Steganografi Audio Digital (WAV) Dengan Teknik Penyisipan *Least Significant Bit* (LSB)

Siti Khomsah

Abstraksi

Steganografi merupakan ilmu yang mempelajari, meneliti, dan mengembangkan seni menyembunyikan sesuatu informasi. Steganografi dapat digolongkan sebagai salah satu bagian dari ilmu komunikasi. Kata steganografi berasal dari bahasa Yunani yang berarti "tulisan tersembunyi". Pada era informasi digital, steganografi merupakan teknik dan seni menyembunyikan informasi dan data digital dibalik informasi digital lain.

Secara teori, semua file umum yang ada di dalam komputer dapat digunakan sebagai media pembawa/wadah dari pesan rahasia. Semua dapat dijadikan tempat bersembunyi, asalkan file tersebut memiliki bit-bit data yang dapat dimodifikasi. Setelah dimodifikasi file media tersebut tidak boleh terganggu fungsinya dan kualitasnya tidak akan jauh berbeda dengan aslinya. Pada penelitian ini digunakan file audio berformat WAV sebagai media pembawa (carrier). Sedangkan data rahasia yang hendak disembunyikan pada media carrier tersebut adalah data audio yang bertipe WAV pula.

Data audio disembunyikan dalam media audio carrier agar dapat meningkatkan keamanan, terutama jika ditransmisikan. Banyak metode steganografi yang melekatkan sejumlah besar informasi rahasia di dalam least significant bit (LSB) media carrier. Karena perasaan manusia yang tidak sempurna dalam hal pendengaran keberadaan informasi rahasia yang ditempelkan tersebut akan sulit diketahui.

Hasil penelitian membuktikan bahwa file audio wav dapat dipakai sebagai sarana penyimpan data audio wav yang lain jika ukuran data audio yang hendak disembunyikan cukup jika menempati LSB dari data audio media carrier.

Kata kunci : *Steganografi, WAV, Decoder, Encoder*

Pendahuluan

Keamanan informasi dan data merupakan komponen penting dalam suatu sistem informasi dan sistem komunikasi. Penerapan sistem keamanan data dan informasi banyak dibutuhkan di berbagai instansi terutama perusahaan terhadap pesaing-pesaingnya, kantor-kantor pemerintahan yang bertanggung jawab terhadap rahasia negara, termasuk didalamnya instansi pertahanan keamanan. Umumnya data dan informasi rahasia akan diamankan dari publik.

Steganografi yang juga merupakan bagian dari kriptografi adalah suatu cara menyembunyikan informasi rahasia dalam suatu media pembawa (*carrier*) sehingga informasi tersebut tidak akan disadari keberadaannya oleh orang yang tidak berhak. Steganografi dapat digolongkan sebagai salah satu bagian dari teknik komunikasi. Kata steganografi berasal dari bahasa Yunani yang berarti "tulisan tersembunyi". Pada era komputer dan internet, steganografi merupakan teknik dan seni menyembunyikan data atau informasi rahasia digital dibalik media pembawa digital, sehingga

informasi digital yang sesungguhnya tidak kelihatan.

Secara teori, semua file gambar yang ada di dalam komputer dapat digunakan sebagai media, seperti file gambar berformat JPG, GIF, BMP, atau di dalam musik MP3, atau bahkan di dalam sebuah film dengan format WAV atau AVI. Semua dapat dijadikan tempat bersembunyi, asalkan file tersebut memiliki bit-bit data yang dapat dimodifikasi. Setelah dimodifikasi file media tersebut harus tidak terganggu fungsinya dan kualitasnya harus tidak jauh berbeda dengan aslinya. Media pembawa (*carrier*) jika berwujud sebuah gambar digital sering disebut sebagai *cover image*.

Teknik steganografi dimungkinkan akan memberikan keamanan informasi baik. Banyak metoda steganografi yang melekatkan sejumlah besar informasi rahasia di dalam data bit penyusun suatu file pada *media cover*. Karena kemampuan lihat manusia yang tidak sempurna dalam hal visualisasi, keberadaan informasi rahasia yang disisipkan tersebut akan tidak terlihat.

Tujuan dari penelitian ini adalah untuk menciptakan aplikasi yang mampu melakukan penyisipan (*encode*) dan membuka (*decode*) pesan rahasia berbentuk file audioWAV pada file audio WAV yang lain dengan teknik steganografi digital.

