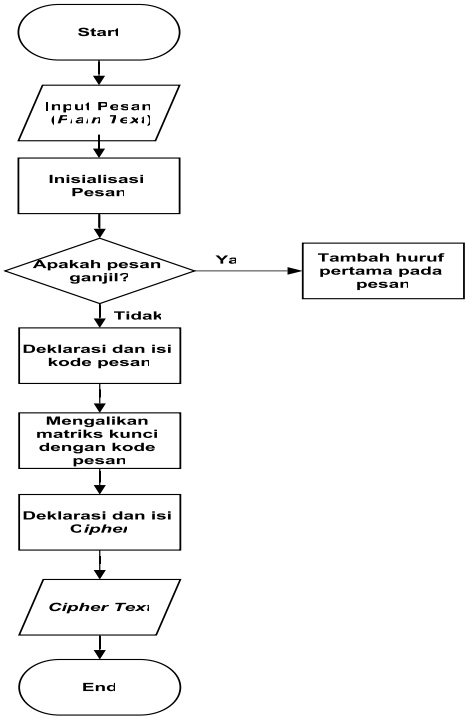
Gambar 1. Mekanisme LSB

Gambar 1 menunjukkan penerapan LSB menggunakan media gambar berbasis pixel dengan nilai 8 bit (*gray value*). Setiap pixel yang terdiri dari 8 bit dibagi menjadi 2 bagian yaitu, 4 bit MSB (*most significant bit*) dan 4 bit LSB (*least significat bit*).

3. HASIL DAN PEMBAHASAN

Dalam perancangan algoritma digunakan pendekatan terstruktur (structured approach) (Gambar 2). Algoritma Hill Cipher digunakan untuk menentukan posisi enkrip plaintext menjadi ciphertext yang secara acak (Gambar 3).





Gambar 2. Algoritma Umum Perangkat Lunak

Gambar 3. Algoritma Hill Cipher

Proses enkripsi pada hill cipher dilakukan per blok plaintext. Ukuran blok tersebut sama dengan ukuran matriks kunci. Sebelum membagi teks menjadi deretan blok-blok, plaintext dikonversi menjadi angka, masing-masing sehingga A=1, B=2, hingga y-25. Z diberi nilai 0.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

Proses enkripsi pada *hill cipher* :
C = K.P(2);
C = Ciphertext
K = Kunci
P = Plaintext
Jika terdapat *plaintext* P:
P = SELAMAT
Maka *plaintext* tersebut dikonversi menjadi:
P = 18 4 11 0 12 0 19
Plaintext tersebt akan dienkripsi dnegan teknik *hill cipher*, dengan kunci K yang merupakan matriks 2x2.

$$K = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$$

Matrik kunci K berukuran 2x2, maka *plaintext* dibagi menjadi blok yang masing-masing bloknya berukuran 2 karakter. Karena karakter terakhir tidak ada memiliki pasangan karakter yang sama yaitu W.P menjadi SELAMAT. Blok pertama dari *plaintext* P adalah:

$$P_{1,2} = \begin{bmatrix} 18 \\ 4 \end{bmatrix}$$

Blok *plaintext* dienkripsi dengan kunci K :

$$C_{1,2} = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} \begin{bmatrix} 18 \\ 4 \end{bmatrix} = \begin{bmatrix} 38 \\ 194 \end{bmatrix}$$

Hasil perhitungan menghasilkan angka yang tidak berkorespondensi dengan huruf, maka lakukan modulo 26 pada hasil sehingga $C_{1,2}$ menjadi:

$$C_{1,2} = \begin{bmatrix} 38 \\ 194 \end{bmatrix} = \begin{bmatrix} 12 \\ 12 \end{bmatrix} (mod\ 26)$$

