PENYEMBUNYIAN PESAN TERENKRIPSI *HILL CIPHER* PADA *FILE AUDIO* DENGAN METODE *LEAST SIGNIFICANT BIT* (LSB)

SKRIPSI

Oleh

Yuni Wahyuningsih
NIM 121810101069

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016

PENYEMBUNYIAN PESAN TERENKRIPSI *HILL CIPHER* PADA *FILE AUDIO* DENGAN METODE *LEAST SIGNIFICANT BIT* (LSB)

SKRIPSI

diajukan guna memenuhi salah satu persyaratan akademik pada program S1 Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember

Oleh

Yuni Wahyuningsih
NIM 121810101069

JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS JEMBER
2016

PERSEMBAHAN

Skripsi ini saya persembahkan untuk:

1. Allah SWT. yang telah memberikan kehidupan yang penuh makna;
2. Ayahanda Suharyono dan Ibunda Gisnawati tercinta, dua orang luar biasa dalam hidup yang telah memberikan segala cinta kasih dan pengorbanannya;
3. Keluarga BATHICS 2012, terimakasih telah berbagi suka dan duka bersama;
4. Sahabat-sahabat terkasih yang selalu mendukung dimanapun berada;
5. Almamater tercinta Jurusan Matematika FMIPA Universitas Jember.

MOTTO

"Sesuatu yang belum dikerjakan, seringkali tampak mustahil; kita baru yakin kalau kita telah berhasil melakukannya dengan baik."

(Evelyn Underhill)

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

nama : Yuni Wahyuningsih

NIM : 121810101069

menyatakan dengan sesungguhnya bahwa karya ilmiah yang berjudul "Penyembunyian Pesan Terenkripsi *Hill Cipher* pada *File Audio* dengan Metode *Least Significant Bit*" adalah benar-benar hasil karya sendiri, kecuali kutipan yang sudah saya sebutkan sumbernya, belum pernah diajukan pada institusi manapun, dan bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isiknya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa ada tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, Desember 2016

Yang menyatakan,

Yuni Wahyuningsih

NIM 121810101069

SKRIPSI

PENYEMBUNYIAN PESAN TERENKRIPSI *HILL CIPHER* PADA *FILE AUDIO* DENGAN METODE *LEAST SIGNIFICANT BIT*

Oleh
Yuni Wahyuningsih
NIM 121810101069

Pembimbing:

Dose Pembimbing Utama : Ahmad Kamsyakawuni, S.Si., M.Kom.

Dosen Pembbing Anggota : Kusbudiono, S.Si., M.Si.

PENGESAHAN

Skripsi berjudul "Penyembunyian Pesan Terenkripsi *Hill Cipher* pada *File Audio* dengan Metode *Least Significant Bit*" telah diuji dan disahkan pada:

hari, tanggal :

tempat : Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember

Tim Penguji:

Ketua,	Sekretaris,
Ahmad Kamsyakawuni, S.Si., M.Kom. NIP. 197211291998021001	Kusbudiono, S.Si., M.Si. NIP. 19770430200501001
Penguji I,	Penguji II,
Prof. Drs. Kusno, DEA., Ph.D. NIP. 196101081986021001	Dr. Mohamad Fatekurohman, S.Si.,M.Si NIP. 196906061998031001

Mengesahkan

Dekan,

Drs. Sujito, Ph.D.

NIP. 196102041987111001

RINGKASAN

Penyembunyian Pesan Terenkripsi *Hill Cipher* pada *File Audio* dengan Metode *Least Significan Bit*; Yuni Wahyuningsih, 1211010169; 2016; 53 Halaman; Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahua Alam Universitas Jember.

Komunikasi merupakan hal yang dilakukan setiap saat untuk saling bertukar informasi. Informasi atau pesan dapat menjadi sesuatu yang sangat berharga dan dijaga kerahasiaannya. Semakin berkembangnya teknologi dalam dunia telekomunikasi, semakin banyak pula kejahatan untuk mengetahui informasi atau pesan yang bukan menjadi haknya. Keamanan informasi dapat dilakukan dengan teknik kriptografi dan steganografi. Penelitian ini berfokus pada enkripsi pesan teks menggunakan algoritma *Hill Cipher*dengan menggunakan kunci matriks persegi berordo 3 x 3 sehingga menghasilkan *ciphertext*. Selanjutnya *ciphertext* tersebut disisipkan ke dalam *file audio* dengan menggunakan metode *Least Significant Bit* (LSB) sehingga dihasilkan *file stego audio*.

Pengujian pertama adalah uji panjang pesan yang mampu ditampung *file audio*. Pengujian kedua dilakukan dengan menghitung waktu eksekusi penyisipan dan ekstraksi. Pesan atau *Plaintext* yang diujikan berupa teks dengan ukuran atau panjang pesan 15 *byte*, 120 *byte* dan 1500 *byte*.*File audio* yang digunakan berupa *audio* berformat .wav dengan *genre* musik instrumental, akustik, R&B / Soul dan suara buatan sendiri berupa bunyi “beep”. Pengujian ketiga yaitu mengenai hasil suara yang dihasilkan setelah *file audio* disisipi pesan teks.

Hasil pengujian menunjukkan bahwa penerapan steganografi pada *file audio*.wav berhasil dilakukan. Kapasitas atau panjang pesan yang mampu ditampung *file audio* .wav dipengaruhi banyaknya *byte homogen*. Waktu eksekusi penyisipan dan ekstraksi yang paling tercepat adalah pada *file* dengan ukuran *file* paling kecil dan

yang paling lama adalah pada *file* dengan ukuran *file* paling besar.Kualitas suara *audio* yang dihasilkan tidak berubah setelah disisipkan pesan dan hasil ekstraksi *file stego audio* memiliki kesesuaian dengan pesan asli.

PRAKATA

Alhamdulillah, puji syukur kehadirat Allah SWT. yang telah memberikan rahmat dan hidayah-Nya sehingga tugas akhiryang berjudul "Penyembunyian Pesan Terenkripsi *Hill Cipher* pada *File Audio* dengan Metode *Least Significant Bit*" dapat terselesaikan dengan baik. Tugas akhir ini disusun untuk memenuhi syarat menyelesaikan pendidikan strata satu (S1) pada Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Uniersitas Jember.

Peulisan tugas akhir ini telah mendapat batuan dari berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih kepada:

1. Drs. Sujito, Ph.D., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember;
2. Ahmad Kamsyakawuni, S.Si., M.Kom. selaku Dosen Pembimbing Utama dan Kusbudiono, S.Si., M.Si. selaku Dosen Pembimbing Anggota yang telah meluangkan waktu, pikiran, dan perhatian dalam penulisan skripsi ini;
3. Prof. Drs Kusno, DEA., Ph.D. dan Dr. Fatekurohman, S.Si., M.Si. selaku dosen penguji atas saran-saran yang diberikan;
4. Kedua orang tua tercinta, Ibu Gisnawati dan Ayah Suharyono yang selalu memberikan dukungan dan doa begitu luar biasa;
5. Dosen dan karyawan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Jember;
6. Teman-teman satu angkatan yang telah memberikan banyak saran, dukungan dan kenangan.
7. Sahabat-sahabat terkasih yang selalu memberikan dukungan dimanapun berada.

Semoga bantuan, bimbingan, dan dorongan beliau dicatat sebagai amal baik oleh Allah SWT dan mendapat balasan yang sesuai dari-Nya. Selain itu, penulis juga

menerima segala kritik dan saran dari semua pihak demi kesempurnaan penyusunan tugas akhir ini. Akhirnya penulis berharap, semoga tugas akhir ini dapat bermanfaat.

Jember, Desember 2016 Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Halaman

DAFTAR TABEL

Halaman

BAB 1. PENDAHULUAN

1.1 Latar Belakang

Komunikasi merupakan hal yang dilakukan setiap saat untuk saling bertukar informasi. Informasi atau pesan dapat menjadi sesuatu yang sangat berharga dan dijaga kerahasiaannya. Semakin berkembangnya teknologi dalam dunia telekomunikasi, semakin banyak pula kejahatan untuk mengetahui informasi atau pesan yang bukan menjadi haknya. Maka dari itu sejalan dengan berkembangnya teknologi informasi, harus disertai dengan perkembangan pengamanan kerahasiaan informasi atau pesan tersebut.

