Implementasi Audio Steganografi dalam Penyembunyian Pesan Rahasia

Gregorius Moses Marevson - 13520052
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail):

***Abstract*—Steganografi sebagai seni penyembunyian pesan rahasia masih sering digunakan. Salah satu media yang digunakan dalam digital steganografi adalah melalui file audio. Beberapa metode yang dapat digunakan dalam audio steganografi mencakup LSB Algorithm, Phase Coding, Echo Hiding, dan Spread Spectrum. Audio Steganografi menggunakan LSB Algorithm memiliki keuntungan berupa kemudahan implementasi dan kecepatan algoritma. Kelemahan dari LSB Algorithm adalah adanya noise yang dapat didengar telinga manusia. Beberapa solusi yang dapat digunakan untuk mengurangi noise tersebut adalah dengan membuat step (jarak) di tiap byte yang diganti atau dengan memilih lagu sedemikian rupa sehingga noise berada di bagian audio yang diam.**

Kata kunci—Steganografi, LSB Algorithm, Noise, File Audio

I. PENDAHULUAN

Steganografi adalah ilmu atau seni untuk menyembunyikan pesan rahasia dengan suatu cara sedemikian sehingga keberadaan pesan tersebut tidak menimbulkan kecurigaan. Steganografi banyak digunakan hingga kini dalam bentuk steganografi digital menggunakan media berupa teks, audio, gambar, ataupun video. Dalam steganografi, kriteria yang harus terpenuhi adalah imperceptible, fidelity, recovery, dan capacity tanpa terlalu memedulikan robustness. Berdasarkan ranah operasinya, steganografi terbagi menjadi dua yaitu spatial domain methods dan transform domain methods.

Salah satu media steganografi digital yang sering digunakan adalah audio. Audio steganografi adalah seni menyembunyikan pesan tersembunyi di dalam file audio dengan cara mengeksploitasi kelemahan indra pendengaran manusia. Secara anatomi, telinga manusia dapat menangkap getaran dengan rentang frekuensi 20Hz hingga 20000Hz. Rentang frekunsi telinga tiap orang berbeda - beda dan dipengaruhi oleh berbagai faktor seperti usia, jenis kelamin, dan kesehatan. Salah satu cara untuk melakukan audio steganography adalah dengan menggunakan suara infrasonik dan ultrasonic untuk mengirimkan pesan tersembunyi diiringi dengan suara audio di frekuensi yang dapat didengar untuk menipu penerima yang tidak dituju.

Adapun beberapa kasus audio steganography yang memancing perhatian adalah sebagai berikut:

- Suara tersembunyi dapat mengaktifkan Smart Home Devices

  Pada tahun 2018, Amazon mengambil langkah untuk mencegah aktivasi tak disengaja perangkat Amazon Echo ketika iklan Alexa Super Bowl ditayangkan. Beberapa berspekulasi bahwa Amazon mengubah rentang frekuensi akustik menjadi lebih baik. Kasus ini dapat terjadi di berbagai perangkat lain dari sumber suara yang tidak terduga seperti televisi.

- Suara tersembunyi dalam film yang dapat mempengaruhi emosi pendengarnya

  Audio steganography banyak dimanfaatkan oleh pembuat film untuk mempengaruhi perasaan penonton lebih lanjut. Beberapa studi mengatakan bahwa suara infrasonik dapat menimbulkan perasaan takut, panik, juga depresi. Hal itu banyak dimanfaatkan untuk membuat perasaan penonton semakin suram di film horror. Masih banyak teknik pemengaruh emosi menggunakan audio yang digunakan dalam dunia perfilman.

- Silent Subliminal Technology

  Beberapa orang percaya penggunaan audio subliminal dapat mempengaruhi alam bawah sadar manusia. Ide dasarnya adalah suara subliminal tidak dapat didengar oleh telinga manusia, tetapi masih dapat ditangkap oleh alam bawah sadar sehingga dapat dimanfaatkan mempengaruhi alam bawah sadar menjadi lebih baik. Biarpun tidak ada riset yang mendukung kebenaran audio subliminal, teknik yang digunakan untuk membuatnya adalah audio steganography.