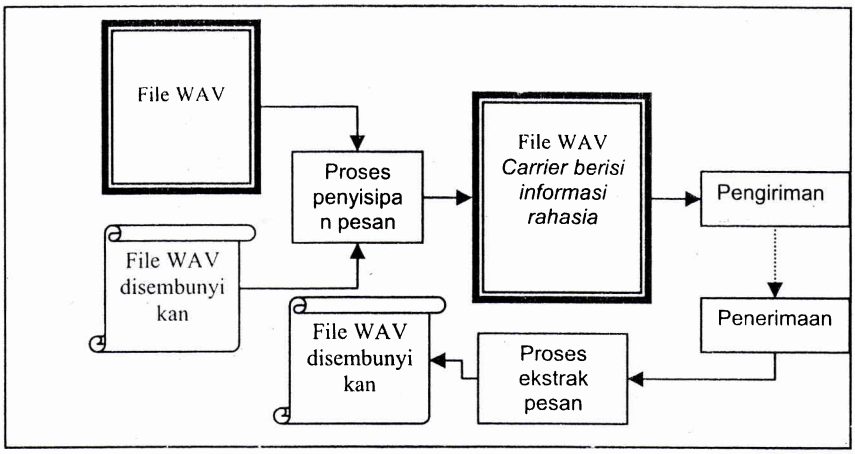
Penelitian dibatasi pada pembuatan aplikasi *decoder* dan *encoder* pesan rahasia dengan memanfaatkan file audio WAV sebagai pembawa pesan rahasia (*cover image*). Teknik steganografi yang dipakai adalah penyisipan bit-bit penyusun informasi dalam *Least Significant Bit* (LSB) penyusun data audio. File audio tersebut setelah disisipi data rahasia tersebut dapat dikirimkan melalui sarana umum. Hanya orang memahami adanya data tersembunyi dan memiliki aplikasi *encoder* saja yang dapat membuka data tersembunyi tersebut.

Teknik Umum Steganografi

Teknik steganografi ditunjukkan pada gambar 1. dimana sistem membutuhkan dua buah komponen yaitu :

1. Media pembawa (*carrier*)
2. Informasi rahasia yang disisipkan dalam *carrier*.

Pada steganografi *digital* file multimedia, seperti *image, audio* dan *video* adalah *carrier* yang sempurna. File tersebut seolah-olah menyediakan ruang untuk melekatkan data pesan tersembunyi dalam bit-bit penyusun data tersebut. Penelitian ini memanfaatkan file audio WAV untuk media *carrier*.



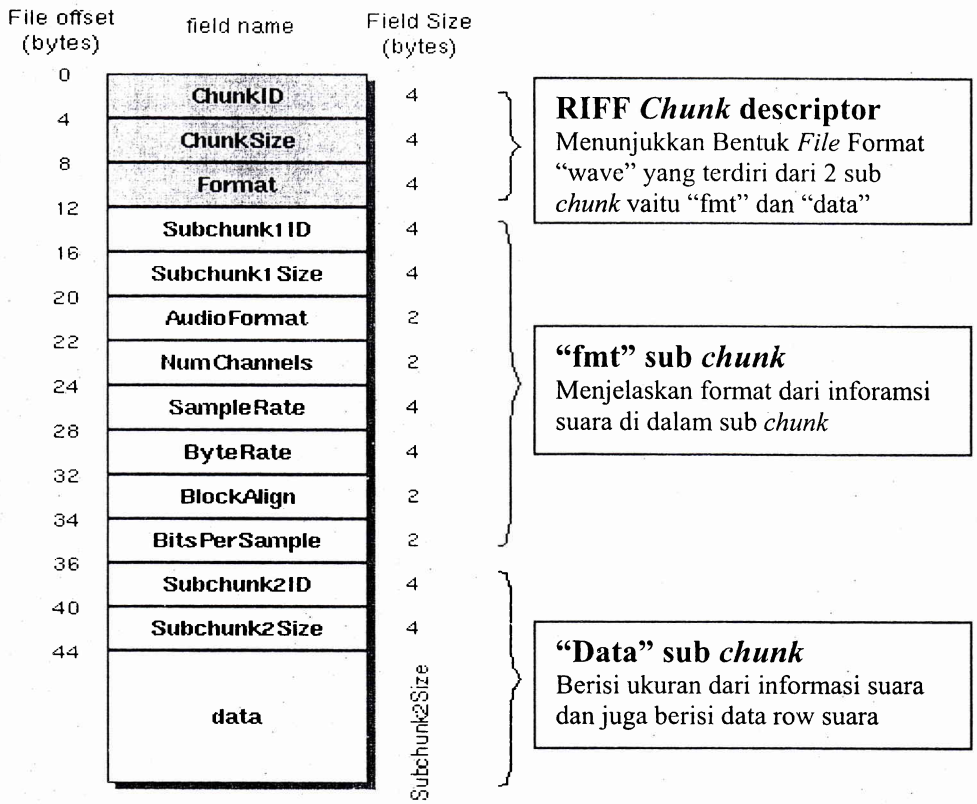
Gambar 1. Teknik Steganografi

File Audio WAV

File WAV telah menjadi salah satu *file* yang paling mendukung format *file audio* digital pada PC seiring dengan perkembangan dan ketenaran *Windows* dan juga pada sebagian besar program yang ditulis diatas*platformWindows*.