Karakter yang berkorespondensi dengan 12 dan 12 adalah M dan M, maka S menjadi M dan E menjadi M. Setelah melakukan enkripsi semua blok pada *plaintext* P maka menghasilkan *ciphertext* C : P = SELAMAT ; C = 12 12 11 21 12 4 10 11 ; C = MMLVMEKL
Dari *ciphertext* yang dihasilkan terlihat bahwa *hill cipher* menghasilkan *ciphertext* yang tidak memiliki pola yang mirip dengan *plaintext*-nya. Proses dekripsi pada *hill cipher* pada dasarnya sama dengan proses enkripsi. Namun matriks kunci harus dibalik (invers) terlebih dahulu. Secara matematis, proses dekripsi pada *hill cipher* :

$$C = K.P\ ;\ K^{-1}.C = K^{-1}.K.P\ ;\ K^{-1}.C = I.P\ ;\ P = K^{-1}.C$$

Proses dekripsi : $P = K^{-1}.C$; menggunkan kunci $K = \begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$
Maka, proses dekripsi diawali dnegan mencari invers dari matriks *K*. Mencari invers dapat dilakukan dengan menggunakan metode operasi baris (*row operation*) atau metode determinan. Setelah melakukan perhitungan, didapat matriks K^{-1} yang merupakan invers dari matriks *K*, yaitu:

$$K^{-1} = \begin{bmatrix} 27 & -54 \\ -18 & 45 \end{bmatrix} = \begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} (mod\ 26)$$

Kunci K^{-1} yang digunakan untuk melakukan dekripsi ini telah memnuhi persamaan (1) karena:

$$K.\ K^{-1} = \begin{bmatrix} 53 & 234 \\ 26 & 105 \end{bmatrix} = K^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} (mod\ 26) = I$$

Ciphertext C = MMLVMEKL, akan didekripsi dengan menggunakan kunci dekripsi K^{-1} dengan persamaan (3). Proses dekripsi ini dilakukan blok per blok seperti pada proses enkripsi. Pertama-tama ubah huruf-huruf pada *ciphertext* urutan numerik. C = 12 12 11 21 12 4 10 11
Proses dekripsi dilakukan sebagai berikut:

$$P_{1,2} = K^{-1}.C_{1,2}\ ;\ P_{1,2} = \begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} \begin{bmatrix} 12 \\ 12 \end{bmatrix} = \begin{bmatrix} 252 \\ 264 \end{bmatrix} (mod\ 26) = \begin{bmatrix} 18 \\ 4 \end{bmatrix}$$

$P_{3,24} = K^{-1}.C_{3,4}$; $P_{3,24} = \begin{bmatrix}4 & 17\\15 & 7\end{bmatrix}\begin{bmatrix}11\\21\end{bmatrix} = \begin{bmatrix}401\\312\end{bmatrix}(mod\ 26) = \begin{bmatrix}11\\0\end{bmatrix}$

Setelah semua blok selesai didekripsi, maka didaapkan hasil *plaintext*: P = 18 4 11 0 12 0 19;
P = SELAMAT

Least Significant Bit adalah salah satu metode untuk menyembunyikan pesan dalam media digital dengan cara menyisipkan pesan tersebut pada satu bit paling kanan ke pixel file obyek. Dalam menyisipkan data pesan ke dalam berkas citra digital dengan menggunakan metode *Least Significan Bit* (LSB) *Modification*. Misalkan untuk menyisipkan suatu segmen pesan hasil dan modulasi sebesar 4 byte dengan modifikasi 1 bit LSB, maka dibutuhkan 32 data citra digital untuk menampungnya. Dari segmen pesan ' 1 0 1 0 ' dengan 4 byte data citra digital: 0 1101110 00100011 01000010 01101101'.

Dengan operasi penggantian bit terakhir dengan 4 bit segmen pesan secara berurutan menjadi:
Data citra digital: '0 1 1 0 1 1 1 0 0 0 1 0 0 0 1 1 0 10 0 0 0 1 0 0 1 1 0 1 1 0 1'
Pesan: 1 0 1 0
Hasil: '0 1 1 0 1 1 1 <u>1</u> 0 0 1 0 0 0 1 <u>0</u> 0 1 0 0 0 0 1 <u>1</u> 0 1 1 0 1 1 0 <u>0</u>'

Dengan sedikit modifikasi ini, maka efek dari perubahan nilai warna yang terjadi akibat perubahan bit tersebut tidak terlalu berpengaruh terhadap kualitas gambar. Perhatikan contoh untuk menyisipkan sebuah karakter A ke dalam citra *grayscale*. Sebuah pesan huruf A akan disisipkan ke dalam citra *grayscale* 8 bit ukuran 10x10 piksel.