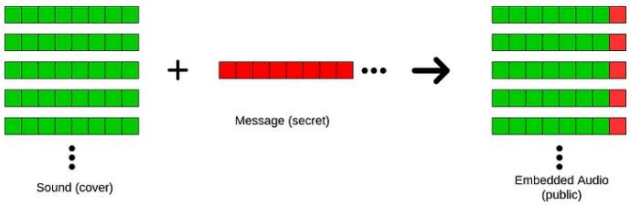
Metode audio steganography sangat banyak seperti LSB (Least Significant Bits) Algorithm, Phase Coding, Echo Hiding, dan Spread Spectrum. Metode yang akan digunakan dalam makalah ini adalah algoritma LSB.

II. DASAR TEORI

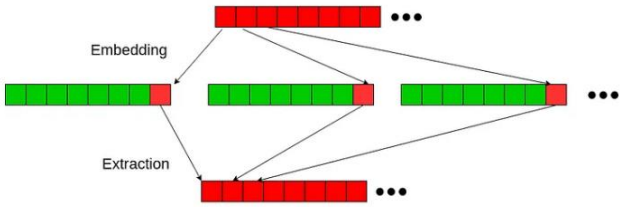
A. Algorithm Least Significant Bits

Algoritma Least Significant Bits (LSB) adalah metode steganografi paling umum yang bisa digunakan untuk berbagai

macam data. Proses encoding dilakan dengan menukar bit terakhir dari tiap byte dalam cover dengan bit-bit pesan yang ingin disembunyikan. Dalam hal ini, pesan tersembunyi harus berukuran 8 kali lebih kecil dari cover agar dapat disembunyikan seluruhnya. Untuk mendapatkan pesan tersembunyi kembali, penerima melakukan ekstraksi pada bit terakhir di tiap byte file tersebut. Hasil ekstraksi kemudian disatukan dan dikonversi ke bentuk pesan semula.



Gambar 1 Encoding dalam Audio Steganography



Gambar 2 Decoding dalam Audio Steganography

Pengubahan bit terakhir dari suatu file biasanya tidak menimbulkan noise yang signifikan, terutama pada file gambar. Hanya saja, telinga manusia lebih sensitif dalam mendeteksi perubahan di suara, sehingga noise yang muncul dapat dideteksi dan menimbulkan kecurigaan. Salah satu cara untuk mengurangi noise tersebut adalah dengan membuat step (jarak) setiap penggantian bit LSB. Step yang lebih besar dapat memperkecil peluang noise terdengar. Selain itu, bisa dilakukan penggantian bit LSB pada saat audio sedang diam. Hal ini dilakukan dengan memilih audio mask yang diam, biasanya, pada awal atau akhir file.

Metode LSB memiliki keuntungan berupa kecepatan melakukan *embedding* bit-bit pesan, dan rendahnya kompleksitas komputasional dibanding dengan algoritma lain. Selain itu, metode LSB adalah metode yang paling mudah dipahami dan diimiplementasi. Sebaliknya, metode ini tidak aman dan mudah di-Steganalisis.

B. Waveform Audio File Format

Waveform Audio File Format (WAVE atau WAV) adalah format file audio untuk menyimpan bitsream audio di komputer pribadi. Format ini dikembangkan pada tahun 1991 oleh IBM dan Microsoft, dan digunakan sebagai format utama di sistem Microsoft Windows untuk audio yang tidak dikompres (uncompressed audio). Bitstream encoding yang biasa digunakan berformat linear pulse-code modulation (LPCM). WAV adalah aplikasi dari metode Resource Interchange File Format (RIFF) untuk menyimpan data dalam chunks (bagian-bagian kecil). Dalam makalah ini, file audio yang digunakan sebagai cover dalam audio steganography berformat *.wav* karena merupakan format yang mendukung lossless compression. Hal ini dapat memudahakan pengetesan noise yang muncul setelah melakukan steganography.