Dengan banyaknya aplikasi yang mendukung penggunaan *file* Wav, steganografi pada *file* Wav menjadi salah satu seni yang perlu dikembangkan untuk menyembunyikan informasi dan data. Steganografi pada *file* suara digital pada dasarnya adalah memanfaatkan keterbatasan kekuatan sistem pendengaran manusia. Karena perasaan manusia yang tidak sempurna dalam hal pendengaran, dimana manusia cendrung lebih memperhatikan arti suara dan melupakan kualitas suara, maka keberadaan informasi rahasia yang ditempelkan pada sebuah *file* suara digital dapat saja tidak dirasakan atau bahkan diabaikan, bagaimanapun informasi rahasia tersebut mungkin saja ditemukan jika belum ditempatkan secara baik. Steganografi pada media *file* suara akan menurunkan kualitas suara yang dihasilkan oleh *file* suara tersebut jika dibandingkan dengan *file* suara asli sebelum disisipkan pesan rahasia.

File format WAV ditujukan untuk kepentingan Windows dan pada umumnya bekerja dengan baik pada mesin Intel, semua nilai-nilai data disimpan didalam pemesanan Little-Endian (lebih didahulukan pada *least significant byte*). Format *file* WAVE adalah suatu subset dari spesifikasi RIFF Microsoft'S untuk penyimpanan *file* multimedia. Suatu *file* RIFF dimulai dengan suatu *file header* yang diikuti oleh suatu urutan *chunk* data. Suatu *file* WAVE sering hanya berupa *file* RIFF dengan "WAVE" *chunk* tunggal yang terdiri dari dua sub *chunks* yaitu " fmt" *chunk* yang menunjukkan/menetapkan format data dan "data" *chunk* yang berisi data sample yang aktual. Format Ini adalah " Format yang *Canonical*".



Gambar 2 Format File Wav

Format *canonical* WAVE dimulai dengan RIFF header:

Tabel 1 Format *canonical* WAVE dimulai dengan RIFF header

Offset	Ukuran	Nama	Keterangan
0	4	***Chunk*ID**	Berisi huruf/tulisan "RIFF" dalam bentuk ASCII (0x52494646 *big-endian form*).
4	4	***Chunk*Size**	36 + *SubChunk*2Size, atau lebih tepatnya: 4 + (8 + *SubChunk*1Size) + (8 + *SubChunk*2Size). Ini menjadi ukuran dari semua *chunk*, semua chunk mengikuti jumlah ini. Ini menjadi ukuran dari keseluruhan *file* dalam bytes dikurangi 8 *bytes* untuk dua *fields* yang tidak ikut dalam perhitungan ini: *Chunkid* dan *Chunksize*
8	4	**Format**	Berisi huruf/tulisan "WAVE" (0x57415645 *big-endian form*).

Format "WAVE" memiliki dua *subchunks*, yaitu : "fmt " dan "data":

1. "fmt " *subchunk* menunjukkan format data suara:

Tabel 2 "fmt " *subchunk* menunjukkan format data suara

Offset	Ukuran	Nama	Keterangan
12	4	***Subchunk*1ID**	Berisi huruf "fmt" (0x666d7420 *big-endian form).*
16	4	***Subchunk*1Size**	16 untuk PCM. Ini adalah ukuran dari semua *subchunk* yang mengikuti nomor ini
20	2	***Audio*Format**	PCM = 1 (contoh: Kuantisasi linier) Nilai selain dari 1 menandai adanya beberapa format terkompresi
22	2	**NumChannels**	Mono = 1, Stereo = 2
24	4	**SampleRate**	Jumlah sample per detik, jumlah ini tidak di pengaruhi oleh jumlah channel,contoh: 8000, 44100, dll
28	4	**Byte Rate**	SampleRate * NumChannels * BitsPerSample/8
32	2	**BlockAlign**	NumChannels * BitsPerSample/8 Banyaknya bytes untuk satu *sample* yang mencakup semua *channel*, yang berupa bilangan bulat
34	2	**Bit Per Sample**	8 bits = 8, 16 bits = 16

2. "data" *subchunk* berisi ukuran dari data dan data suara sebenarnya:

Tabel 3 "data" *subchunk* berisi ukuran dari data dan data suara sebenarnya

Offset	Ukuran	Nama	Keterangan
36	4	***Subchunk*2ID**	Berisi huruf/tulisan "data" dalam bentuk ASCII (0x64617461 big-endian form).
40	4	***Subchunk*2Size**	NumSamples * NumChannels * BitsPerSample/8 Menunjukkan banyaknya byte dalam data
44	*	**Data**	Data suara yang sesungguhnya (*The actual sound data*)

Penyembunyian Data pada *File* WAV dengan Metode Penyisipan LSB

Setiap *file* WAV dengan format data 8 bit akan memiliki susunan data sepanjang 8 bit dalam bentuk biner pada setiap *sample* data dari

file tersebut pada kedua *channel* datanya(*channel* kiri dan kanan), jika *file* Wav ini hanya memakai *channel* data pada sebelah kiri saja (*channel* memakai speaker sebelah kiri) maka *file* Wav tersebut bertipe *mono* dimana *file* Wav tersebut hanya disusun oleh data biner pada *channel* sebelah kiri saja, sedangkan jika *file* Wav tersebut memakai dua *channel* (kanan dan kiri) maka *file* Wav tersebut disebut bertipe *stereo* dimana *file* Wav tersebut disusun oleh data biner pada *channel* sebelah kiri dan kanan.