1	6	5	3	7	4	7	4	1	0
3	5	3	5	5	5	5	7	7	0
0	0	0	2	2	6	6	6	6	6
5	5	4	4	4	4	4	4	7	3
2	2	0	0	0	0	1	1	1	1
7	5	5	5	7	7	7	6	3	3
3	3	3	3	3	3	3	3	7	5
5	5	5	5	5	5	5	5	2	3
0	0	0	0	0	0	4	4	4	4
3	3	3	3	3	1	1	1	6	2

Langkah pertama adalah mengunbah kedua data tersebut (huruf A dan citra) menjadi biner. Nilai biner untuk A adalah 10000011. Karena jumlah digit biner huruf A hanya 8 bit maka jumlah piksel citra *grayscale* yang dibutuhkan cukup 8 piksel saja. 8 piksel pertama dari citra yang diubah menjadi biner.

8 piksel pertama

1	6	5	3	7	4	7	4	1	0
3	5	3	5	5	5	5	7	7	0
0	0	0	2	2	6	6	6	6	6
5	5	4	4	4	4	4	4	7	3
2	2	0	0	0	0	1	1	1	1
7	5	5	5	7	7	7	6	3	3
3	3	3	3	3	3	3	3	7	5
5	5	5	5	5	5	5	5	2	3
0	0	0	0	0	0	4	4	4	4
3	3	3	3	3	1	1	1	6	2

Langkah kedua mengganti bit terakhir dari piksel citra dengan bit dari huruf A 1 piksel media.

Tabel 1 piksel citra yang diambil

Piksel Citra		Huruf A
Decimal	Biner	
1	00000001	1
6	00000110	0
5	00000101	0
3	00000011	0
7	00000111	0
4	00000100	0
7	00000111	1
4	00000100	1

piksel citra yang berubah	
Decimal	Biner
1	00000001
6	00000110
4	00000110
2	00000100
6	00000010
4	00000110
7	00000111
5	00000101

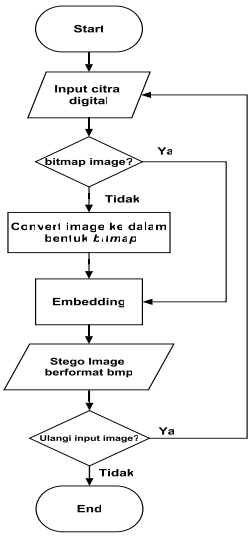
bit-bit yang ditandai dengan kotak. Bit-bit piksel citra mengalami perubahan;

8 Bit yang pertama

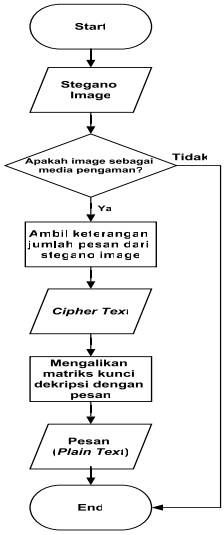
1	6	4	2	6	4	7	5	1	0
3	5	3	5	5	5	5	7	7	0
0	0	0	2	2	6	6	6	6	6
5	5	4	4	4	4	4	4	7	3
2	2	0	0	0	0	1	1	1	1
7	5	5	5	7	7	7	6	3	3
3	3	3	3	3	3	3	3	7	5
5	5	5	5	5	5	5	5	2	3
0	0	0	0	0	0	4	4	4	4
3	3	3	3	3	1	1	1	6	2

penampung = 8 bit. Untuk menampung 1 bit data pesan diperlukan 1 piksel citra media penampung berukuran 8 bit karena setiap 8 bit hanya bisa menyembunyikan satu bit di LSB-nya. Oleh karena itu, citra ini hanya mampu menampung data pesan sebesar maksimum 16384/8 = 2048 bit dikurangi panjang nama filenya karena penyembunyian data rahasia tidak hanya menyembunyikan isi data tersebut, tetapi juga nama filenya. Semakin besar data yang disembunyikan di dalam citra, semakin besar pula kemungkinan data tersebut rusak akibat manipulasi pada citra penampung, media penyimpanan hasil steganografi adalah citra digital berformat bitmap. Proses ini merupakan memanipulasi pesan chiperteks heksadesimal hasil enkripsi yang disisipkan dalam bitmap. Citra digital yang diinput dalam berbagai format yang nantinya pada proses steganografi akan dikonversikan dalam bentuk bitmap, Gambar 4. Proses Penyisipan Pesan, cover image yang digunakan format bitmap. Untuk lebih Jelasnya proses penyisipan pesan dan Penyisipan pesan steganografi LSB pada citra digital, Gambar 5. Proses extraction dilakukan hampir sama halnya dengan proses embedding atau penyisipan tetapi ada beberapa perbedaan yakni proses ini terlebih dahulu dipilih stego object yaitu image yang sudah disisipi sebuah pesan dan mendekripsi chipertext menjadi plaintext, Gambar 6.

