Real Time Audio Steganographic Countermeasure

T.M. Srinivas and P.P. Amritha

Abstract Steganographic techniques are used to embed data into a cover file using different algorithms. In this paper audio steganography countermeasure is discussed which uses a technique called double stegging or steganographic jamming where variations of LSB embedding algorithms are used for audio steganography prevention which can be used in real time under acceptable information loss. We then proceed to show that this method renders hidden embedded data unrecoverable. The resulting audio quality after Steganographic Jamming is evaluated. Mean Opinion Score and Signal to Noise Ratio are used to calculate the quality of output audio file which shows the effectiveness of the technique described.

Keywords Audio steganographic countermeasure · Double stegging · Steganographic jamming · Least significant bit

1 Introduction

The science of hiding information within media is called Steganography. While cryptography protects by randomizing the content of secret data, Steganography keeps hidden the existence of secret data itself [5]. Steganography uses digital media like images, audio and video as cover. In this experiment the focus is only on audio steganography where audio files are used as cover media. Application of Steganography in audio is challenging as Human Auditory System (HAS) is more sensitive to small changes in audio data than Human Visual System (HVS) [6, 14]. Motivation for this experiment is to prevent use of this technology in organized crime [7] and also more recently increasing insider threats at organizations. Further

T.M. Srinivas (✉)
TIFAC CORE in Cyber Security, Amrita School of Engineering, Coimbatore, India
e-mail: srinivas31.meharwade@gmail.com

P.P. Amritha
Amrita Vishwa Vidyapeetham, Amrita University, Coimbatore, India
e-mail: pp_amritha@cb.amrita.edu

S.C. Satapathy et al. (eds.), *Data Engineering and Intelligent Computing*,
Advances in Intelligent Systems and Computing 542,
DOI 10.1007/978-981-10-3223-3_27

new ideas and approaches emanate for robust design of steganographic algorithms by looking at steganographic prevention methods [1].

When steganography is applied to audio data there are three important requirements to be taken care of Robustness, Inaudibility or Undetectability and Capacity [8]. Hence audio quality is not to be degraded while defending against steganography to make the defense technique undetectable. This work separates itself work from steganalysis which focuses on detecting and finding hidden data present in cover medium whereas our goal is to interfere with the steganographic receiver whether or not there exists a steganographic communication. Hence this method can be applied to all audio material as this method does not introduce any noticeable disturbances to the audio signal.

In this paper the use of Least Significant Bit (LSB) embedding and it's variant for steganographic jamming are discussed to make the data, hidden using Steganographic techniques irrespective of underlying algorithm, irrecoverable without substantial reduction in audio quality of the steganographic jammed audio media. Audio quality measurement techniques are then used to grade the quality of the output audio data in terms of Mean Opinion Score (MOS), Waveform Amplitude Distribution Analysis-Signal to Noise Ratio (WADA-SNR) [2], National Institute of Standards and Technology-Signal to Noise Ratio (NIST-STNR) and Bit Error Rate (BER).

2 Related Work

Steganographic Jamming by way of "Double-Stegging" as an idea is presented by [9]. Here they have discussed about the same application doing the second embedding process which embedded the secret data the first time. Also they are discussing about image files whereas the focus here is on audio as cover file and also no evaluation of output audio quality or efficiency of the proposed algorithm is discussed.

In [1] although their aim is to prevent audio steganography, it is completely different from what is experimented here as their work is about combining basic signal processing techniques in a novel way to prevent audio steganography by interfering with the steganographic receiver.

In [3] they discuss about audio steganographic prevention in cloud storage systems where they propose two algorithms, the enhanced-RS algorithm and the SADI algorithm. Here they first detect whether there is steganography present in the stored stationary files using the enhanced-RS algorithm and then try to destroy the hidden data using the SADI algorithm which works by interchanging the bits based on the minimum Manhattan distance. As the author initially find the media with stego content and then try to destroy the hidden data, it cannot be used in real time whereas the system suggested here can work in real time.

