New Technique for Hiding Data in Audio File

Mohammed Salem Atoum [†] and Osamah Abdulgader Al- Rababah [††], Alaa Ismat Al-Attili[†††]

[†]Computer Science Student in Faculty of Computer Science & Information SystemUniversiti Teknologi Malaysia81310 Johor Bahru, Johor

[††]College of Alkamel - Computer Department -King Abdulaziz University-110Alkamel 21931 Jedda, Saudia Arabia

[†††]AL-Zahra College for Women (University College) - IT Department P.O. BOX: 3365 POSTAL CODE: 111 Airport Heights- Muscat – Oman

Summary
Hiding information inside audio files becomes a challenging discipline, since the Human Auditory System (HAS) is highly sensitive. One of the main obstacles of the data hiding in audio is to develop a system which has the quality to include a big amount of data and without affecting the quality of sound.
This paper proposes an information-hiding method to hide more information into audio media file (MP3). The bits of information will be hidden between frames (BF) in MP3 file.
In the experimental results, we hide more characters in audios and extract them correctly. The audios with secret information are indiscernible to human ear.

Key words:
Data Hiding, Steganography, Audio File.

1. Introduction

Secret digital music MP3 files, are important and popular audio compression standard in the Internet. MP3 uses the destructive compression technologies to achieve high compression rate and the original file is shrunk to a very small size. [1] MP3 compression [2, 3], is too time-consuming [4]. To protect digital media files, researchers propose and improve many data-hiding algorithms, which are known as steganographic algorithms.
Steganography means covered writing. Steganography can be classified into three types: pure steganography, secret key steganography, public key steganography [5].
All data-hiding applications require hiding algorithms on the sender side and a detector mechanism or algorithm on the receiver side. The hiding hidden message or data in the can be retrieved by authorized people only.
The most crucial parameters of the data-hiding applications are: security, reliability, invisibility, complexity, and data-hiding capacity, these parameters are mostly related to each other [6].
In recent years, various techniques for steganography in digital audio with various purposes have been developed [7-11].
In [7] Presented spread spectrum audio data hiding. This method is different from the traditional way in that it is based on dealing with the original sound by sub-and phase shifting. The proposed scheme can provide the robust aerial data transmission compared to the traditional schemes.
In [8] Proposed a method by compressing MP3 and modifying spectrum values of audio layers to embed secret information into audios especially music files in MP3.This method can extract secret information without original audio and it has characteristics that information hiding technique must be responsible for prerequisites.
In [9] Proposed data hiding algorithm based on Piecewise Linear Haar (PLHaar) transform. The PLHaar transform maps an n-bit integer to another n-bit integer which don't guarantee under flow or over flow to occur. Secret information is embedded into the Least Significant Bits (LSB's) of the PLHaar transform coefficients. Although the proposed algorithm has a slight decline in the quality of the stego audio compared with the algorithm based on Haar transform, it showed an ability of retrieve embedded secret information correctly without any additional operation. The researcher recommended expanding upon the PLHaar transform to reversible data hiding for audio.
In [10-11] proposed methods of audio steganography based on modification of least significant bits (LSB).
In this paper, we propose new steganographic method based on a novel data-embedding algorithm. The main objective of the study is to embed data between MP3 frames.

2. Tables, Figures and Equations

This method is based on embedding data between frames in MP3 file.
The major requirements of this method:
1. k-lite multi-media player program (windows media player classic).
2. The MP3 file must be of CBR type only.
The following algorithm demonstrates the way of embedding and extracting encrypted text file