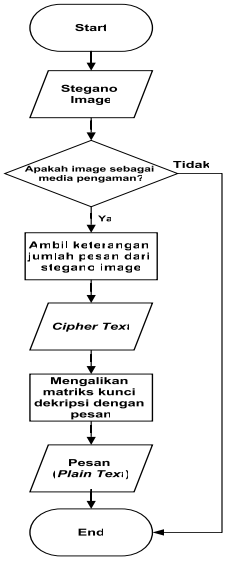
.



Gambar 4. Proses Pemilihan Citra Digital

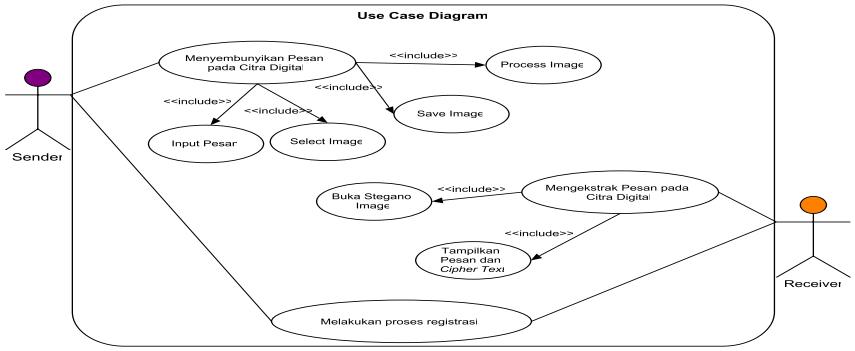


Gambar 5. Penyisipan Pesan



Gambar 6. Proses Ekstrasi Pesan

Use case diagram (Gambar 7), *user* pengirim memilih citra yang akan disisipi oleh teks. Kemudian *user* pengirim menuliskan pesan yang akan disembunyikan di dalam citra tersebut. Hasilnya adalah berupa citra yang telah disisipi oleh teks (*stego image).* Sedangkan *user* penerima adalah menerima *stego image,* kemudian memasukkan *password,* dan melakukan ekstraksi, sehingga pesan yang disisipkan dan disembunyikan akan muncul.



Gambar 7. Use Case Diagram

Implementasi sistem merupakan tahapan dalam menerapkan sistem yang telah dibangun, dimana nantinya akan diketahui kualitas dari sistem yang dirancang. Dalam melakukan implementasi dipersiapkan beberapa sarana yang berhubungan dengan perangkat keras (*hardware*) dan perangkat lunak (*software*). Aplikasi dirancang untuk membantu menyembunyikan pesan teks yang bersifat rahasia pada sebuah citra digital dengan format *bitmap*. Tampilan antar muka (*interface*) dari aplikasi pengamanan pesan pada citra digital dengan menggabungkan teknik kriptografi dan steganografi. *Form* Penyisipan Pesan (*Encode*) merupakan tampilan dimana *user* mamasukan/*input* citra digital dan pesan teks untuk proses pengamanan pesan teks kedalam suatu citra digital, Gambar 8. *Form Decode* merupakan tampilan dimana user mengembalikan data teks yang telah di enkripsi kebentuk pesan semula, Gambar 9.