Untuk melakukan proses steganografi pada *file* Wav dengan metode LSB, dapat dilakukan dengan berbagai cara, tergantung jenis *file*nya. Jika *file* Wav adalah Wav bertipe *mono* maka setiap satu karakter pesan akan disimpan dalam setiap beberapa *sample* data Wav. Misalkan ada data "A" dengan bit biner "01000001", sedangkan *file* wav memiliki data-data yang berurutan dari *sample* data yang ke-1 sampai *sample* data yang ke-8 seperti berikut:

$(00100110)_1(11101001)_2(11001000)$
$_3(00100110)_4(11001000)_5(11101000)_6(11001$
$000)_7(11101000)_8$

Untuk melakukan proses steganografi LSB data "A" pada potongan *file* wav tersebut dilakukan dengan menempelkan nilai bit biner "A" pada biner nomor tertentu dari potongan *file* Wav tersebut. Nomor bit biner yang akan ditempelkan dipilih berdasarkan kriteria tertentu yang diinginkan, misalnya akan ditempelkan pada nilai biner *sample* data no-1 samapi no-4, maka nilai biner "A " dibagi menjadi 4(empat) bagian yang sama, misalnya nilai biner "A" adalah "01000001" akan menghasilkan bit-bit sebagai berikut: "0","1","0","0","0","0","0" dan "1", kemudian bit-bit tersebut ditempelkan pada nilai biner dari *sample* data *file* Wav.

Proses penempelan bit binger dari huruf "A" dilakukan pada bit-bit rendah (*least significant bit*) dari penyusun biner potongan *file* wav yang diinginkan, jadi bit biner "A" yang telah dipecah ditempelkan dengan cara mengganti 1(satu) bit biner paling belakang dari nilai biner potongan *file* Wav yaitu :

$(00100110)_1(11101001)_2(11001000)$
$_3(00100110)_4(11001000)_5(11101000)_6(11001$
$000)_7(11101000)_8$

Nilai biner "A" yang telah dipecah menjadi delapan bagian akan ditempelkan pada posisi bit-bit/angka yang dicetak miring sehingga data Wav tersebut setelah ditempelkan akan menjadi :

$(00100110)_1(11101001)_2(11001000)$
$_3(00100110)_4(11001000)_5(11101000)_6(11001$
$000)_7(11101001)_8$

Perhatikan bit-bit yang dicetak tebal, perubahan hanya akan terjadi pada beberapa bit rendah saja, sehingga *file* Wav akan menghasilkan suara yang tidak begitu jauh berbeda dengan *file* aslinya.

Untuk *file* wav 8 bit yang bertipe *stereo*, kedua *channel* data dapat digunakan untuk menyembunyikan data tergantung dari metode atau pada bagian data Wav mana sebuah data akan ditempelkan, misalnya pada sebuah potongan *file* Wav 8 bit bertipe *stereo* akan ditempelkan karakter "A" dimana masing-masing nilai data biner pada 2 *sample* data pertama *file* Wav tersebut adalah:

Sample data Ke-: 1 2
LeftChannel: $(00100110)_1$ $(11101001)_2$
RightChannel: $(11001000)_3$ $(00100110)_4$

Dari data yang didapat, kemudian ditandai dengan memberi nomor dari kiri ke kanan dan dari atas kebawah sehingga data biner tersebut seolah-olah berurutan. Sama halnya dengan penempelan pada *file* Wav *mono*, kemudian nilai bit biner dari huruf "A" yang telah dipecah menjadi empat bagian ditempelkan kepada potongan *file* Wav tersebut sesuai dengan nomor urut yang telah dibuatkan. Hasil dari penempelan huruf "A" pada potongan *file* Wav tersebut adalah:

Sample data Ke-: 1 2

LeftChannel :$(00100101)_1$ $(11101000)_2$

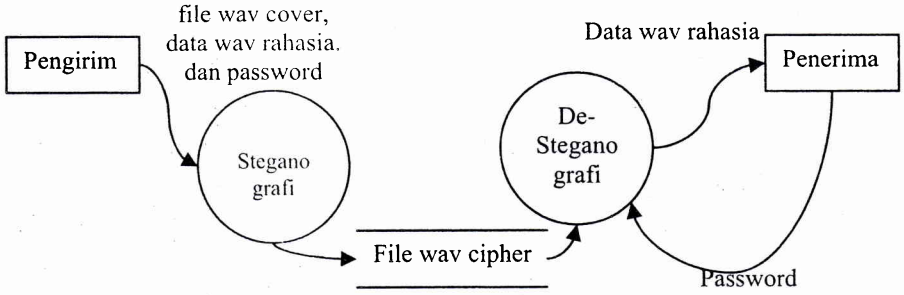
RightChannel :$(11001000)_3$ $(00100101)_4$

Data akan mengalami perubahan sangat sedikit sehingga suara yang dihasilkan oleh *file* Wav yang telah disteganografikan tidak akan berbeda jauh dengan *file* Wav aslinya.