Berbagai macam teknik digunakan untuk menjaga keamanan informasi atau pesan. Teknik yang dapat digunakan yaitu teknik kriptografi dan teknik steganografi. Teknik Kriptografi merupakan proses mengamankan pesan dengan mengubah ke dalam bentuk himpunan karakter acak yang tidak dapat dibaca (Scheiner, 1996). Informasi atau pesan tersebut harus tetap rahasia selama pengiriman dan harus tetap utuh pada saat penerimaan ditujuan. Terdapat banyak algoritma yang dapat digunakan dalam teknik kriptografi, diantaranya adalah Algoritma *Hill Cipher*.

Algoritma*Hill Cipher* termasuk kepada algoritma kriptografi klasik yang sangat sulit dipecahkan oleh kriptanalis apabila dilakukan hanya dengan mengetahui berkas *ciphertext* saja. Karena *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. Rosdiana (2015) membahas algoritma *Hill Cipher* dimana data yang dirahasiakan disembunyikan dalam media Citra digital.

Steganografi adalah ilmu dan seni menyembunyikan pesan rahasia ke dalam media digital dengan metode tertentu sehingga orang lain tidak menyadari ada sesuatu di dalam media digital tersebut (Munir, 2006). Steganografi berbeda dengan

Kriptografi, meskipun tujuannya sama yaitu untuk menyembunyikan pesan. Perbedaan yang antara keduanya yaitu kriptografi berfokus pada bagaimana melindungi informasi atau pesan agar tetap terjaga kerahasiaannya, sedangkan steganografi berfokus pada cara meminimalkan kecurigaan tentang keberadaan pesan rahasia.

Steganografi membutuhkan dua properti, yaitu media penampungan dan pesan rahasia. Media penampungan yang umum digunakan adalah media gambar, audio, dan video. Pesan yang disembunyikan dapat berupa sebuah teks, gambar, suara, maupun video. Media penampungan pesan yang digunakan dalam penelitian ini yaitu berupa *file* audio. Media penampungan berbentuk *audio* dapat berupa *Widebond Angular Vibration Experiment* (.wav), *Motion Picture Expert Group Audio Stream Layer* III (.mp3), dll.

File audio yang akan digunakan yaitu *file* audio dalam format .wav. Menurut Santoso, dkk (2014), wav merupakan bentuk format *file* yang fleksibel untuk menyimpan semua kombinasi *audio*, baik *rates* maupun *bitrates*. Hal ini menyebabkan format *file* dalam bentuk .wav sangat layak untuk menyimpan dan mengarsipkan rekaman asli. Pada penelitian sebelumnya (Wirawan, 2011) membahas tentang penerapan steganografi pada berkas audio .wav. Jenis pesan yang disisipkan adalah pesan gambar dengan format JPEG/JPG. Dalam penelitiannya, penyisipan pesan dengan gambar tidak berpengatuh terhadap ukuran berkas *audio*, akan tetapi berkas *audio* yang telah disisipi pesan (*stego*) tidak tahan terhadap kompresi, manipulasi amplitudo dan pemotongan audio.

Berbagai macam metode dapat digunakan dalam hal penyembunyian pesan tersebut. Metode yang akan dilakukan dalam steganografi pada penelitian ini adalah metode *Least Significant Bit* (LSB), yaitu metode penyembunyian informasi atau pesan dengan mengganti bit terakhir pada data pesan dengan bit paling rendah dalam media *audio*.

Pada penelitian ini penulis akan melakukan penyembunyian pesan terenkripsi dari kriptografi *Hill Cipher* dengan steganografi metode *Least Signifikan Bit* pada *file*

audio. Penerapan kriptografi dan steganografi ini diharapkan mampu meningkatkan keamanan data teks yang disembunyikan karena memanfaatkan dua teknik pengamanan data.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diurakan di atas, rumusan masalah yang dikemukakan adalah:

a. Bagaimana menerapkan steganografi pada *file audio* .wav dengan metode *Least Significant Bit* (LSB) terhadap pesan terenskripsi dengan algoritma *Hill Cipher*?
b. Bagaimana pengaruh variasi ukuran *file audio* dan ukuran (panjang) pesan terhadap waktu eksekusi?
c. Bagaimana kualitas suara yang dihasilkan oleh *file audio* setelah disisipkan pesan?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

a. Menyisipkan pesan teks ke dalam *file audio* dengan menggunakan metode *Least Significant Bit*;
b. Mengetahui pengaruh variasi ukuran *file audio* dan ukuran (panjang) pesan terhadap waktu eksekusi;
c. Menguji kualitas suara yang dihasilkan oleh *file audio* setelah disisipkan pesan.

1.4 Batasan Masalah

Dalam penelitian tugas akhir ini diberlakukan beberapa batasan masalah, diantaranya adalah:

a. Pesan rahasia yang disisipkan berupa *file* jenis teks;
b. Media penampung pesannya adalah *file audio* berformat .wav;
c. Untuk *file audio* dengan *bitrate* 128;
d. Kode *Hill Cipher* yang digunakan yaitu ASCII *Printable Characters*dari 32 sampai 127, yaitu sebanyak 95 karakter.

1.5 Mafaat Penelitian

Manfaat penelitian tugas akhir ini adalah meningkatkan keamanan informasi atau pesan dengan mengenkripsi dan mendekripsi pesan teks menggunakan algoritma *Hill Cipher* dan meminimalkan kecurigaan tentang pesan dengan menyisipkan pesan tersebut dalam *file audio* dengan steganografi menggunakan metode LSB sehingga hal tersebut dapat dijadikan solusi dalam hal pengiriman pesan rahasia dengan aman.

BAB 2. TINJAUAN PUSTAKA

2.1 Kriptografi

Kriptografi (*cryptography*) berasa dari bahasa Yunani yaitu "*crypptos*" yang artinya "*secret*" atau rahasia dan "*graphein*" yang artinya "*writing* atau tulisan. Jadi kriptografi memiliki arti yaitu "*secret writing*" atau tulisan rahasia (Ariyus, 2008). Kriptografi secara umum adalah ilmu sekaligus seni untuk menjaga kerahasiaan pesan (data atau informasi) dengan cara menyamarkan menjadi bentuk tersandi yang tidak bermakna. Pesan yang dirahasiakan dinamakan *Plaintext*, sedangkan pesan hasil penyamaran dinamakan *Ciphertext*. Proses penyamaran dari *plaintext* ke *ciphertext* disebut dengan enkripsi (*encryption*) dan proses kebalikan dari *ciphertext* ke *plaintext* disebut dengan dekripsi (*decryption*).

Gambar 2.1 Proses Enkripsi dan Dekripsi

2.1.1 Terminologi Kriptografi

Di dalam kriptografi sering ditemukan berbagai istilah atau *terminology*. Beberapa istilah yang harus diketahui adalah sebagai berikut (Schneiner, 1996):

a. Pengiriman dan Penerimaan pesan

   Seorang pengirim pesan (*sender*) ingin mengirim pesan kepada seorang penerima pesan (*receiver*). Pengirim menginginkan pesan dapat dikirim secara aman, yaitu ia yakin bahwa pihak lain tidak dapat membaca isi pesan.

b. Pesan, *Plaintext*, dan *Ciphertext*

Pesan adalah data atau informasi yang dapat dimengerti maknanya. Nama lain untuk pesan adalah *Plaintext* (Plainteks). Agar pesan tidak dapat dimengertimaknanya oleh pihak lain, maka pesan disandikan ke bentuk lain. Bentuk pesan yang tersadi disebut *Ciphertext* (Ciperteks).

c. Enkripsi dan Dekripsi

Proses menyandikan *plaintext* menjadi *ciphertext* disebut enkripsi (*encryption*). Sedangkan proses mengembalikan *ciphertext* menjadi *plaintext* disebut deripsi (*decryption*).

d. *Cipher* dan Kunci

Algoritma kriptografi disebut juga *cipher*, yaitu aturan untuk enkripsi dan dekripsi, atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Beberapa *cipher* memerlukan algoritma yang berbeda untuk enkripsidan dekripsi. Konsep matematis yang mendasari algoritma kriptografi adalah relasi antara dua buah himpunan yang berisi eleme-elemen plainteks dan himpunan yang berisi cipherteks. Enkripsi dan dekripsi merupakan fungsi yang memetakan elemen-elemn antara dua hipunan tersebut. Misal P menytakan plainteks dan C menyatakan cipherteks, maka:

$E(P) = C \rightarrow$ fungsi enkripsi E memetakan P ke C

$D(C) = P \rightarrow$ fungsi dekrisi D memetakan C ke P

Karena proses enkripsi kemudian dekripsi mengembalikan pesan ke pesan asal, maka persamaan $D(E(P)) = P$ harus benar.