Gambar 8. *Form Encode*

Gambar 9. *Form Decode*

Pengujian Algoritma *Hill Cipher*

Sebelum masuk kedalam proses penyandian terlebih dahulu ditetapkan pesan yang akan disandikan dan kunci matriks 2x2. Pesan yang disandikan maksimal 66 karakter tiap karakter harus berada diantara A-Z yang berjumlah 26 huruf dalam proses penyandian ini huruf besar dan kecil tidak dibedakan. Dan juga dalam penyandian ini tidak menggunakan kode ascii huruf tetapi dengan kode angka berdasarkan urutan huruf yaitu A=0, B=1, ... Z=25 dan spasi tidak dihitung dalam penyandian. Kunci yang telah ditetapkan digunakan di dalam proses enkripsi pesan berikut ini adalah langkah-langkah penyandian pesan mengunakan kunci matriks 2x2 :

1. Sisipkan pesan dan kunci matrik

   Pesan : P = SELAMAT Kunci : K = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix}$

Untuk mengetahui kode-kode dari pesan akan diberikan tabel kode masing-masing huruf dari A sampai dengan Z.

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	**O**	**P**	**Q**	**R**	**S**	**T**	**U**	**V**	**W**	**X**	**Y**	**Z**
13	14	15	16	17	18	19	20	21	22	23	24	25

2. Mengubah pesan menjadi kode dan matrik 2x2. Setelah diketahui masing-masing kode huruf maka pesan teks yang akan di sandikan diubah kedalam kode-kode angka dari tabel diatas, yaitu : Kode pesan : P = 18 4 11 0 12 0 19 19

Kemudian setelah didapat kode untuk masing-masing huruf kemudian setiap dua kode diubah kedalam matriks ordo 2 x 1, agar dapat dikalikan dengan kunci yang mempunyai matriks 2x2.

$$P = \begin{bmatrix} 18 \\ 4 \end{bmatrix} \begin{bmatrix} 11 \\ 0 \end{bmatrix} \begin{bmatrix} 12 \\ 0 \end{bmatrix} \begin{bmatrix} 19 \\ 19 \end{bmatrix}$$

Setelah matrik disusun maka Rumus: C=(P*K) mod 26

3. Mengalikan matriks pesan dan matriks kunci

Matriks pesan dikalikan dengan matriks kunci, dan hasil perkalikan tersebut diubah lagi kedalam huruf dengan referensi tabel kode masing-masing huruf. Proses perkalian matriks kunci dengan matriks pesan :

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 18 \\ 4 \end{bmatrix} = \begin{bmatrix} 38 \\ 194 \end{bmatrix}$

Karena hasil perkalian melebihi 25 maka hasil perkalian ini harus di mod 26 :

C = $\begin{bmatrix} 38 \\ 194 \end{bmatrix} Mod\ 26 = \begin{bmatrix} 12 \\ 12 \end{bmatrix}$; untuk K*P (LA)

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 11 \\ 0 \end{bmatrix} = \begin{bmatrix} 11 \\ 99 \end{bmatrix} mod\ 26 = \begin{bmatrix} 11 \\ 21 \end{bmatrix}$; untuk K*P (MA)

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 12 \\ 0 \end{bmatrix} = \begin{bmatrix} 12 \\ 108 \end{bmatrix} mod\ 26 = \begin{bmatrix} 12 \\ 4 \end{bmatrix}$; untuk K*P (TT)

C = $\begin{bmatrix} 1 & 5 \\ 9 & 8 \end{bmatrix} x \begin{bmatrix} 19 \\ 19 \end{bmatrix} = \begin{bmatrix} 114 \\ 323 \end{bmatrix} mod\ 26 = \begin{bmatrix} 10 \\ 11 \end{bmatrix}$

Setelah semua matriks pesan dikalikan dengan matriks kunci dan lakukan modulus dengan 26 hasil matrik *cipher*

$\begin{bmatrix} 12 \\ 12 \end{bmatrix} \begin{bmatrix} 11 \\ 21 \end{bmatrix} \begin{bmatrix} 12 \\ 4 \end{bmatrix} \begin{bmatrix} 10 \\ 11 \end{bmatrix}$

4. Mengubah matriks menjadi deret pesan. Setelah didapat hasil matriks dari perkalian antara matriks kunci dan pesan kemudian matriks disusun kembali berurutan:

C = 12, 12, 11, 21, 12, 4, 10, 11

5. Mengubah kode pesan menjadi huruf (karakter)

Kode diatas disusun kembali kedalam bentuk huruf dengan menggunakan tabel 5.1. Dan hasil ini adalah chiper hasil dari penyandian pesan yang akan di gunakan dalam proses steganografi, yaitu Ciphertext: MMLVMEKL

Setelah proses enkripsi selesai dilakukan, maka untuk mendeskripsi cipherteks menjadi pesan kembali sebenarnya hampir sama dengan cara enkripsi. Tetapi kunci yang digunakan harus di invers terlebih dahulu. Untuk lebih jelasnya tentang proses dekskripsi chiperteks akan dijelaskan secara rinci tentang tahap-tahap deksripsi.