III. IMPLEMENTASI

Program yang dibuat adalah program audio steganography sederhana yang dapat menyembunyikan sebuah teks rahasia. Program dibuat dalam bahasa python menggunakan library scipy.io.wavfile untuk membantu pembacaan file audio. Program terbagai menjadi dua bagian yaitu file encrypt.py dan decrypt.py.

A. Proses Encoding

Pada encrypt.py, dilakukan proses encoding pesan rahasia ke dalam cover file audio. Secara umum, program encrypt.py melakukan proses encoding sebagai berikut:

1. read audio_data, secret_message
2. append secret_message with end_symbol
3. convert secret_message into list of bits
4. encode each bit into LSB of audio_data
5. write output

Hal itu dilakukan dalam kode berikut

```
from scipy.io.wavfile import read, write
import numpy as np

filename = "song/" + input("Enter file name (.wav): ")
rate, data = read(filename)

secret = input("Enter secret message: ") + "###"
bits = list(map(int, ''.join([bin(ord(c)).lstrip('0b').rjust(8,'0') for c in secret])))

shape = data.shape
data = np.reshape(data, data.size)
for i, bit in enumerate(bits):
    data[i] = (data[i] & 254) | bit

data = np.reshape(data, shape)

output = input("Enter output name (.wav): ")
write(f"output/{output}", rate, data)
```

Program menerima nama file audio sebagai cover dengan format *.wav* pada folder song/. Program menerima secret message berupa teks dan menggabungkannya dengan end symbol "###". End symbol akan digunakan untuk mempermudah proses decoding. Setelah itu dilakukan penggantian LSB. Terakhir, program meminta nama file output dan menyimpannya di folder output.

B. Proses Decoding

Pada decrypt.py, dilakukan proses decoding pesan rahasia dalam cover file audio. Secara umum, program decrypt.py melakukan proses decoding sebagai berikut:

1. read encoded_audio
2. extract each LSB of the encoded_audio
3. combine the result into a single string
4. split the string using end symbol as separator
5. write the clean message

Hal itu dilakukan dalam kode berikut

```
from scipy.io.wavfile import read, write
import numpy as np

filename = "output/" + input("Enter file name (.wav): ")
rate, data = read(filename)

shape = data.shape
data = np.reshape(data, data.size)

byte_extract = [bytes & 1 for bytes in data]
string_extract = "".join(chr(int("".join(map(str,byte_extract[i:i+8])),2)) for i in range(0,len(byte_extract),8))
secret = string_extract.split("###")[0]

print("Secret message: " + secret)
```

Program menerima file yang akan di-decode. Dilakukan proses ekstraksi tiap LSB dalam file cover. Hasil ekstraksi digabungkan menjadi satu teks. Teks tersebut kemudian di-split menggunakan end symbol "###" untuk mendapatkan hasil bersihnya. Akhirnya program menampilkan pesan yang disembunyikan.

IV. Pengujian dan Analisis

Pengujian yang dilakukan akan mengukur hal sebagai berikut:

1. Apakah pesan rahasia berhasil di-encode dan di-decode?
2. Seberapa besar noise yang dapat didengar pada audio yang sudah di-encode?