Kriptografi mengatasi masalah keamanan data dengan menggunakan kunci, yang dalam hal ini algoritma tidak dirahasiakan lagi, tetapi kunci harus tetap dijaga kerahasiaannya. Kunci (*key*) adalah parameter yang digunakan untuk transformasi enkripsi dan dekripsi. Kunci biasanya berupa *string* atau deretan .

2.1.2 Tujuan Kriptografi

Dari paparan awal dapat dirangkum bahwa kriptografi bertujuan untuk memberi layanan keamanan. Yang dimaksud keamanan adalah (Rinaldi, 2006):

a. Kerahasiaan (*confidentiality*)

   Adalah layanan yang bertujuan untuk menjaga agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak.

b. Integritas Data (*data integrity*)

   Adalah layanan yang menjamin bahwa pesan masih asli atau beum pernah dimanipulasi selama pengiriman.

c. Otentikasi (*authentication*)

   Otentikasi (*authentication*) adalah layanan yang berhubung dengan identfikasi, baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user authentication* atau *entify authentication*) maupun mengidentifikasi kebenaran sumber pesan (*data origin authentication*).

d. Penyangkalan (*non-repudiation*)

   Penyangkapan (*noni-repudiation*) adalah layanan untuk mencegah entitas yang berkomunikasi melakukan penyangkalan, yaitu pengirim pesan menyangkal melakukan pengiriman atau penerima pesan menyangkal telah menerima pesan.

2.1.3 Algoritma Kriptografi

Algoritma kriptografi disebut juga *cipher* yaitu aturan untuk *enchipering* dan *dechipering*, atau fungsi yang digunakan untuk enkripsi dan dekripsi. Beberapa *cipher* memerlukan algoritma yang berbeda untuk *enchipering* dan *dechipering*. Keamanan algoritma kriptografi sering diukur dari banyaknya kerja yang dibutuhkan untuk memecahkan ciperteks menjadi plainteks tanpa mengetahui kunci yang digunakan. Apabila semakin banyak proses yang diperlukan, maka semakin kuat algoritma tersebut dan semakin aman digunakan untuk menyandikan pesan (Satria, 2009).

2.2 Algoritma *Hill Cipher*

Algoritma *Hill Cipher* adalah suatu fungsi matematis yang digunakan untuk melakukan enkripsi dan dekripsi. Ada dua macam algoritma kriptografi, yaitu

algoritma simetris (*symmetric algorithm*) dan algoritma asimetris (*asymmetric algorithms*). *Hill Cipher* merupakan *poly alphabetic cipher* dapat dikategorikan sebagai *block cipher*, karena teks yang akan diproses dibagi menjadi blok-blok dengan ukuran tertentu. Setiap karakter dalam satu blok akan saling mempengaruhi karakter lainya dalam proses enkripsi dan dekripsinya, sehingga karakter yang sama tidak dipetakan menjadi karakter yang sama pula (Wiyanarko, 2009).

2.2.1 Sejarah *Hill Cipher*

Sejak kekaisaran Romawi, kriptosistem yang lebih rumit dikembangkan oleh orang seperti ahli Matematika Italia Leon Battista Alberti (lahir pada tahun 1404), Matematikawan Jerman Johannes Trithernius (lahir pada tahun 1492), seorang *kriptographer* dan diplomat Perancs Blaise de Vigenere (1523-1596), Lester S. Hill, yang menemukan *Hill Cipher* pada tahun 1929. *Hill Cipher* merupakan jenis laindari *polygraphic cipher*. Sandi ini mengekripsi suatu string huruf menjadi bentuk string yang lain dengan panjang yang sama. Teknik *hil Cipher* dikembangkan oleh Leter S. Hill pada Hunter Collage dan dipublikasikan pada *Americian Mathematical Monthly*, *Volume 36*, *Issue 6* (Juni-Juli, 1929) halaman 302-312.

Hill Cipher menggunakan matriks untuk mentransformasikan *string* berupa blok huruf. Berdasarkan pada aljabar linier dan sandi *Vigenere*, *Hill Cipher* merupakan *block cipher*. *Hill Cipher* tidak mengganti setiap abjad yang sama pada *plaintext* dengan abjad lainnya yang sama pada *chipertext* karena menggunakan perkalian matriks pada dasar enkripsi dan dekripsinya. *Hill Cipher* termasuk pada algoritma kriptografi klasik yng sangat sulit dipecahkan oleh kriptanalis apabila dilakukan hanya dengan mengetahui berkas *chipertext* saja. Namun, teknik ini dapat dipecahkan dengan cukup mudah apabila kriptanalis memiliki berkas *chipertext* dan potongan berkas *plaintext*. Teknik kriptanalis ini disebut *known-plaintext attack* (Widyanarko, 2009).

2.2.2 Dasar Teknik *Hill Cipher*

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 *Hil Cipher* adalah matriks n x $n$dengan n juga merupakan ukuran blok. Jika kunci disebut dengan K, maka K adalah sebagai berikut (Widyanarko, 2009):

$$K = \begin{bmatrix} k_{11} & k_{12} & \dots & k_{1n} \\ k_{21} & k_{22} & \dots & k_{2n} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nn} \end{bmatrix}$$

Matriks K yang menjadi kunci harus merupakan matriks yang *invertible*, yaitu memiliki *inverse* K^{-1}, sehingga:

$$K \,.\, K^{-1} = \mathrm{I}$$

Kunci harus memiliki invers karena matriks K^{-1} tersebut adalah kunci yang digunakan untuk melakukan dekripsi.

2.2.3 Enkripsi pada *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* terlebih dahulu dikonversi menjadi angka. Secara sistematis, proses enkripsi pada *Hill Cipher* adalah:

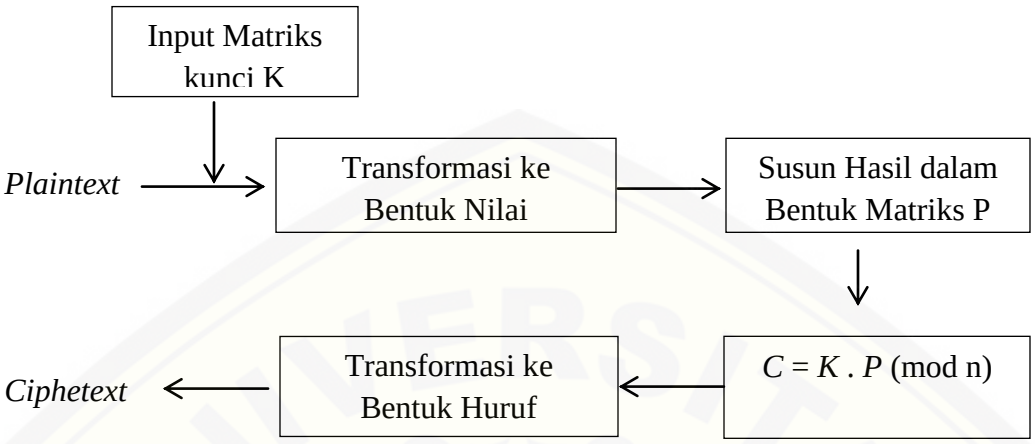
$$C = K.P \qquad (2.1)$$

dengan:

C = *Chipertext*

K = Kunci

P = *Plaintext*



Gambar 2.2. Ilustrasi Proses Enkripsi *Hill Cipher*

Proses enkripsi pada *hill cipher* dilakukan per blok plainteks. Ukuran blok tersebut sama dengan ukuran matriks kunci. Sebelum membagi teks menjadi deretan blok-blok, plainteks terlebih dahulu dikonversi menjadi angka, masing-masing sehingga A=0, B=1, hingga Z=25.