1. Invers Matriks Kunci

Pengembalian pesan terdapat ketetapan yang telah ditentukan tahap pertama yaitu dengan melakukan invers terhadap matriks kunci yang digunakan dalam penyandian pesan. Rumus untuk menginvers matriks yang ber ordo 2x2. Proses untuk mengetahui invers matriks kunci invers

$$K^{-1} = \begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix}$$

Setelah invers matriks kunci selesai dilakukan maka didapat matriks. Kunci-1 untuk mendeskripsikan chiper menjadi pesan kembali matrik invers kunci dikalikan dengan matriks chipperteks. Ubah chipperteks ke dalam bentuk kode kembali sama persis seperti proses pengubahan pesan kedalam kode sewaktu proses penyandian.

2. Menyiapkan Pesan Cipher

Ciphertext: MMLVMEKL

Setelah diketahui cipherteks, ubah kedalam bentuk kode berdasarkan urutan angka.

3. Mengubah Chiper Menjadi Kode Dan Matriks

C = 12, 12, 11, 21, 12, 4, 10, 11

Setelah diketahui masing-masing kode dari cipherteks, kemudian diubah kedalam bentuk matriks *cipher*:

C =$\begin{bmatrix} 12 \\ 12 \end{bmatrix} \begin{bmatrix} 11 \\ 21 \end{bmatrix} \begin{bmatrix} 12 \\ 4 \end{bmatrix} \begin{bmatrix} 10 \\ 11 \end{bmatrix}$

Setelah diketahui matriks chipper kemudian chiper dikalikan dengan invers matrik kunci dan modulus dengan 26 dan hasil itu adalah pesan asli dari penyandian yang telah dilakukan sebelumnya. Rumus untuk menentukan pesan teks dari chiperteks P = K-1*C.

4. Mengalikan matriks pesan chipper dengan invers matriks kunci. Proses perkalian antara invers matriks kunci dengan matriks chipperteks, untuk kata MM:

P =$\begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} x \begin{bmatrix} 12 \\ 12 \end{bmatrix} = \begin{bmatrix} 252 \\ 264 \end{bmatrix} mod\ 26 = \begin{bmatrix} 18 \\ 4 \end{bmatrix}$; untuk kata LV:

P =$\begin{bmatrix} 4 & 17 \\ 15 & 7 \end{bmatrix} x \begin{bmatrix} 11 \\ 21 \end{bmatrix} = \begin{bmatrix} 401 \\ 312 \end{bmatrix} mod\ 26 = \begin{bmatrix} 11 \\ 0 \end{bmatrix}$; untuk kata ME:

P = $\begin{bmatrix}4 & 17\\15 & 7\end{bmatrix} x \begin{bmatrix}12\\4\end{bmatrix} = \begin{bmatrix}116\\208\end{bmatrix} mod\ 26 = \begin{bmatrix}12\\0\end{bmatrix}$; untuk kata KL:

P = $\begin{bmatrix}4 & 17\\15 & 7\end{bmatrix} x \begin{bmatrix}10\\11\end{bmatrix} = \begin{bmatrix}227\\227\end{bmatrix} mod\ 26 = \begin{bmatrix}19\\19\end{bmatrix}$

Dan seletelah semua matriks chipper kalikan dengan inverst matriks kunci K-1 maka hasil perkalian dan modulus 26:

$$P = \begin{bmatrix}18\\4\end{bmatrix}\begin{bmatrix}11\\0\end{bmatrix}\begin{bmatrix}12\\0\end{bmatrix}\begin{bmatrix}19\\19\end{bmatrix}$$