Perancangan Aplikasi Decoder dan Encoder

1. Gambaran umum Proses *Decoder* dan *Encoder*

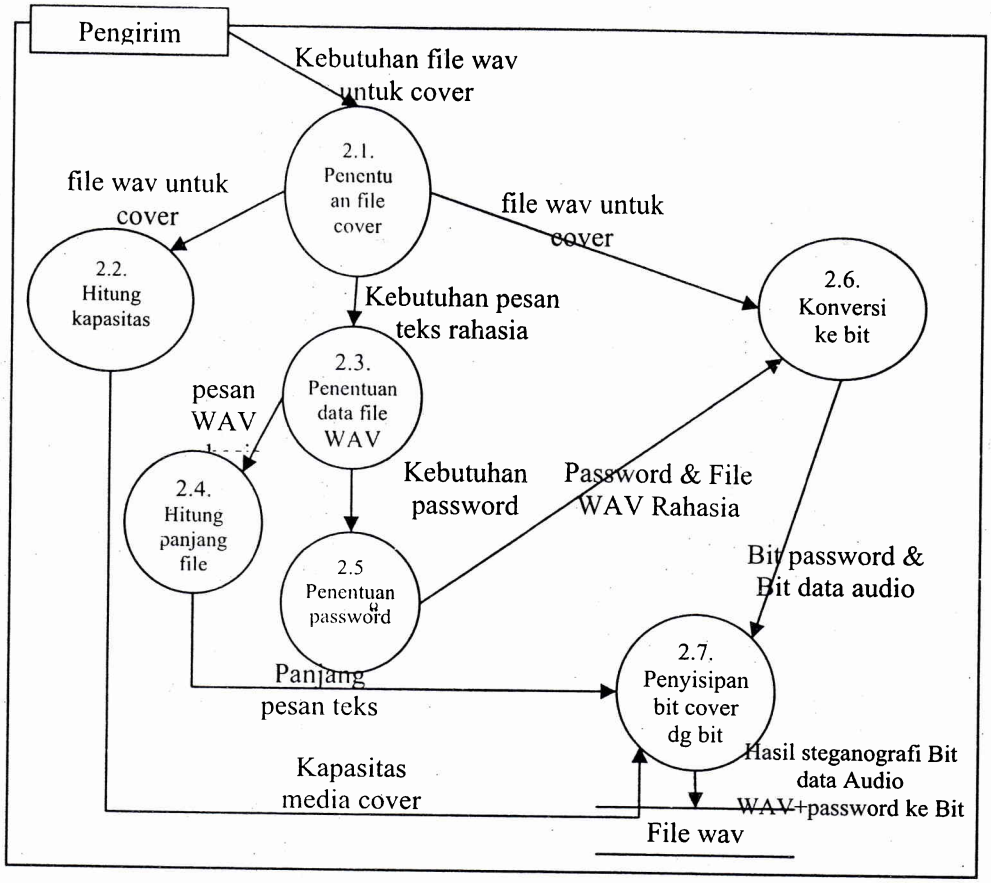
Berikut ini adalah gambaran umum proses steganografi dan proses de- steganografi pada file *audio* WAV dengan data rahasia juga berupa file *audio* WAV(gambar 3):



Gambar 3. Diagram Seluruh Sistem Steganografi *Audio* WAV

2. Aplikasi Steganografi (*Encoder*)

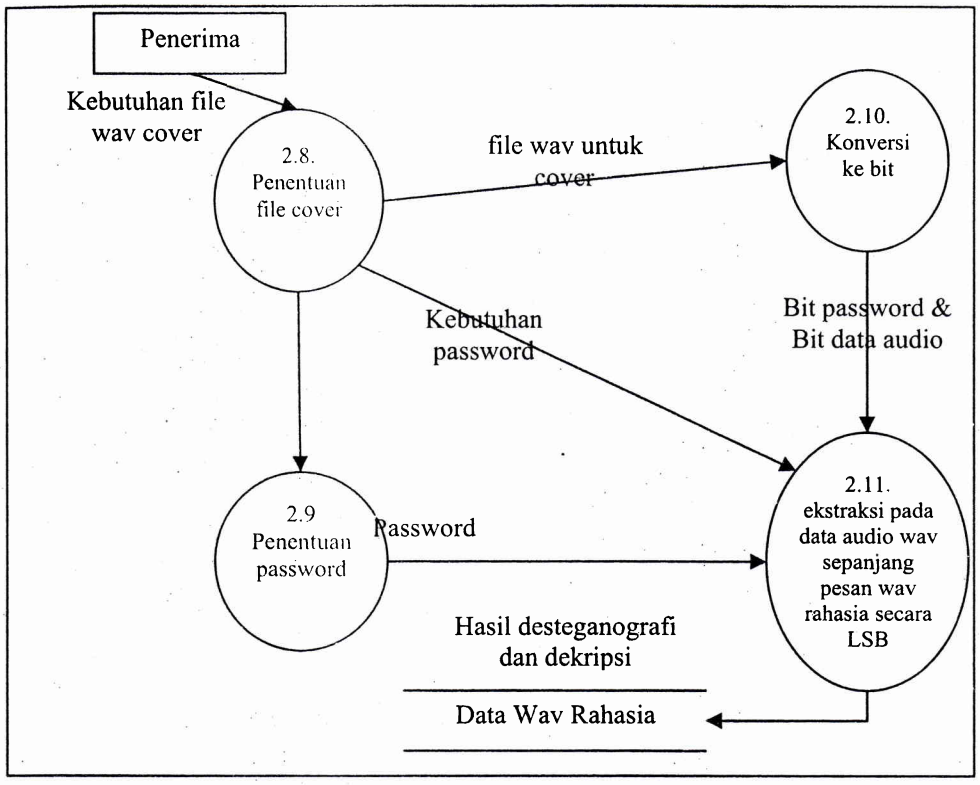
Aplikasi encoder merupakan aplikasi untuk menyisipkan dalam media *carrier*. Diagram proses kerja aplikasi tersebut adalah (lihat gambar 4):