In [11] examination on steganographic tools for hiding information is presented which also discusses the different approaches these tools used to hide data and the

supported types of cover media is presented of which our interest is only on tools supporting audio as cover media.

In 2006, Floriano De Rango [4] discusses about subjective and objective measurement methods available for evaluating quality of audio data. It also discusses the drawbacks like more time consumption, very slow and expensiveness of subjective measurement methods over objective measurement methods. Perceptual Evaluation of Speech Quality (PESQ) methodology is used to evaluate quality of the output audio data as it is more accurate than others and also reference original file is available for comparison.

3 Procedure and Implementation

In this section we describe how steganographic jamming is implemented and also discuss the various tools that were implemented for audio steganography and the effectiveness of the jamming technique in destroying the data hidden by various tools. A tool was developed based on the below methodology to perform steganographic jamming of cover audio file. The data used for steganographic jamming can be any file or text which is encrypted before embedding into the cover media. The encryption key is derived from the key that is used for double stegging. Password based encryption scheme "PBEWithMD5AndDES" from java library is used to encrypt the message bits to be hidden so that the data is randomized, also a random salt is introduced in the code for generation of key so as to prevent dictionary attacks. The encryption algorithm used is Data Encryption Standard (DES) and the hashing algorithm used is Message Digest (MD5). The decoding process is not explained here as it is not mandatory or necessary to decode the hidden data from the double stegged file as the sole purpose here is the destruction of the data hidden in the cover file.

3.1 Methodology

Method 1: LSB Embedding

1. Read the cover audio file and then a copy of file is generated which is used to hide data.
2. Read the data to be used to for steganographic jamming, if the data size is less than size of cover audio divided by sample size of audio, convert it into binary sequence of message bits.
3. The above message bits are then encrypted using password based encryption and the LSB of each sample of cover audio is replaced with the encrypted message bits.

4. The modified cover samples are then written to a file forming output audio signal.

Method 2: Variable LSB Embedding

1. Read the cover audio file and then a copy of file is generated which is used to hide data.
2. Read the data to be used to for steganographic jamming, if the data size is less than size of cover audio divided by sample size of audio, convert it into binary sequence of message bits.
3. The above message bits are then encrypted using password based encryption and the first and third LSB of alternate samples of cover audio is replaced with the encrypted message bits.
4. The modified cover samples are then written to a file forming output audio signal.

3.2 Tools Implemented

A number of steganographic tools are available in the market today. Here the discussion is only on the tools that can work with audio as cover media and are either open source or freeware. Following are the tools of the many tools that were able to successfully hide and extract the secret data into and from the audio cover file. There are other tools available like Steganofile, Xiao, S_Tools, Silenteye but we will not be experimenting on them as they either failed to hide data successfully in cover file or could not successfully extract hidden data from the cover file.

1. **Openpuff**: Openpuff supports the following audio formats wav, aiff, next/sun, and mp3. It also supports various encryption algorithms for security along with scrambling and whitening. The highest capacity of data that can be embedded is one by sixteenth of the file size of cover audio data. Openpuff uses LSB embedding algorithm along with data whitening (addition of random noise) for hiding data in cover media.
2. **DeepSound**: DeepSound supports AES encryption for security and wav and flac audio file formats for cover media. It uses either LSB embedding algorithm or variable LSB embedding algorithm depending on the size of the secret data that is to be hidden. The maximum embedding capacity offered by this tool is half the size of cover media file.
3. **Steghide**: Steghide is a command line interface (CLI) tool which uses graph theoretic approach to hide secret data into the audio cover file. The maximum capacity offered by this tool is one by sixteenth the size of cover file. It supports. au and wav audio file formats and also implements a checksum to check for integrity of extracted data.

4. **DeEgger**: This tool takes binary of given secret file and merges the binary code of cover file with the secret file in turn increasing the total size of the cover file and hence not making it a good steganographic tool. This tool hides the secret data in the cover file based on unique secret tags assigned to the data that is to be hidden. Hence deleting, or adding noise to the secret tag will render the hidden secret data corrupted and unable to recover. Since file size of cover media increases as the size of the secret data increases, the maximum capacity is not dependent on cover media.
5. **OurSecret**: This tool works similar to DeEgger in the way that it also increases the size of cover file and hence making it hard not to be suspicious as it hides binary data of secret file in between samples of audio cover data and also has same drawbacks as DeEgger.