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

Secara matematis, proses enkripsi pada *hill cipher* adalah:

$C = K.P$

C =*Ciphertext*

K = Kunci

P =*Plaintext*

Contoh kasus misalnya ingin menyembunyikan sebuah pesan berisi nama ABDUL HALIM dengan kunci matriks ukuran 2 x 2, $K = \begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix}$, kemudian bagi plainteks sesuai dengan ukuran matriks.

Blok I	Blok II	Blok III	Blok IV	Blok IV
A B	D U	L H	A L	I M
0 1	3 20	11 7	0 11	8 12

Blok I

$$\begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 * 0 + 3 * 1 \\ 3 * 0 + 2 * 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Setelah itu hasil di mod 26 seperti berikut:

3 mod 26 = 3

2 mod 26 = 2

Hasilnya 3 dan 2 atau D dan C.

Blok II

$$\begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 3 \\ 20 \end{bmatrix} = \begin{bmatrix} 5 * 3 + 3 * 20 \\ 3 * 3 + 2 * 20 \end{bmatrix} = \begin{bmatrix} 75 \\ 49 \end{bmatrix}$$

Setelah itu hasil di mod 26 seperti berikut:

75 mod 26 = 23

49 mod 26 = 23

Hasilnya 23 dan 23 atau X dan X.

Blok III

$$\begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} 11 \\ 7 \end{bmatrix} = \begin{bmatrix} 5 * 11 + 3 * 7 \\ 3 * 11 + 2 * 7 \end{bmatrix} = \begin{bmatrix} 76 \\ 47 \end{bmatrix}$$

Setelah itu hasil di mod 26 seperti berikut:

76 mod 26 = 24

47 mod 26 = 21

Hasilnya 24 dan 21 atau Y dan V.

Blok IV

$$\begin{bmatrix}5 & 3\\3 & 2\end{bmatrix}\begin{bmatrix}0\\11\end{bmatrix}=\begin{bmatrix}5*0+3*11\\3*0+2*11\end{bmatrix}=\begin{bmatrix}33\\22\end{bmatrix}$$

Setelah itu hasil di mod 26 seperti berikut:

33 mod 26 = 7

22 mod 26 = 22

Hasilnya 24 dan 21 atau H dan W.

Blok V

$$\begin{bmatrix}5 & 3\\3 & 2\end{bmatrix}\begin{bmatrix}8\\12\end{bmatrix}=\begin{bmatrix}5*8+3*12\\3*8+2*12\end{bmatrix}=\begin{bmatrix}76\\48\end{bmatrix}$$

Setelah itu hasil di mod 26 seperti berikut:

76 mod 26 = 24

48 mod 26 = 22

Hasilnya 24 dan 21 atau Y dan W.

maka hasil *ciphertext* dari ABDUL HALIM adalah DCXXYVHWYW (Halim, Abdul. 2013).

2.2.4 Dekripsi pada *Hill Cipher*

Proses dekripsi pada *Hill Cipher* pada dasarnya sama dengan proses enkripsinya. Namun, matriks kunci harus dibalik (invers) terlebih dahulu. Secara matematis, proses dekripsi pada *Hill Cipher* dapat diturunkan pada persamaan:

$$C = K\,.\,P$$

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

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

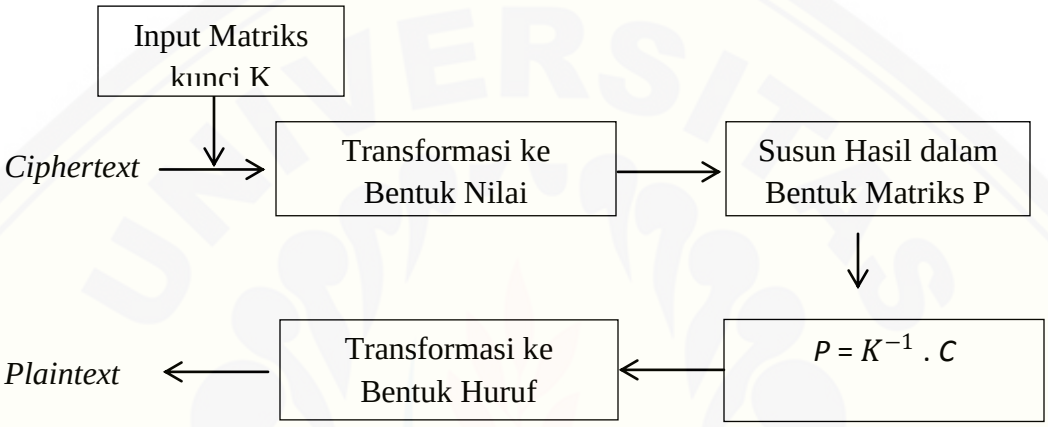
$$P = K^{-1}\,.\,C$$

Sehingga diperoleh persamaan dekripsinya:

$P = K^{-1} . C$ (2.2)

Dimana untuk menentukan K^{-1} dengan menggunakan rumus:

$$K^{-1} = \frac{1}{|K|} Adj\ (K)$$



Gambar 2.3. Ilustrasi Proses Dekripsi *Hill Cipher*

Proses dekripsi diawali dengan menghitung invers dari matriks *K*. Maka proses dekripsi sebagai berikut:

$K = \begin{bmatrix} 5 & 3 \\ 3 & 2 \end{bmatrix} \text{mod } 26$

Det (K) = 5 x 2 – 3 x 3 =1

Maka untuk mencari K^{-1} adalah:

$K^{-1} = \frac{1}{K} Adj\ (K)$

$K^{-1} = \frac{1}{1} \begin{bmatrix} 2 & -3 \\ -3 & 5 \end{bmatrix}$

$K^{-1} = \begin{bmatrix} 2 & -3 \\ -3 & 5 \end{bmatrix}$

Setiap yang bernilai negatif ditambah 26 agar nilai tetap positif, ini digunakan karena yang digunakan yaitu 0-25, sehingga $K^{-1} = \begin{bmatrix} 2 & 23 \\ 23 & 5 \end{bmatrix}$. Hasil dari enkripsi tersebut, yaitu DCXXYVHWYW akan didekripsi dengan cara membagi dengan blok ada hasil *ciphertext* yang sudah ada. Proses dekripsinya asama dengan proses enkripsi, namun dengan kunci yang berbeda. Setelah semua blok selesai didekripsi, maka didapatlah plainteks semula, yaitu ABDULHALIM (Halim, Abdul. 2013).

Namun demikian, pesan rahasia yang ingin disampaikan pengirim kepada penerimanya tentunya tidak hanya berupa huruf saja, melainkan juga dapat berupa spasi, angka, atau simbol yang lain. Jika pada *Hill Cipher* yang biasa digunakan hanya diterapkan pada 26 karakter saja seperti pada penjelasan di atas, maka pada pada penelitian ini mengunakan ASCII *Printable Characters* yaitu kode dengan karakter sebanyak 95.

2.3 ASCII *Printable Characters*.

Kode ASCII merupakan suatu standar internasional dalam kode huruf dan simbol yang bersifat universal. Kode ASCII digunakan oleh komputer dan alat komunikasi lainnya untuk menunjukkan teks (Rachmawanto, 2010).

ASCII *Printable Characters* merupakan karakter kode dari 32 – 127 dari kode ASCII, dimana berisi huruf, simbol, tanda baca yang dapat ditemukan pada *keyboard*. Kode ASCII *Printable Characters* dapat dilihat pada tabel 2.1 berikut.