Manuscript received April 5, 2011
Manuscript revised April 20, 2011

2.1 Embedding

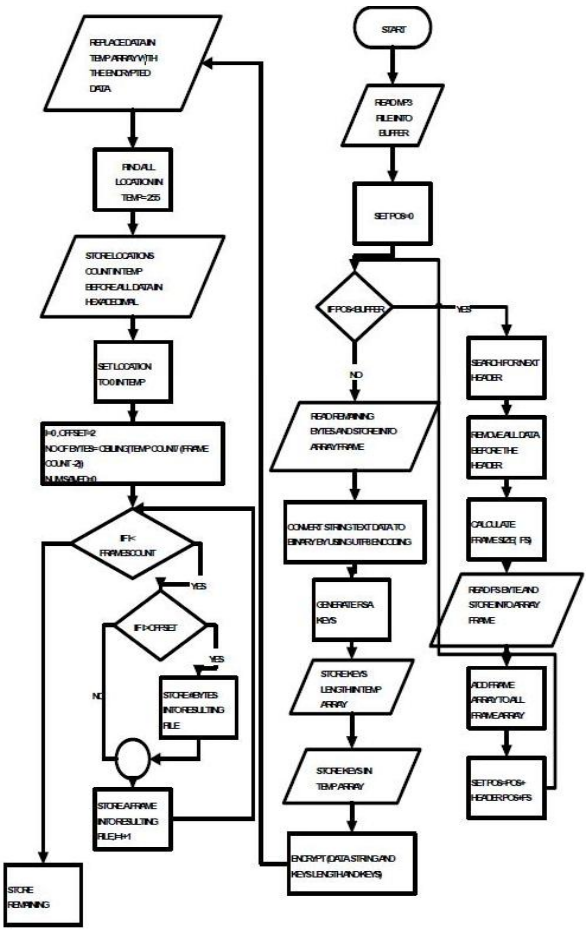

Fig.1. shows the flow chart for Embedding text of (BF)

1) Read MP3 file into buffer
2) Pos=0
3) While (Pos < Buffer)
{
Search for next header (header position)
Remove all data before the header
Calculate FS
Read FS byte and store into array frame
Add frame array to all frame array
Pos = Pos + Header Pos + FS
}
4) Read remaining and store
5) Convert string text data to binary by using UTF8 encoding
6) Generate RSA keys
{
Store keys length in temp array
Store keys in temp array
}
7) Encrypt (data string and keys length and keys) and store in temp
8) Find all location in temp=255
{
Store locations count in temp before all data in hexadecimal
Store all location in hexadecimal
}
9) I=0 , Offset=2
NOB = Ceiling(temp count/ (frame count -2))
Num Saved =0
While I < All frames count
{
If I> Offset then
{
If (I*NOB + NOB <= temp count + Offset) then
Store NOB in result
Num saved + = NOB
}
Else If num saved < n then store from (I-offset) NOB to end of temp
}

Store frame I in result
Store remaining

2.2 Extracting

1) Read MP3 file into buffer
2) Pos=0
3) While (Pos<Buffer)
{
Search for next header (header position)
Save all data before the header
Calculate FS
Read FS byte and store into array frame
Add frame array to all frame array

Pos = Pos + Header Pos + FS
}
4) Read hexadecimal from no of location.
5) Read hexadecimal from all location.
6) Convert hexadecimal into 255.
7) Read (key length and keys.)
8) Decrypt data.
9) Convert data to string using UTF8 encoding .
10) Show text.

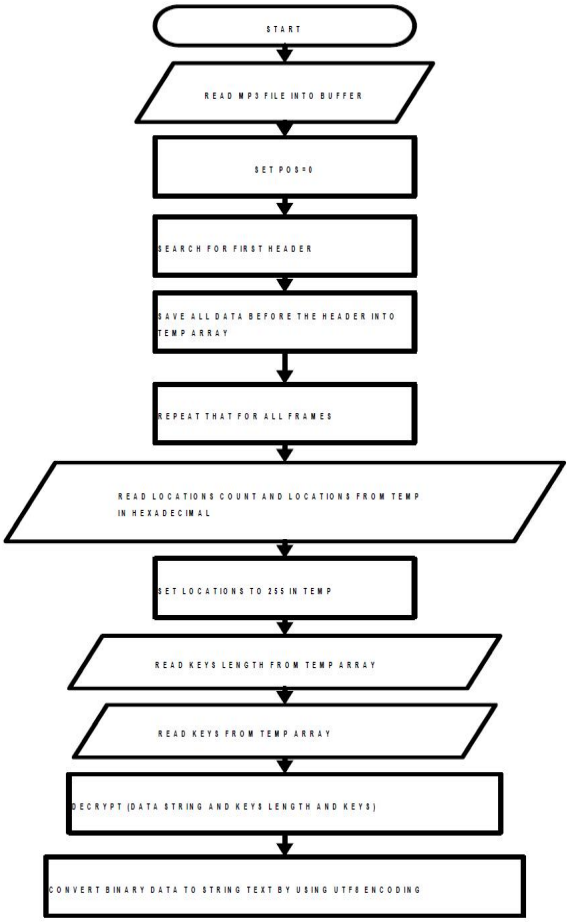

Fig.2. Flow Chart for extracting text of (BAF)

3. Experimental Results and Analysis

The optimum evaluative criterion is embedding as much text as possible with least time and high audio quality. A number of tests has been conducted on the proposed method and Before All Frames method (BAF), which summed up by embedding all encrypted text before the first frame in MP3 file.
Different criteria have selected to measure either of the best previous methods. Three scales have been used to evaluate those methods.
1- Time of embedding and extracting.
2- Size of file embedded.
3- Audio quality.