Gambar 4. Diagram Proses *Encode*

3. Aplikasi Decoder

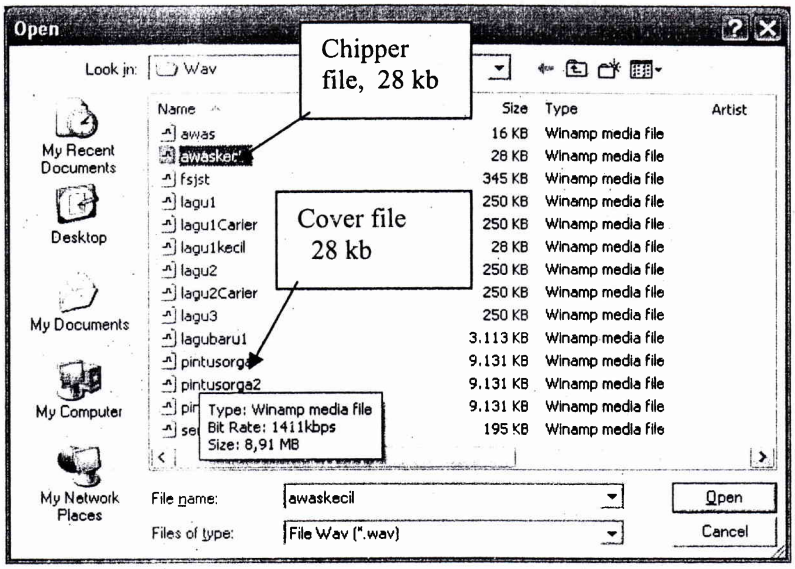
Kebalikan dari proses encode adalah proses decode dimana pesan tersembunyi diekstrak dan di dekripsi dari cover image. Proses ini digambarkan dalam diagram sebagai berikut:



Gambar 4. Diagram Proses *Decode*

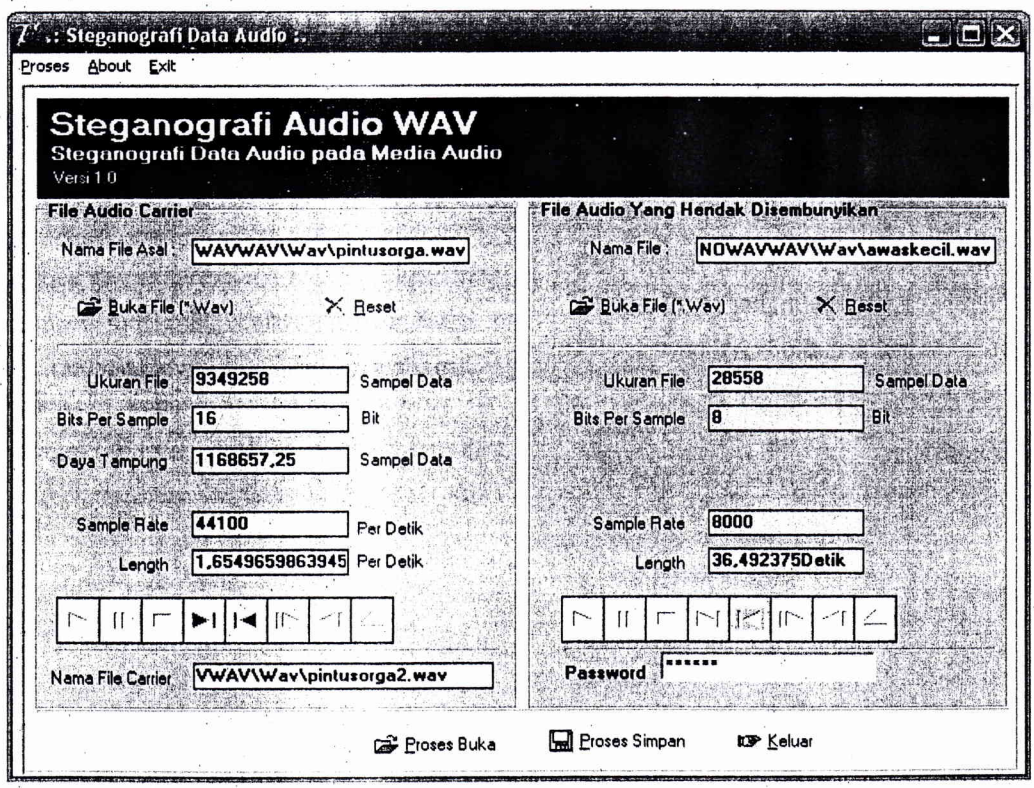
4. Pemrograman

Pengolahan bit stream penyusun suatu file gambar bisa dibantu dengan beberapa development tool Delphi. Dengan aplikasi Delphi pengaksesan bit stream dan penanganan file *audio* telah tersedia fungsinya. Delphi memiliki *object stream* yang mampu menangani manipulasi bit-bit penyusun file, dan fungsi-fungsi komputasi bit baik encode bit ke berbagai bentuk bilangan seperti heksadesimal, desimal atau biner. Hasil pemrograman adalah sebagai berikut:



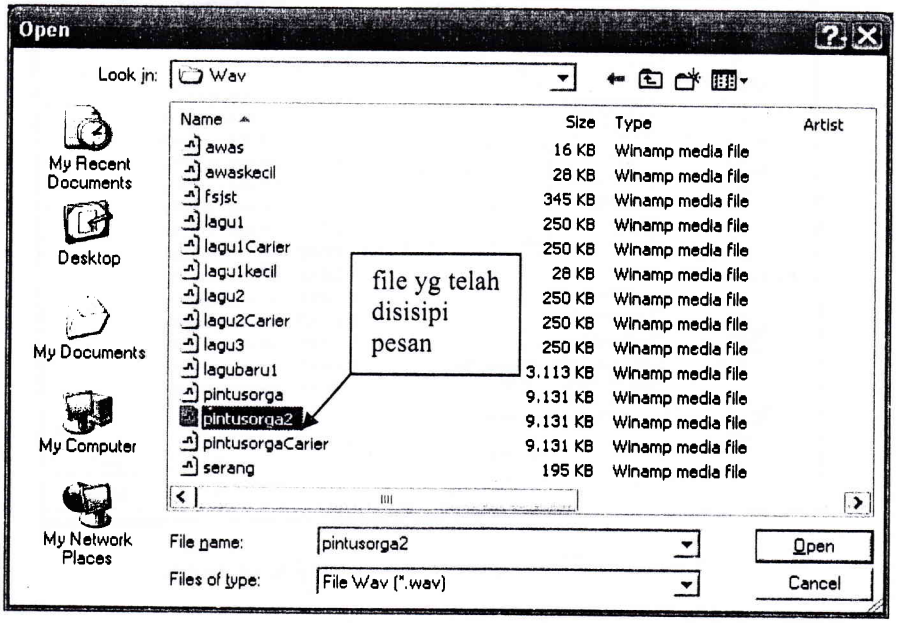
Gambar 5. File chiper dan cover yang akan diolah

Gambar diatas menunjukkan bagaimana memilih file cover dan file *cipher*.



Gambar 6. Aplikasi Steganografi *Audio* WAV

akan disimpan menjadi file baru (disebut file *carrier*) yaitu pintusorga2.wav (lihat gambar 7).