Tabel 2.1. ASCII *Printable*

Co-de	*Sym-bol*	*Binary*	Urutan moodulo	*Co-de*	*Sym-bol*	*Binary*	Urutan moodulo
32	space	00100000	0	80	P	01010000	48
33	!	00100001	1	81	Q	01010001	49
34	“	00100010	2	82	R	01010010	50

35	#	00100011	3	83	S	01010011	51
36	$	00100100	4	84	T	01010100	52
37	%	00100101	5	85	U	01010101	53
38	&	00100110	6	86	V	01010110	54
39	‘	00100111	7	87	W	01010111	55
40	(	00101000	8	88	X	01011000	56
41	)	00101001	9	89	Y	01011001	57
42	*	00101010	10	90	Z	01011010	58
43	+	00101011	11	91	[	01011011	59
44	,	00101100	12	92	\	01011100	60
45	-	00101101	13	93	]	01011101	61
46	.	00101110	14	94	^	01011110	62
47	/	00101111	15	95	_	01011111	63
48	0	00110000	16	96	`	01100000	64
49	1	00110001	17	97	a	01100001	65
50	2	00110010	18	98	b	01100010	66
51	3	00110011	19	99	c	01100011	67
52	4	00110100	20	100	d	01100100	68
53	5	00110101	21	101	e	01100101	69
54	6	00110110	22	102	f	01100110	70
55	7	00110111	23	103	g	01100111	71
56	8	00111000	24	104	h	01101000	72
57	9	00111001	25	105	i	01101001	73
58	:	00111010	26	106	j	01101010	74
59	;	00111011	27	107	k	01101011	75
60	<	00111100	28	108	l	01101100	76
61	=	00111101	29	109	m	01101101	77
62	>	00111110	30	110	n	01101110	78

63	?	00111111	31	111	o	01101111	79
64	@	01000000	32	112	p	01110000	80
65	A	01000001	33	113	q	01110001	81
66	B	01000010	34	114	r	01110010	82
67	C	01000011	35	115	s	01110011	83
68	D	01000100	36	116	t	01110100	84
69	E	01000101	37	117	u	01110101	85
70	F	01000110	38	118	v	01110110	86
71	G	01000111	39	119	w	01110111	87
72	H	01001000	40	120	x	01111000	88
73	I	01001001	41	121	y	01111001	89
74	J	01001010	42	122	z	01111010	90
75	K	01001011	43	123	{	01111011	91
76	L	01001100	44	124	|	01111100	92
77	M	01001101	45	125	}	01111101	93
78	N	01001110	46	126	~	01111110	94
79	O	01010000	47	127	DEL	01111111	95

Tabel 2.1 Kode ASCII *Printable Characters*

2.4 Steganografi

Steganografi berasal dari bahasa Yunani yang terdiri dari dua kata, yaitu *Steganos* dan *Graphia*. *Steganos* berarti tersembunyi dan *Graphia* artinya tulisan. Dengan demikian, steganografi adalah ilmu atau seni untuk menyembunyikan pesan. Pesan tersebut disembunyikan dengan tujuan agar tidak diketahui oleh orang lain. Orang yang dapat mengetahui pesan tersebut adalah dirinya sendiri dan orang lain yang dikehendaki. Steganografi membahas cara untuk menyamarkan dan

menyembunyikan pesan rahasia sehingga pesan tersebut tampak seperti informasi lainnya (Chandraleka, 2009).

Steganografi membutuhkan dua properti, yaitu wadah penampungan dan data rahasia yang akan disembunyikan. Steganografi digital menggunakan media digital sebagai wadah penampung, misalnya menggunakan citra atau gambar, suara atau *audio*, teks, dan video. Data rahasia yang disembunyikan juga dapat berupa citra atau gambar, suara atau *audio*, teks, dan video (Munir, 2006).

Steganografi berbeda dengan kriptografi. Letak perbedaannya adalah hasil keluarannya. Hasil dari kriptografi biasanya berupa data yang berbeda dari bentuk aslinya dan biasanya seolah-olah tidak teratur dan dapat dikembalikan ke bentuk semula. Sedangkan steganografi memiliki bentuk persepsi yang sama dengan bentuk aslinya, tentunya persepsi disini oleh manusia, tetapi tidak oleh komputer atau perangkat lainnya (Bender, 1996).

2.4.1 Sejarah Steganografi

Penggunaan steganografi setelah digunakan beradab-abad yang lalu, bahkan sebelum istilah steganografi itu sendiri muncul. Berikut adalah contoh penggunaan steganografi di masa lalu (Munir, 2006):

a. Steganografi sudah dikenal oleh bangsa Yunani. Herodatus, penguasa Yunani, mengirim pesan rahasia dengan menggunakan kepala budak atau prajurit sebagai media. Dalam hal ini, rambut budak dibotaki, lalu pesan rahasia ditulis pada kepala budak. Ketika rambut budak tumbuh, budak tersebut diutus untuk membawa pesan rahasia di balik rambutnya.
b. Bangsa Romawi mengenal steganografi dengan menggunakan tinta tak-tampak (*invisible ink*) untuk menuliskan pesan. Tinta tersebut dibuat dari campuran sari buah, susu, dan cuka. Jika tinta digunakan untuk menulis, maka tulisannya tidak tampak. Tulisan di atas kertas dapat dibaca dengan cara memanaskan kertas tersebut.

c. Metode lain yang digunakan oleh masyarakat Yunani kuno adalah dengan menggunakan lilin sebagai penyembunyian pesan mereka. Pesan dituliskan pada suatu lembaran, dan lembaran tersebut akan ditutup dengan lilin untuk menyembunyikan pesan yang telah tertulis. Pihak penerima kemudian akan menghilangkalilin dari lembaran tersebut untuk melihat pesan yang disampaikan oleh pihak pengirim.

2.4.2 Media Steganografi

Beberapa contoh media penyisipan pesan rahasia yang digunakan dalam teknik steganografi adalah (Alatas, 2009):

a. Teks

Dalam algoritma stegnografi yang menggunakan teks sebagai media penyisipannya biasanya digunakan teknik NLP (*Natural Language Processing*) sehingga teks yang telah disisipi pesan rahasia tidk akan mencurigakan untuk orang yang melihatnya.

b. *Audio*

Format ini pun sering dipilih karena biasanya berkas dengan format ini berukuran relatif besar, sehingga dapat menampung pesan rahasia dalam jumlah yang besar pula.

c. Citra

Format ini pun paling sering digunakan karena merupakan salah satu format file yang sering dipertukarkan dalam dunia internet. Alasan lainnya adalah banyaknya tersedia algoritma steganografi untuk media penampung yang berupa citra.

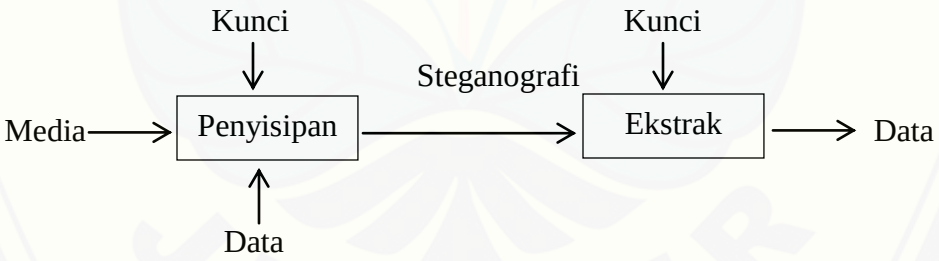
d. Video

Format ini merupakan format dengan ukuran file yag relatif sangat besar namun jarang digunakan arena ukurannya yang terlalu besar sehingga mengurangi kepraktisannya dan juga kurangnya algoritma yang mendukung format ini.

2.4.3 Konsep Steganografi

Konsep dari steganografi adalam menyembunyikan pesan dalam media lain, sehingga pesan tidak dapat diterjemahkan secara langsung. Secara umum terdapat dua proses dalam steganografi, yaitu proses penyisipan (*Embedding* / *encoding*) untuk menyembunyikan pesan dan ekstraksi (*extraction* / *decoding*) untuk mengekstraksi pesan yang disembunyikan. Pesan dapat berupa *plaintext*, *ciphertext*, citra atau apapun yang dapat ditempelkan ke dalam *bit-stream*.