4 Evaluation and Results

The following performance measures have been incorporated in this experiment after steganographic jamming has been implemented on the cover media using the tool developed on Java platform. None of the tools discussed above were able to recover the hidden files after performing steganographic jamming on the cover file which had data hidden in it.

The quality of output audio signal after steganographic jamming was rated using PESQ [13] which automatically rates audio signal quality objectively through software based on human perception of speech quality. It grades the resulting output based on MOS scale as shown in Table 1. which varies from 1 (Very Annoying) to 5 (Imperceptible). For this test three audio signals as described in Table 2 were composed. These signals were used as cover files and secret data was first hidden into it by the above tools and then steganographic jamming technique was performed. The resulting quality of audio signal was measured using PESQ reference implementation software P.862 [12] recommendation and values obtained are listed in Table 3. As the PESQ method is a full reference method of objectively testing quality of audio signal, the steganographic cover files were used as reference for measuring output with double stegged audio signal. Results shows that the output audio quality is well maintained for both the methods of embedding.

Table 1 ITU-T conversation opinion scale for MOS

Perceived distortion level	Quality	Grade
Imperceptible	Excellent	5
Perceptible but not annoying	Good	4
Slightly annoying	Fair	3
Annoying	Poor	2
Very annoying	Bad	1

Table 2 Description of audio signal used for test

	Music file 1	Music file 2	Music file 3
Number of channels	2	2	2
Sample rate	8000	16,000	16,000
Length of audio signal (s)	54.3	54.3	300
Bits per sample	16	16	16

Table 3 PESQ obtained after steganographic jamming of cover signal

Steganography tools		LSB embedding	Variable LSB embedding
OpenPuff	Music file 1	4.499	4.201
	Music file 2	4.389	4.1139
	Speech signal	4.201	4.031
DeepSound	Music file 1	3.61	3.32
	Music file 2	3.621	3.292
	Speech signal	3.502	3.04
StegHide	Music file 1	4.4	4.15
	Music file 2	4.36	4.13
	Speech signal	4.23	4.06
DeEgger	Music file 1	3.90	3.65
	Music file 2	3.84	3.6
	Speech signal	3.6	3.48
OurSecret	Music file 1	4.0005	3.625
	Music file 2	4.102	3.675
	Speech signal	3.85	3.312

Table 4 shows the bit error rate obtained when stego cover media was compared with double stegged cover media indicating corruption/partial destruction of hidden data and hence rendering the secret data unrecoverable by the stego tools and also maintains good output audio quality. As seen in Table 4 bit error rate is varying from 6.3 to 10.03% for steganographic jamming using LSB embedding and from 12.23 to 23.63% using Variable LSB embedding indicating good results steganographic jamming using variable LSB embedding technique because higher BER indicates the effectiveness of the algorithm without substantial reduction in MOS score.

Table 5 gives the NIST Signal to Noise ratio values computed using [10] of Original File, Cover File and the Double Stegged file which shows that there is very little degradation in audio quality or very less noise was introduced using steganographic jamming. And hence maintain good audio quality output.

Table 6 shows Waveform Amplitude Distribution Analysis—Signal to Noise Ratio [2] of original, steg and double stegged files. It can be seen that there is very little reduction in quality of audio signal after double stegging hence making it a good technique to effectively curb illegal and malicious use of steganography.