5. Mengubah matriks pesan menjadi deret kode

Setelah diketahui masing-masing matriks pesan kemudian matriks pesan diurutkan kedalam bentuk bilangan bulat biasa P = 18, 4, 11, 0, 12, 0, 19, 19

6. Mengubah kode menjadi pesan kembali

Langkah terakhir adalah mengubah pesan yang masih berbentuk kode menjadi bentuk huruf dengan berpedoman pada tabel kode masing-masing huruf yang tertera pada tabel 5.1, hasilnya:

P = HALO; Pengujian Metode *Least Significant Bit* (LSB)

Proses penyisipan pesan chiper kedalam citra gambar. Gambar yang digunakan adalah gambar berwarna 24 bit, yaitu gambar yang terdiri dari 3 warna R, G, B masing-masing warna mempunyai kedalaman warna sebesar 8 bit. Karena masing-masing warna bernilai 8 bit, maka pesan akan disisipkan kedalam bit R, bit G dan bit B tiap-tiap pixel. Misalkan pesan yang akan disisipkan sebanyak 8 bit, maka pesan yang 8 bit tersebut hanya akan disisipkan pada dua 3 pixel, karena tiap pixel memiliki kapasitas 24 bit dan masing-masing bit pesan hanya disisipkan pada 8 bit citra gambar. Dibawah ini adalah langkah-langkah proses steganografi untuk menyisipkan pesan kedalam citra gambar, Berikut ini merupakan bagaimana cara kerja dari algoritma LSB dimana teks HALO akan disisipkan kedalam gambar, namun terlebih dahulu teks tersebut diubah kedalam biner dengan nilai. Tabel 2.

Tabel 2. Kode ASCII Pesan yang akan disisipi

Teks	Biner
S	00010010
E	00000100
L	00001011
A	00000000
M	00001100
A	00000000
T	00010011

Setelah diubah kedalam biner lalu pesan akan disisipkan kedalam gambar pada warna (RGB) dengan metode LSB dengan nilai biner gambar awal Tabel 3.

Tabel 3. Kode media/gambar yang akan disisip

01110111	01110110	01110100	01000111
01110011	01110100	01110110	01110000
00110110	01110111	11110111	01110110
10110111	11110111	01110111	11110111
11010111	01110110	11110111	01110110
11110111	10110111	01111111	01110111
01110100	11000111	11110111	00010111
01111111	11010111	01111111	01110100

disisipkan pesan maka nilai biner gambar akan berubah menjadi Tabel 4.

Tabel 4. Kode media/gambar yang sudah disisip

0111011**0**	01110110	01110100	0100011**0**
01110011	0111010**1**	0111011**1**	0111000**1**
00110110	01110111	11110111	0111011**1**
1011011**0**	1111011**0**	0111011**0**	1111011**0**
11010111	01110110	11110111	0111011**1**
1111011**0**	1011011**0**	01111111	01110111
01110100	1100011**0**	1111011**0**	00010111
0111111**1**	11010111	0111111**0**	0111010**1**

Proses ekstraknya adalah mengambil nilai paling kanan dari biner yang disisipkan. Data biner yang telah diambil isi pesannya Tabel 5.

Tabel 5. Kode ASCII hasil ekstrak

Teks	Biner
S	00010010
E	00000100
L	00001011
A	00000000
M	00001100
A	00000000
T	00010011

Beberapa pengujian yaitu pengujian hasil teori dan hasil praktek (pengaplikasian), lalu pengujian perbandingan gambar sebelum dan sesudah pesan disisipkan.

Tabel 6. Analisa Hasil Pada Enkripsi Hill Cipher

ANALISA HASIL PADA ENKRIPSI *HILL CIPHER*		
Hill Cipher	Teori	Praktek
Pesan	SELAMAT	MMLVMEKL
Hasil	Huruf S menjadi M Huruf E menjadi M Huruf L menjadi L Huruf A menjadi V Huruf M menjadi M Huruf A menjadi E Huruf T menjadi K Sehingga kata SELAMAT menjadi kata MMLVMEKL	Dalam prakteknya dengan menekan tompol submit maka hasilnya adalah sebagai berikut: 

Pengujian selanjutnya adalah analisa hasil sebelum dan sesudah enkripsi/disisipkan pesan pada citra BMP. Adapun pesan yang disisipkan adalah “Proses mengubah citra analog menjadi citra digital disebut digitalisasi citra. Ada dua hal yang harus dilakukan pada digitalisasi citra, yaitu digitalisasi spasial yang disebut juga sebagai sampling (penerokan) dan digitalisasi intensitas yang sering disebut sebagai kuantisasi” Tabel 7.