Embedding merupakan proses menyisipkan pesan ke dalam *file* yang belum dimodifikasi, yang disebut media *cover* (*file audio*). Kemudian media *cover* dan pesan yang ditempelkan membuat media *stego* (*stego audio*). *Extraction* adalah proses menguraikan pesan yang tersembunyi dalam media *stego*. Suatu kunci khusus (*stego key*) juga dapat digunakan secara tersembunyi pada saat penguraian selanjutnya dari pesan. Ringkasnya, steganografi adalah teknik menanamkan *embedde message* pada suatu *file audio*, dimana hasilnya berupa *stego audio*. Pihak yang terkait dengan steganografi antara lain *embeddor*, *extractor*, dan *stegoanalyst*. Skema penyisipan dan ekstraksi dalam steganografi diperlihatkan pada Gambar 2.4:



Gambar 2.4. Skema penyisipan dan ekstraksi dalam steganografi.

2.5 Perbedaan Steganografi dengan Kriptografi

Steganografi merupakan pelengkap dri kriptografi, bukan pengganti. Sebab, dari kedua disiplin ilmu tersebut dapat digunakan konsep secara bersamaan ataupun secara terpisah. Seperti halnya pesan yang terenkripsi disembunyikan ke dalam suatu media audio. Proses enkripsi merupakan teknik dalam ilmu kriptografi, sedangkan menyembunyikan pesan yang telah terenkripsi merupakan teknik ilmu steganografi.

Dari segi tujuan, kriptografi bertujuan untuk menyembunyikan isi (*content*) pesan agar pesan tidak dapat dibaca. Sedangkan steganografi bertujuan untuk menyembunyikan keberadaan (*existence*) pesan untuk menghindari kecurigaan (*conspicous*) (Prasetyo, 2010).

2.6 *File Audio*

Audio atau suara adalah fenomena fisik yang dihasilkan oleh getaran suara benda yang berupa sinyal analog dengan amplitudo yang berubah secara kontinu terhadap waktu yang disebut frekuensi. Selama bergetar, perbedaan tekanan terjadi di udara sekitarnya. Pola osilasi yang terjadi dinamakan sebagai gelombang. Gelombang mempunyai pola sama yang berulang pada interval tertentu, yang disebut sebagai periode. Contoh suara periodik adalah instrumen musik, nyanyian burung, sedangkan contoh nonperiodik adalah batuk, percikan ombak dan lain-lain (Binanto, 2010).

Suatu format *file audio* adalah format *file* untuk menyimpan data audio digital pada sistem komputer. Data ini dapat disimpan tanpa dikompresi, atau dikompresi untuk mengurangi ukuran *file*. Ini bisa menjadi sebuah *raw bitsream*, tetapi biasanya merupakan sebuah *container* format atau format data *audio* dengan lapisan penyimpanan yang telah ditetapkan.

Penting untuk membedakan antara format *file* dan *codec audio*. *Codec* melakukan *encoding* dan *decoding* data *audio* mentah, sementara data itu sendiri disimpan dalam *file* dengan format *file audio* tertentu. Meskipun sebagian besar format *fileaudio* hanya mendukung satu jenis data *audio* (dibuat dengan *coder audio*), *multimedia container format* (seperti Matroska atau AVI) dapat mendukung beberapa jenis data *audio* dan *video*. Ada tiga kelompok utama format *fileaudio*:

a. Format *audio* yang terkompresi, seperti WAV, AIFF, AU atau raw *header-less* PCM.
b. Format *audio* dengan kompresi *lossless*, seperti FLAC, *Monkey's Audio* (yang berekstensi APE), WavPack (yang berekstensi WV), TTA, ATRAC *Advance*

Lossless, *Apple Lossless* yang berekstensi m4a), MPEG-4 ALS, MPEG-4 DST, *Windows Media Audio Lossless* (WMA *Lossless*), dan *Shorten* (SHN).

c. Format *audio* dengan kompresi *lossy*, seperti MP3, *Vorbis*, *Musepack*, AAC, ATRAC dan *Windows Media Audio Lossy* (WMA Lossy).

2.6.1 Membaca Sinyal *Audio*

Sinyal adalah suatu isyarat atau pemberitahuan yang dapat ditangkap oleh indera untuk kepentingan penyampaian, petunjuk, atau informasi. Sinyal merupakan sebuah fungsi yang berisi informasi mengenai keadaan tingkah laku dari sebuah fungsi yang berisi informasi mengenai keadaan tingkah laku dari sebuah sistem secara fisik. Meskipun sinyal dapat dapat diwujudkan dalam beberapa cara, dalam berbagai kasus, informasi terdiri dari sebuah pola dari beberapa bentuk yang bervariasi. Sebagai contoh sinyal mengkin berbentuk sebuah pola dari banyak variasi waktu atau sebagian saja.

Secara matematis, sinyal merupakan fungsi dari satu atau lebih variabel yang tidak bergantung (*independent variable*). Untuk analisis, sebuah sinyal dapat didefinisikan sebagai fungsi matematika yang secara umum dapat ditulis sebagai berikut:

$$y = f(x)$$

dengan x adalah variabel atau peubah yang independen (nilainya tidak bergantung pada peubah lain) dan y (sinyal) merupakan peubah yang tidak independen (dalam hal ini y bergantung pada nilai x. Peubah independen menentukan domain (daerah asal) dari sinyal, misal:

1. $y = \sin(\omega . t)$ adalah suatu fungsi dengan variabel dalam domain waktu (*time-domain*), t merupakan sinyal yang berubah terhadap waktu (*time-signal*).
2. $x(\omega) = 1/(-m\omega^2 + ic\omega + k)$ merupakan sinyal yang mempunyai domain frekuensi yaitu ω atau disebut *frequency-domain signal*.

2.6.2*Audio Digital* WAV

Format *file audio* tanpa proses kompresi yang paling sering ditemui adalah PCM (*Pulse Code Modulation*), yang biasanya tersimpan sebagai*file* .wav di dalam *Windows* dan sebagai .aiff di dalam *Mac* OS.WAV adalah bentuk format *file* yang fleksibel untuk menyimpan semua kombinasi *audio* baik*rates* maupun*bitrates*. Hal ini menyebabkan format *file* dalam bentuk .wav sangat layak untuk menyimpan dan mengarsipkan rekaman asli. Untuk format *audio lossless*, akan dibutuhkan lebih banyak proses pada saat direkam, tetapi akan sangat efisien dalam hal penggunaan memori. WAV, seperti halnya seluruh format *file* yang tidak dikompres, akan meng-*encoding*-kan semua suara, baik suara yang kompleks maupun tanpa suara, dengan jumlah bit yang sama setiap waktunya. Contohnya: sebuah *file* yang menyimpan rekaman dari orkestra selama satu menit akan sama besar dengan *file* yang menyimpan satu menit keadaan diam tanpa suara apabila keduanya disimpan dalam bentuk format WAV. Apabila *file* di *encoding* dengan format *file lossless*, maka dengan contoh yang sama, *file* pertama akan menempati lebh sedikit memori, sedangkan *file* kedua sangat sedikit menggunakan memori. Namun bagaimanapun juga, untuk meng-*encodingfile* ke dalam format *file lossless* akan membutuhkan waktu yang jauh lebih lama dibandingkan dengan format *file* yang tidak dikompres sama sekali, yakni dalam format WAV.