Table (1) shows the results of methods BAF and BF using RSA algorithm for embedding files from both time and size point of views:

Table (1) Embedding with RSA algorithm

size file embed KB	#byte	#frame	original size	results file size	time of embedding BF(Sec)	time of embedding BAF(Sec)
1KB	3175	12281	5138432	5136260	1.944	1.6719
2KB	5279	12281	5138432	5138364	1.94575	1.765625
4KB	9527	12281	5138432	5142612	1.946375	1.9015
6KB	13703	12281	5138432	5146788	1.984375	1.923846875
8KB	18047	12281	5138432	5151132	1.99945	
10KB	22103	12281	5138432	5155188	2.01145	
20K	43343	12281	5138432	5176428	2.02112	
40K	85367	12281	5138432	5218452	2.029923	
80K	170543	12281	5138432	5303628	2.0337432	
100K	212687	12281	5138432	5345772	2.109365	
200K	424535	12281	5138432	5557620	2.568749	
400K	847567	12281	5138432	5980652	3.118749	
800K	1693999	12281	5138432	6827084	4.243749	
1024KB	2166935	12281	5138432	7300020	4.88749	
2048KB	4331847	12281	5138432	9464932	8.046873	
4096KB	8666335	12281	5138432	13799420	14.26561	
40960KB	86609191	12281	5138432	91742276	134.2	

From table (1) it is clearly seen that the embedded file size will affect the cover file size since the embedding process is to insert the hidden text between frames.
Although this method allow embedding large files within cover file, but it is recommended to embed relatively small files (with respect to cover size) in order not to make the stego file suspicious.
Figure (3) shows the results of the analysis as shown in table (1)
BF can embed 40960 KB in a short time and maintain the same sound quality where as BAF did not exceed 6 KB.

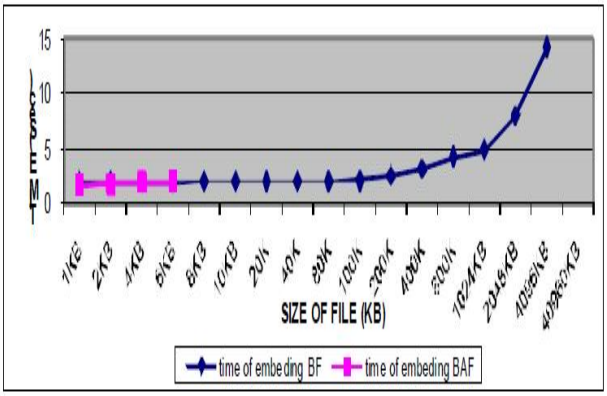

Figure (3) Embedding with RSA algorithm

Table (2) shows the results of methods BAF and BF using RSA algorithm for extracting files time.
Figure (4) shows the analysis results of table (2)
The figure above shows that the time consumed in extracting increases with the size of the file. BF proved its ability in extracting files in a short time.

Table (2) Extracting with RSA algorithm

size file extract KB	#byte	#frame	Original file size	Result file size	time of extracting BF(Sec)	time of extracting BAF(Sec)
1KB	3175	12281	5136260	5138432	1.284375	1.225
2KB	5279	12281	5138364	5138432	1.334375	1.328125
4KB	9527	12281	5142612	5138432	1.465625	1.4125
6KB	13703	12281	5146788	5138432	1.578125	1.528125
8KB	18047	12281	5151132	5138432	1.7	
10KB	22103	12281	5155188	5138432	1.809375	
20K	43343	12281	5176428	5138432	2.4313	
40K	85367	12281	5218452	5138432	3.699999	
80K	170543	12281	5303628	5138432	6.1375	
100K	212687	12281	5345772	5138432	7.346875	
200K	424535	12281	5557620	5138432	13.609375	
400K	847567	12281	5980652	5138432	25.94688	
800K	1693999	12281	6827084	5138432	50.63125	
1024KB	2166935	12281	7300020	5138432	64.9	
2048KB	4331847	12281	9464932	5138432	128.5	
4096KB	8666335	12281	13799420	5138432	255	
40960KB	86609191	12281	91742276	5138432	2618.2	