Table 4 BER comparison table

Steganography tools		LSB embedding	Variable LSB embedding
OpenPuff	Music file 1	0.091	0.1337
	Music file 2	0.095	0.1402
	Speech signal	0.08	0.1562
DeepSound	Music file 1	0.102	0.212
	Music file 2	0.956	0.2232
	Speech signal	0.1003	0.2363
StegHide	Music file 1	0.063	0.1223
	Music file 2	0.0593	0.1245
	Speech signal	0.0821	0.1335

Table 5 NIST SNR values (dB)

Steganography tools		Original file	Cover file	Double stegged file
OpenPuff	Music file 1	7.8	7.8	7.8
	Music file 2	7.5	7.5	7.5
	Speech signal	7.3	7.3	7.2
DeepSound	Music file 1	7.8	7.8	7.7
	Music file 2	7.5	7.3	7.2
	Speech Signal	7.3	7.3	7.2
StegHide	Music file 1	7.8	7.8	7.7
	Music file 2	7.5	7.5	7.5
	Speech signal	7.3	7.2	7.1

Table 6 WADA-SNR Values (dB)

Steganography tools		Original file	Cover file	Double Stegged file
OpenPuff	Music file 1	8.8	8.8	8.6
	Music file 2	8.9	9.1	8.7
	Speech signal	8.9	8.8	8.7
DeepSound	Music file 1	8.8	8.8	8.7
	Music file 2	8.9	8.5	8.4
	Speech signal	8.9	8.4	8.3
StegHide	Music file 1	8.8	8.8	8.6
	Music Ffe 2	8.9	9.1	8.7
	Speech signal	8.9	8.8	8.8

5 Conclusion

Double Stegging or Steganographic Jamming was implemented as technique to destroy embedded data in a given audio signal irrespective of the underlying steganographic algorithm used. It was proved that our method was effective as the

steganographic tools failed to recover the hidden data and also there was very less noise/distortion introduced in the cover media after steganographic jamming. Quality of the output signal was measured using Mean Opinion Score, Bit Error Rates and Signal to Noise Ratio. The above results also suggest that double stegging using variable LSB embedding algorithm was more successful in destroying secret data which was indicated by higher bit error rates.

References

1. Nutzinger, M.: Real time attacks on audio steganography. J. Inf Hiding Multimed. Signal Process. **3**(1), (2012) ISSN 2073-4212
2. Kim, C., Stern, R.M.: Robust Signal-to-Noise Ratio Estimation Based on Waveform Amplitude Distribution Analysis. In: Interspeech pp. 2598–2601 (2008)
3. Thwarting Audio Steganography Attacks in Cloud Storage Systems. In: International Conference on Cloud and Service Computing, pp. 259–265
4. De Rango, F., Tropea, M., Fazio, P., Marano, S.: Overview on VoIP: subjective and objective measurement methods, IJCSNS Int. J. Comput. Sci. Netw. Secur. **6**(1B), (2006)
5. Katzenbeisser, S., Petitcolas, F.A.P.: Information Hiding Techniques for Steganography and Digital Watermarking, pp. 121–148. Artech House, Boston, London (2000)
6. Basu, P.N., Bhowmik, T.: on embedding of text in audio-a case of steganography. In: Proceedings. Of International Conference on Recent Trends in Information, Telecommunication and Computing, pp. 203–206 (2010)
7. Shelley, L., Picarelli, J.: Methods not motives: Implications of the convergence of international organized crime and terrorism. Int. J. Police Pract.Res. **3**, 305–318 (2002)
8. Al-Ani, Z.K., Zaidan, A.A., Zaidan, B.B., Alanazi, H.O.: Overview: Main fundamentals for steganography. J. Comp. **2**(3), 158–165 (2010)
9. Adee, S.: Spy vs. spy. http://spectrum.ieee.org/computing/software/spy-vs-spy/
10. Ellis, D.: Lab for Recognition and Organization of Speech and Audio (LabROSA.) http://labrosa.ee.columbia.edu/projects/snreval/
11. Karadogan, I., Das, R.: An examination on information hiding tools for steganography. Int. J. Inf. Secur. Sci. **33**, 200–208
12. PESQ reference implementation software P.862 recommendation. http://www.itu.int/rec/T-REC-P.862-200511-I!Amd2/en
13. Recommendation ITU-T P. 862, Perceptual evaluation of speech quality: an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs, Technical report (2001)
14. Premalatha, P., Amritha, P.P.: Optimally locating for hiding information in audio signal. Int. J. Comput. Appl. (0975–8887) **65** (14), (2013)