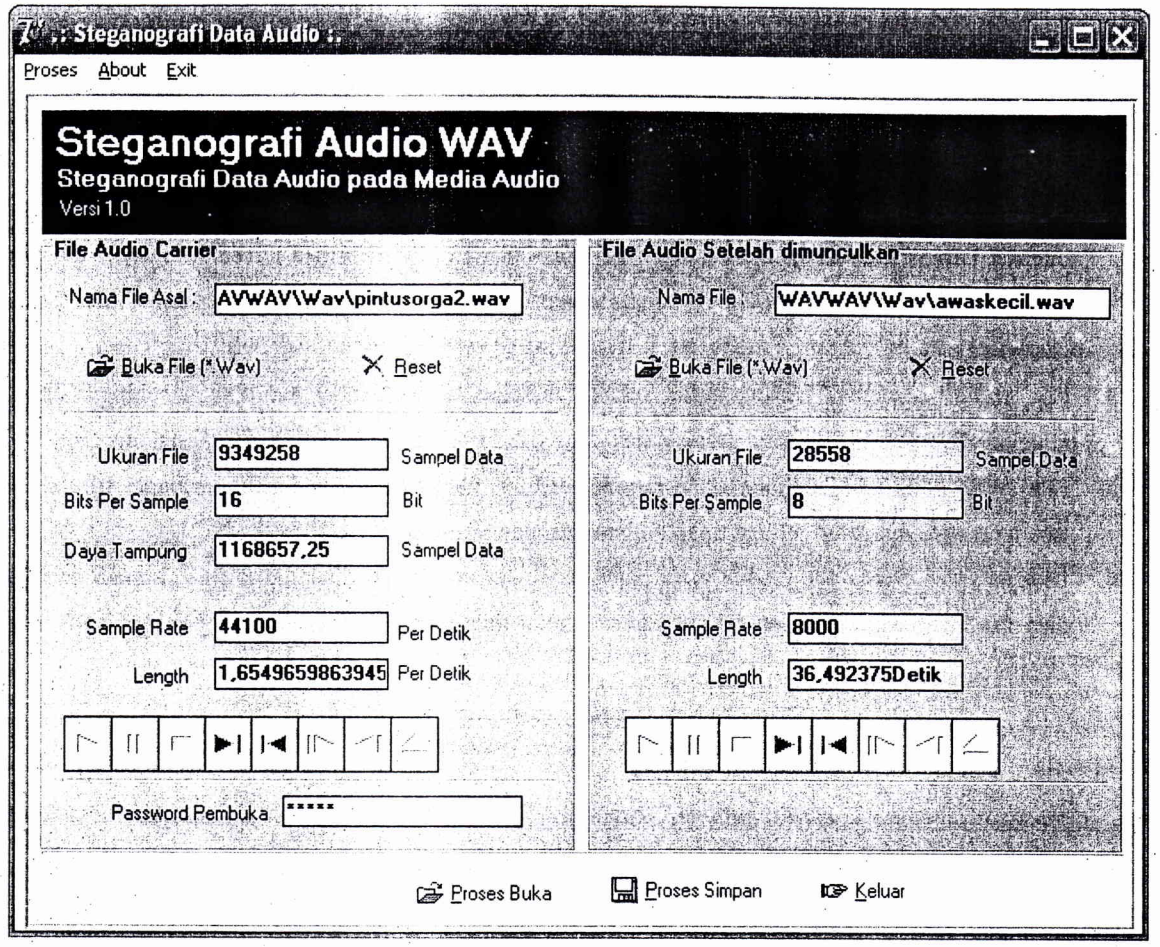
Gambar 7. File Hasil Proses Penyisipan

File WAV cover harus berukuran 8 kali lebih besar daripada file WAV*cipher* karena untuk penyimpanan satu buah sampel data pada file WAV *cipher*, dimana satu buah sampel data tersebut terdapat 8 bit data digital membutuhkan delapan buah sampel data file WAV cover.

Jumlah sampel data file cover = 9349258 sampel data

Jumlah sampel data file *cipher* = 28558 sampel data

Maka file *cipher* tersebut bisa disisipkan dalam file cover karena kapasitas file cover adalah 1168657,25. (lebih kecil daripada kapasitas maksimal file cover tersebut).



Gambar 8. Aplikasi De Steganografi *Audio* WAV

Gambar 7 diatas merupakan proses de-steganografi dimana file *cipher* dimunculkan dari file cover membentuk file WAV yang sama fungsidanukurannya seperti file *cipher* sebelum proses steganografi. File *cipher* hanya dapat dibuka jika password diisikan dengan benar.

Kesimpulan

Strategi steganografi seperti yang telah diuraikan di atas dapat diambil kesimpulan sebagai berikut.

1. Keamanan berlipat ganda terutama jika pesan hendak dikirimkan melalui sarana publik seperti internet karena orang lain tidak akan menyadari adanya pesan rahasia tersebut. Pesan semakin aman karena untuk membukanya membutuhkan aplikasi ekstrak/decode yang tepat, dan kunci password.
2. File *audio* yang mengandung pesan rahasia tersebut tidak berubah ukurannya sehingga tidak menimbulkan kecurigaan bagi pihak-pihak yang tidak berkepentingan terhadap pesan tersebut.
3. File *audio* yang mengandung pesan rahasia tersebut tidak berubah fungsinya sebagai file *audio* dan secara pendengaran telinga tidak berbeda dengan suara aslinya
4. Ukuran sampel data file *cipher* yang akan disisipkan maksimal adalah 1/8 dari jumlah sampel data file cover karena setiap bit penyusun sampel data file *cipher* akan menggantikan LSB satu sampel data file cover.

Reference

http://ccrma.stanford.edu/courses/422/projects/WaveFormat/, *WAVE PCM soundfile format*, craig@ccrma.stanford.edu, 2003

Masaleno, Andino,2003, *Pengantar Steganografi*, Kuliah Umum IlmuKomputer.Com

Nicholas J. Hopper, Toward a theory of Steganography, School of Computer Science Carnegie Mellon University Pittsburgh, July 2004

Raharjo, Budi,1998-2005, *Keamanan Sistem Informasi Berbasis Internet*, PT Insan Infonesia Bandung & PT INDOCISC – Jakarta

Sellars, Duncan, *An Introduction to Steganography*, http://www.cs.uct.ac.za/courses/CS400W/NIS/papers99/dsellars/stego.html

Sukmawan, Budi, *Steganografi*, http://students.ukdw.ac.id/~22033120/ steganografi.html

Biodata penulis

Siti Khomsah, Skom. Mengajar di jurusan Manajemen Informatika STTI Respati Yogyakarta.

Minat Penelitian pada Sistem Informasi Geografis, DSS, Keamanan Sistem.

Email: camil_107@yahoomail.com

VOLUME: 1, Nomor 4, Maret 2007 ISSN : 1907-2430

Jurnal Teknologi Informasi RESPATI

Sekolah Tinggi Teknologi Informasi RESPATI
Yogyakarta