Pada *platform Windows*, ekstensi file yang paling umum untuk *file audio* adalah ".wav". MATLAB dapat membaca gelombang *file audio* seperti melalui perintah "wavread". Sebagai contoh MATLAB dapat membaca *file* gelombang *audio* yang hanya berbunyi "beep" yang diberi nama *file* "sinus.wav". *File audio* tersebut akan dirubah menjadi Biner agar pesan dapat dengan mudah disisipkan. Adapun langkah-langkah untuk mengubah sinyal *audio* menjadi Biner adalah sebagai berikut:

d. Membacasinyal*audio*

Untuk membaca sinyal *audio* pada MATLAB dapat dilakukan dengan menuliskan perintah "[y f fs]=wavread('nama file.wav')" dimana "y" merupakan kolom yang

berisi *sample* dari sinyal *audio*, "f" merupakan banyaknya *frame* dan "fs" merupakan *sample rate*.

e. Mengambil nilai integer asli dari sinyal *audio* sesuai dengan panjang pesan yang dipakai

   Setelah mendapatkan sinya *audio*, selanjutnya dapat diambil nilai integer asli dari sinyal *audio* untuk resolusi 8-bit, y (nilai yang diperoleh oleh wavread) dikalikan dengan 128 dengan menuliskan perintah "y(1 : panjang pesan)*128". Pada langkah ini akan diperoleh nilai integer asli dari *audio* dalam bentuk Desimal.

f. Mengubah Desimal menjadi Biner

   Pada langkah sebelumnya telah didapat nilai Desimal dari *audio*, maka pada langkah ini nilai Desimal tersebut akan dirubah menjadi Biner. Proses perubahan Desimal menjadi Biner ini dapat dilakukan dengan menuliskan perintah "dec2bin". Setelah didapatkan Biner dari *audio*, maka *audio* tersebut telah siap untuk disisipkan pesan.

2.7 *Least Significant Bit* (LSB)

Metode LSB (*Least Significant Bit*) ini merupakan metode yang komputasinya tidak terlalu kompleks dan pesan yang disembunyikan cukup aman serta dapt menyimpan pesan dengan ukuran relatif besar. Metode ini memodifikasi nilai yang paling kurang signifikan dari jumlah bit dalam1 *byte file carrier*. Bit yang memiliki signifikan paling tinggi adalah numerik yang memiliki nilai tertinggi (misal, 2^7 = 128), artinya bila terjadi perubahan pada bit ini akan menghasilkan perubahan yang sangat signifikan. Bit yang memiliki signifikansi paling rendah adalah numerik yang memiliki nilai terendah (misal, 2^0 = 1), artinya bila terjadi perubahan pada bit ini akan menghasilkan perubahan yang tidak terlalu signifikan. Sebagai contoh, akan dilakukan proses penyembunyian karakter 'G' (ASCII 71) pada berkas *carrier* yang berukuran 8 *byte*. *Least Significant Bit* dari *file carrier* ditandai dengan garis bawah. Berkas *carrier* dalam Biner dengan ukuran 8 *byte*:

‘10010101 00001101 11001001 10010110
00001111 11001011 10011111 00010000’

Karakter ‘G’ dalam bentuk dengan ukuran 1 *byte*:

‘01000111’

Kedelapan bit ini nantinya akan dimasukkan ke dalam *Least Significant Bit* dari tiap-tiap *byte* pada *file carrier* seperti berikut ini:

‘10010101 00001101 11001001 10010110
00001111 11001011 10011111 00010000’

Karakter ‘G’ dalam bentuk dengan ukuran 1 *byte*:

‘01000111’

Proses *Least Significant Bit Modification*:

‘10010100 00001101 11001000 10010110
00001110 11001011 10011111 00010001’

Least Significant Bitfile carrier yang berubah (ditunjukkan dengan karakter hitam tebal). Berdasarkan teori yang didapat adalah bahwa kemungkinan terjadinya perubahan bit adalah sekitar 50%, karena peluang perubahannya adalah antara 0 atau 1 dan denga mengubah *Least Significant Bit* maka ukuran dari *file* pembawa tidak akan berubah sehingga akan sulit untuk terdeteksi (Bender, 1996).

BAB 3. METODE PENELITIAN

3.1 Studi Literatur

Studi literatur dilakukan dengan melakukan pemahaman mengenai konsep dari kriptografi dan steganograf, serta *file audio* yang berfokus pada algoritma *Hill Cipher*, metode *Least Significant Bit* (LSB), dan *file audio* .wav. Literatur pendukung ini berupa jurnal, artikel, buku, dan sumber lainnya.

3.2 Tahap Identifikasi Masalah

Tahap ini dilakukan untuk penentuan hal penting, seperti pengumpulan data *file audio*, pengelompokan *file audio* berdasarkan kapasitas maksimum pesan yang dapat ditampung dalam *file audio* .wav, serta pengumpulan data *file* pesan teks yang akan disisipkan. Tahap ini digunakan sebagai dasar permasalahan yang akan dianalisis. Tahap ini juga merupakan tahap mengkaji dan membatasi masalah yang akan diimplementasikandalam sistem.

3.3 Tahap Implementasi

Ada 4 tahapan dalam implementasi aplikasi ini dengan mengkombinasian proses kriptografi yang terdiri dari tahap enkripsi dan dekripsi pesan teks dari *ciphertext* menjadi *plaintext*, serta proses steganografi yang terdiri dari tahap penyisipan pesan teks ke dalam *file audio* dan tahap ekstraksi *audio* stego sehingga menjadi*file audio* dan *ciphertext*.

Gambar 3.1 Skema Implementasi dan Steganografi

File audio sebagai media penampung dan *file* teks menjadi masukan pesan rahasia yang akan disisipkan. *File* teks terlebih dahulu melalui tahap enkripsi menghasilkan *ciphertext*, kemudian *ciphertext* disisipkan ke dalam *file audio*. Hasil dari proses tersebut adalah *file audio*-stego. Tahap ekstraksi digunakan untuk memisahkan antara *file audio* dan pesan teks yang disisipkan. Pesan teks yang dihasilkan masih dalam keadaan terenkripsi sehingga harus melalui tahap dekrispsi terlebih dahulu agar menghasilkan pesan rahasia yang dapat diketahui maknanya.

3.3.1 Tahap Enkripsi

Tahap enkripsi merupakan suatu proses untuk mengolah *plaintext* menjadi sebuah *ciphertext* yang tidak dapat diterjemahkan secara langsung. Pada proses enkripsi, sebelum membagi menjadi deretan blok-blok, pesan terlebih dahulu dikonversi ke dalam *code* pada tabel ASCII *Printable Characters*. Setelah dikonversi kemudian dibagi perblok sesuai ukuran matriks yang digunakan, dan tiap blok dienkripsi dengan kunci K seperti pada persamaan 2.1 dan akan menghasilkan *ciphertext*.

3.3.2 Tahap Penyisipan Pesan

Tahap penyisipan pesan dilakukan pada *fileaudio* wav dengan menggunakan metode *Least Significant Bit* (LSB). Penyisipan pesan dilakukan dengan mengganti *byte* terakhir pada *file audio* menggunakan metode LSB menghasilkan objek *stego*. Adapun langka-langkah dalam penyisipan pesan adalah sebagai berikut:

a. Meyiapkan kunci yang digunakan sebagai kunci enkripsi, pesan rahasia dalam bentuk *text*, dan *file audio* .wav sebagai media penampung.
b. Pesan *text* kemudian dienkripsi menggunakan algoritma *Hill Cipher* menghasilkan *ciphertext*.
c. Untuk mendapatkan *byte* homogen dari *audio*, dapat dilakukan langkah-langkah sesuai dengan Sub Sub-bab 2.6.2 sampai menghasilkan Biner dari *audio*.

d. Setelah mendapatkan Biner dari *audio*, penyisipan pesan dilakukan dengan mengganti bit terakhir pada *audio* menggunakan metode LSB. Setelah mengganti bit terakhir dari *audio* dengan bit-bit dari *ciphertext*,kemudian Biner yang telah disisipi *ciphertext* dirubah kembali menjadi Desimal.

e. Setelah dirubah kembai menjadi Desimal, kemudian kembai disimpan dengan menghasilkan *audio* yang telah disisipkan pesan namun masih dengan suara yang sama.

3.3.3 Tahap Ekstraksi Pesan

Untuk mengambil atau mengungkapkan pesan teks atau informasi yang telah disisipkan dalam *file stego* maka dibutuhkan proses pengekstrakan agar pesan teks atau informas tersebut dapat dikembalikan tanpa mengubah bit-bit dari *file audio* yang digunakan. Adapun langkah-langkahnya adalah sebagai berikut:

a. Menyiapkan *file audio* yang telah disisipi *ciphertext* (*stego audio*).
b. Melakukan langkah yang sama pada Sub Sub-bab 2.6.2 sampai menghasilkan Biner.
c. Setalah mendapatkan Biner dari *stego audio,* kemudian diambil bit paling akhir.
d. Kelompokkan menjadi 8 bit.
e. Setelah mengelompokkan menjadi 8 bit, ubah ke dalam bentuk Desimal, setelah itu diubah menjadi karakter sesuai kode ASCII untuk mendapatkan *ciphertext*nya.