A. Uji Fungsionalitas Encode dan Decode

Pengujian dilakukan dengan melakukan encoding pada encrypt.py, menyimpan hasilnya, lalu melakukan decoding pada decrypt.py. File yang menjadi cover disimpan terlebih dahulu dalam folder song/. Terdapat dua file audio yang akan digunakan dalam pengujian yaitu bocchi.wav dan rain.wav.

```
(env) PS C:\Users\sinmo\Documents\GitHub\steganography> python encrypt.py
Enter file name (.wav): bocchi.wav
C:\Users\sinmo\Documents\GitHub\steganography\encrypt.py:5: WavFileWarning: Reache
d EOF prematurely; finished at 11460687 bytes, expected 4294967303 bytes from head
er.
  rate, data = read(filename)
Enter secret message: Test secret message!!!
Enter output name (.wav): bocchi_secret.wav
(env) PS C:\Users\sinmo\Documents\GitHub\steganography>
```

Pada file bocchi.wav, secret dimasukkan sebagai "Test secret message!!!", lalu hasilnya disimpan dalam file bocchi_secret.wav.

```
(env) PS C:\Users\sinmo\Documents\GitHub\steganography> python encrypt.py
Enter file name (.wav): rain.wav
C:\Users\sinmo\Documents\GitHub\steganography\encrypt.py:5: WavFileWarning: Reache
d EOF prematurely; finished at 26550351 bytes, expected 4294967303 bytes from head
er.
  rate, data = read(filename)
Enter secret message: will it rain tonight?
Enter output name (.wav): rain_secret.wav
(env) PS C:\Users\sinmo\Documents\GitHub\steganography>
```

Pada file rain.wav, secret dimasukkan sebagai "will it rain tonight?", lalu hasilnya disimpan dalam file rain_secret.wav. Setelah itu, dilakukan decoding pada kedua file untuk mendapatkan pesan rahasia.

```
(env) PS C:\Users\sinmo\Documents\GitHub\steganography> python decrypt.py
Enter file name (.wav): bocchi_secret.wav
Secret message: Test secret message!!!
(env) PS C:\Users\sinmo\Documents\GitHub\steganography> python decrypt.py
Enter file name (.wav): rain_secret.wav
Secret message: will it rain tonight?
(env) PS C:\Users\sinmo\Documents\GitHub\steganography>
```

Pada gambar di atas, terlihat bahwa hasil teks rahasia berhasil di-decode menjadi pesan semula.

B. Uji noise

Pengujian noise dilakukan secara subjektif dengan cara mendengar file audio hasil encoding. Untuk file bocchi_secret.wav dan rain_secret.wav, noise tidak terdengar dengan jelas. Hal ini bisa disebabkan oleh audio diam di awal file bocchi.wav atau pendeknya pesan teks sehingga noise hilang begitu cepat.

V. Kesimpulan dan Saran

Audio steganografi dengan metode LSB berhasil diimplementasi untuk menyembunyikan pesan berupa teks

dalam cover berupa file audio .wav. Program berhasil melakukan encoding dan decoding pesan. Program memiliki noise yang masih dapat didengar karena LSB yang diimplementasi tidak memiliki step (jarak).

LINK KODE PROGRAM

Berikut adalah link github kode program: https://github.com/Moses-ui/steganography

UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena atas karunia-Nya lah, penulis dapat menyelesaikan makalah ini. Penulis mengucapkan terima kasih kepada Bapak Rinaldi Munir selaku dosen Mata Kuliah IF4020 Kriptografi yang telah mengajari penulis dengan antusias selama satu semester ini, kiranya Pak dosen bisa terus mengajar dengan semangat menggebu-gebu. Penulis juga mengucapkan terima kasih atas dukungan dan doa dari keluarga tercinta dalam setiap proses perkuliahan.

REFERENSI

[1] Munir, Rinaldi. (2024). "Steganografi - Bagian 1" Program Studi Informatika ITB. https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2023-2024/07-Steganografi-Bagian1-2024.pdf

[2] Munir, Rinaldi. (2024). "Steganografi - Bagian 2" Program Studi Informatika ITB. https://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2023-2024/08-Steganografi-Bagian2-2024.pdf

[3] https://sumit-arora.medium.com/audio-steganography-the-art-of-hiding-secrets-within-earshot-part-2-of-2-c76b1be719b3

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 25 Juni 2024

Gregorius Moses Marevson - 13520052