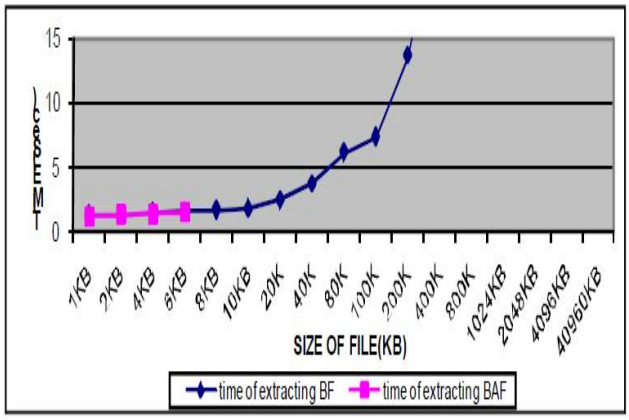

Figure (4) Extracting with RSA algorithm

4. Conclusion:

Steganography is really an interesting subject and is outside the mainstream of cryptography and system administration that most of us deal with day after day. But it is also quite real; this is not just something that is used in the lab or an arcane subject of study in academia.

In general, the suggested method satisfies the capacity and complexity of steganography properties. In addition to that, the suggested method for hiding is robust against noise. It is also considered highly secure since data is encrypted using RSA algorithm before embedding data which makes the system secure especially against passive attack.
To overcome the shortcomings of the proposed method, we suggest: Before All Frame (BAF).
The merits of Before All Frames (BAF):
Extracting the text is faster than other methods for limited size, because the whole text lies in the beginning.
The song can be played on all multi-media audio players.
High audio quality.
A relatively good text file size.
It can be played whether the MP3 file encodes VBR or CBR .
Taking in consideration that the major issue is the number of frames in the cover file not the file size (as number of frames will not actually depend on audio file size only but also on frame size).

References

[1] W. Jonker, J.-P. Linnartz, "Digital Rights Management in Consumer Electronics Products", IEEE Signal Processing Magazine, Vol. 21, No. 2, pp.82-91, 2004.
[2] D. Pan, “A tutorial on MPEG/Audio compression”, IEEE Multimedia, 2(2), pp. 60-74, 1995.
[3] E. Ambikairajah, AG Davis and W. Wong, “Auditory masking and MPEG-1 audio compression”, IEE Electronics & Communication Engineering Journal, 9(4), pp.165-175, 1997.
[4] Cheng-Te Wang, Tung-Shou Chen and Wen-Hung Chao, “A new audio watermarking based on modified discrete cosine transform of MPEG/Audio Layer III”, Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control, pp.984-989,2004.
[5] Cheng-Te Wang, Tung-Shou Chen and Wen-Hung Chao, “A new audio watermarking based on modified discrete cosine transform of MPEG/Audio Layer III”, Proceedings of the 2004 IEEE International Conference on Networking, Sensing & Control, pp.984-989, 2004.

[6] Wang H, Wang S. Cyber warfare: steganography vs. steganalysis. Communications of the ACM 2004;47(10).
[7] H. Matsuka, “Spread Spectrum Audio Steganography using Sub-band Phase Shifting,” IEEE Int. conf. Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP' 06), pp. 3-6, Dec. 2006 , Pasadena, CA, USA.
[8] B. Deng and J. Tan and B. Yang and X. Li, "A Novel Steganography Method Based on Modifying Quantized Spectrum Values of MPEG/Audio Layer III” , Proceedings of the 7th WSEAS International Conference on Applied Informatics and Communications, Athens, Greece, IEEE, pp.325-330,August 24-26, 2007.
[9] Diqun Yan and Rangding Wang,” Reversible Data Hiding for Audio Based on Prediction Error Expansion”, CKC Software Laboratory, University of Ningbo, Ningbo 315211, China, International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IEEE, pp.249252,2008.
[10] N. Cvejic, T. Seppanen, “Increasing robustness of LSB audio steganography using a novel embedding method,” In Proc. IEEE Int. Conf. Info. Tech.: Coding and Computing, Vol. 2, pp. 533-537, April 2004.
[11] N. Cvejic, T. Seppanen, “Increasing the capacity of LSBbased audio steganography.” IEEE Workshop on Multimedia Signal Processing, pp. 336-338, 2002.