Tabel 7. Perbandingan Citra BMP

ANALISA HASIL PADA CITRA BMP		
Detail	Citra Asli	Citra Stegano
Nama Citra	logo aplikasi.bmp	CITRA1.bmp
Ukuran (Mb)	1.20	1.20
Dimensi (Pixel)	650 x 650	650 x 650

Citra		

IV. KESIMPULAN

Berdasakan dari analisa, perancangan dan implementasi pada aplikasi pengamanan pesan pada citra digital dengan menggabungkan metode *Least Significant Bit* (LSB) dan algoritma *hill cipher*, dapat diambil kesimpulan:

1. Proses pengamanan pesan pada citra digital aman dan tidak diketahui secara kasat mata, karena besar dari *bitmap* hasil steganografi tidak terlihat secara signifikan perubahan setelah dari proses penyisipan biner teks ke dalam biner *bitmap* menggunakan metode *least significant bit* (LSB) yaitu penggantian bit terakhir sehingga kapasitas dari *bitmap* sebelum dan sesudah disteganografi tidak mengalami perubahan yang signifikan.
2. Pengujian pesan teks menggunakan algoritma *hill cipher* berhasil dilakukan sesuai tepat dengan alur atau langkah-langkahnya sehingga menghasilkan cipherteks yang berupa pengacakan huruf abjad.
3. Pesan yang akan diambil dari bitmap dapat dilanjutkan ke proses dekripsi yang bertujuan untuk mengembalikan pesan *ciphet text* ke bentuk semula (*plain text*) melalui proses dekripsi algoritma *hill cipher* yang sesuai.

DAFTAR PUSTAKA

M. M. Amin, 2016. Implementasi Kriptografi Klasik Pada Komunikasi Berbasis Teks, Jurnal Pseudocode, Volume III Nomor 2, September hal. 129-136.

Supiyanto. 2015. Implementasi Hill Cipher Pada Citra Menggunakan Koefisien Binomial Sebagai Matriks Kunci, Supiyanto, Seminar Nasional Informatika UPN ”Veteran” Yogyakarta, hal. 284-292.

Anita S, RMS. 2019. Sistem Bilangan Digital, 1, Serang Banten, CV. AA. Rizky.

Satriya, T, C, K., Dedih, Supriyadi, Implementasi Kriptografi Algoritma Rivest Shamir Adleman dengan Playfair Cipher pada Pesan Teks Berbasis Android, JOIN (Jurnal Online Informatika), Volume 2 No. 2, hal : 102-109.

Abdul H, H. 2013. Implementasi Algoritma Hill Cipher Dalam Penyandian Data, Pelita Informatika Budi Darma, Volume : IV, Nomor : 2.

Anita, S, RM, Sinaga. 2017. Implementasi Teknik Threshoding Pada Segmentasi Citra Digital, Jurnal Manajemen Dan Informatika Pelita Nusantara, Volume 1 No 2 hal : 48-51.

Indra, G. Sumarno. Eka, I. Heru, S, T. 2017. Pengamanan Berkas Dokumen Menggunakan Fungsi Algoritma Steganografi LSB, Jurnal Informatika Mulawarman, Vol. 12, No. 2.

Niria, L. Anita, S, RMS. 2018. Implementasi Steganografi LSB Dengan Enkripsi Vigenere Cipher Pada Citra, ScientiCO : Computer Science Informatics Journal Vol. 1, No. 2, hal : 47-58.

Anita, S, S. 2017. SEGMENTASI RUANG WARNA L*a*b Jurnal Mantik Penusa Vol. 3, No. 1 , pp. 43-46.

Rohmat, N, I. Ilham. MS. 2017. PERANCANGAN APLIKASI STEGAKRIP DENGAN METODE LSB DAN ALGORITMA RSA BERBASIS WEB, Jurnal Computech & Bisnis, Vol. 11, No 1, 98-109.