3.3.4 Tahap Dekripsi

Proses dekripsi merupakan sistem untuk mengolah data acak (*ciphertext*) menjadi data awal (*plaitext*). Proses dekripsi pada dasarnya sama dengan proses enkripsinya, namun matriks kunci harus dibalik (invers) terlebih dahulu seperti pada persamaan 2.2.

3.4 Tahap Perancangan dan Pembuatan Program

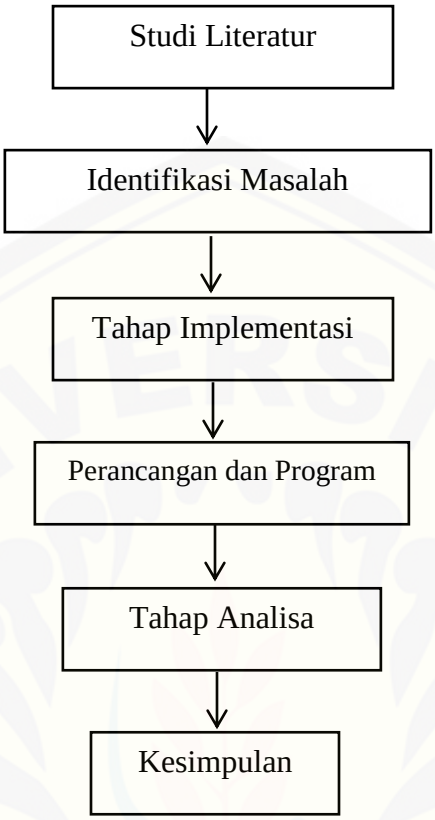
Tahap perancangan program menggunakan *software* MATLAB dan melakukan perancangan desain GUI untuk membuat tampilan layaknya sebuah aplikasi, seperti tata letak tombol-tombol untuk setiap proses yang dibutuhkan, serta tata letak *properties* pendukung lainnya.

Sedangkan Pembuatan program dilakukan berdasarkan konsep algoritma *Hill Cipher* untuk proses enkripsi dan dekripsi pesan teks dan *file audio* hasil steganografi dengan metode *Least Significant Bit*.

3.5 Tahap Analisa

Dalam tahap ini analisa difokuskan pada hal-hal berikut ini:

a. Melakukan proses enkripsi menggunakan algoitma *Hill Cipher* dengan kunci yang telah ditetapkan dan juga penerapan metode *Least Significant Bit* (LSB).
b. Menyiapkan *Sample* yang terdiri dari *plaintext*sisipan dan *file audio*.
c. Ukuran *file* teks yang ditentukan.
d. Genre musik dan ukuran *file audio*.
e. Kapasitas pesan teks yang mampu ditampung oleh *file audio* wav.
f. Perhitungan waktu eksekusi, meliputi: waktu proses enkripsi, waktu proses dekripsi, waktu penyisipan pesan ke dalam *file audio*, dan waktu ekstraksi *file* .wav *stego*.
g. Bagaimana suara *file* wav stego yang dihasilkan.



3.2 Skema Langkah – Langkah Penelitian

BAB 5. PENUTUP

5.1 Kesimpulan

Berdasarkan penelitian yang dilakukan, maka dapat diperoleh beberapa kesimpulan sebagai berikut:

a. Kriptografialgoritma *Hill Cipher* menggunakan matriks kunci pada proses enkripsi dan dekripsinya. Pesan yang dienkripsi dan didekripsi yaitu berupa pesan teks. Proses *encoding* atau penyisipan pesan teks ke dalam *file audio* menggunakan metode *Least Significant Bit*, yaitu dengan mengganti bit ke delapan atau bit yang paling tidak berarti pada *file audio* dengan bit pesan secara berurutan. Untuk dapat berhasil menyisipkan pesan ke dalam *file audio*, jumlah bit pesan harus sama dengan jumlah *byte* suatu *file audio* atau jumlah *byte* suatu *file audio* harus lebih banyak daripada jumlah bit pesan.
b. Ukuran pesan yang mampu ditampung *file audio* dipengarui oleh banyaknya atau panjang *bytefile audio*. Semakin banyak *byte* suatu *file audio* wav, maka kapasitas pesan yang mampu ditampung juga semakin banyak. Apabila panjang *bytefile audio* lebih kecil daripada panjang pesan, maka pesan tidak dapat disisipkan ke dalam *file audio*.
c. Waktu eksekusi penyisipan dan ekstraksi yang paling tercepat adalah pada *file* dengan ukuran *file* paling kecil dan yang paling lama adalah pada *file* dengan ukuran *file* paling besar.Adapun kualitas suara yang dihasilkan yaitu tidak berubah setelah disisipkan pesan.

5.2 Saran

Saran yang dapat diberikan untuk penelitian selanjutnya yaitu pengujian dilakukan dengan menggunakan jenis *audio* yang berbeda, misalkan mp3, midi, acc,

ataupun yang lainnya. Pesan yang dapat disisipkan tidak hanya berupa teks saja, tetapi dapat menyisipkan pesan berupa gambar, *audio*, ataupun video.

DAFTAR PUSTAKA

Alatas, P. 2009. *Implementasi Teknik Stegangrafi Dengan Metode LSB Pada Citra Digital*. Tugas Akhir, Universitas Gunadarma.

Ariyus, D. 2008. *Pengantar Ilmu Kriptografi Teori, Analisis, dan Implementasi*.Yogyakarta: C.V Andi Offset.

Bender. 1996. *Technique For Data Hiding*. IBM system Journal vol 35, No.3&4, Germany.

Binanto, I. 2010. *Multimedia Digital-Dasar Teori dan Pengembangannya*. Yogyakarta: Penerbit Andi

Chandraleka, H. 2009. *Mengamankan Data Pribadi ala Agen Rahasia*. Jakarta: Elex Media Komputindo.

Halim, A. 2013. *Implementasi Algoritma Hill Cipher dalam Penyandian Data*. Medan: STIMIK Budi Darma, Vol.IV, No.2 – Agustus 2013.

Prasetyo, F. 2010. *Steganografi Menggunakan Metode LSB dengan Software Matlab*. Skripsi. Jakarta: Universitas Islam Negri Syarif Hidayatullah.

Munir, R. 2006. *Kriptografi*. Bandung: Departemen Teknik Informatika, Institut Teknologi Bandung.

Rachmawanto, E.H. *Teknik Keamanan Data menggunakan Kriptografi dengan Algoritma Vernam Cipher dan Steganografi Metode End of File (EOF).* Tidak Diterbitkan. Skripsi. Semarang: Universitas Dian Nuswantoro.

Rosdiana, G.T. 2015. *Pengkodean Citra Digital Hasil Steganografi dengan Metode Least Significant Bit untuk Data Teks Terenkripsi Dengan Algoritma Hill Cipher*. Skripsi. Jember: Universitas Jember.

Santoso, S, dkk. 2014. *Steganografi Audio* (WAV) *Menggunakan Metode* LSB (*Least Significant Bit*). Universitas Budi Luhur Jakarta: *Jurnal Informatika* ISSN: 1978 – 8282, Vol.9 No.2 – Januari 2016.

Satria, E. 2009. *Studi Algoritma RIJNDEAL dalam Sistem Keamanan Data.* USU Repository, Medan.

Shneiner, B. 1996. *Apllied Cryptography, Protocols, Algorithms, and Source Code in C.* New York: A John Wiley&Sons, Inc.

Widyanarko, A. 2009. *Studi dan Analisis mengenai Hill Cipher Teknik Kriptanalisis dan Upaya Penanggulannya.*

Wirawan, S. 2011. *Implementasi Steganografi pada Berkas Audio WAV untuk Penyisipan Pesan Gambar Menggunakan Metode Low Bit Coding.* Tugas Akhir. Depok: Universitas Guna Dharma.