An Optimized Method for Concealing Data using Audio Steganography

Md. Shafakhatullah Khan
Asst. Professor
Dept of IT and CA
Aurora's Engineering College,
Bhongir, Nalgonda, A.P, India.

V.Vijaya Bhasker
B.Tech scholar
Dept of CSE
Aurora's Engineering College,
Bhongir, Nalgonda, A.P, India.

V. Shiva Nagaraju
B.Tech scholar
Dept of CSE
Aurora's Engineering College,
Bhongir, Nalgonda, A.P, India

ABSTRACT
In this paper, we propose a new approach which is sophisticated for concealing the data. We are using Audio Steganography to make the confidential data more secure, so that the data cannot be tracked and modified by the intruders. To make the confidential data secure we are using efficient and reliable algorithms. This paper describes how the data is secured form the intruders even though they trace the audio file which contains the confidential data.

General Terms
Audio Steganography, Security, Key Management.

Keywords
Audio Steganography Advanced Encryption Standard, Security, Spectrum Spread, Key Management.

1. INTRODUCTION
Since a decade internet is ruling for each and every means of communication. The emerging of the technologies made internet more user-friendly. Because of its rapid growth everyone is sharing confidential data over internet. The possibility of pirate access to private information of the people and companies is also increased. To overcome this problem we are proposing a novel approach. Steganography is a powerful tool which increases security in data transferring and archiving. In the case of Steganography the confidential data is first encapsulated within another object which is called "cover object", to form "stego object" and then this new object can be transmitted or saved. It causes the existence of the confidential data and even its transmission becomes secure and safe [1].

Since few years so many techniques for Steganography in digital audio with various purposes have been developed and introduced [2-9]. Among the techniques of Steganography, audio Steganography is the best technique which is based on modification of least significant bits (LSB). This paper presents an enhancement of spread spectrum audio data hiding methods. In this proposed system we are using phase shifting in audio signals to reduce the correlation with PN (Pseudo-random Noise) signal per each sub-band. The embedded data signals are easily identified when the audio is de-spreading the compound signals. The paper gives the subjective test results and the measurements of noise resiliency. The spread spectrum audio steganography reduces the error probability by increasing the spreading rate and coding gain. It takes long time to send 1 bit of information when spreading rates are high, because of that data transmission speed decreases. In that scenario to increase the efficiency of the performance, we need to use low spreading rate in encoding method which provides good robustness to the embedded private information [9].

The basic idea behind this paper is to provide an optimized method for concealing the private data from intruders and sent to the destination in a safer and secure manner. Though it is well modulated software it has been limited to certain restrictions. The quality of sound depends on the size of the audio which the user selects and length of the message. Even though it shows changes in bit level deviations in the frequency chart, as a whole we cannot determine the change in the audio [10].

The basic theme of this paper is to prevent the data from the intruders. There are so many data encryption algorithms are proposed to hide the data but almost all the proposed algorithms has few disadvantages. The maximum possible ways to hack the data is using the following types of attacks:

- Brute force attack
- Differential cryptanalysis
- Linear cryptanalysis
- Davis attack
- Side channel attacks

The above mentioned attacks are mostly dangerous attacks which are used by the intruders to reveal the confidential data. In this paper we are talking about one of the data encryption which is advanced and sophisticated than the technique used by R. Sridevi et al [10]. Using that method we are going to overcome the above problems.

We are using audio steganography in this paper instead of image steganography because audio steganography is more challenging than image steganography. In this work initially we do the encryption of the data using Advanced Encryption algorithm. Then that data is embedded into the audio file and that audio file will be encrypted using frequency hopped spread-spectrum technique for steganography in digital audio [11].

2. RELATED WORK

2.1 Advanced Encryption algorithm
Advanced Encryption Standard (AES) is a specification for the encryption of electronic data. This technique is initially adopted by the U.S. Defense, now it is used worldwide. It replaces DES [12].

2.1.1 AES algorithm:

AES is the norms of electronic data encryption used by the U.S. National Institute of Standards and Technology. It is a symmetric block cipher system. This algorithm uses replace / exchange network. The data block length and key length are variable. Three key lengths: 128,192, 256, whose iteration cycle Nr is 10, 12 and14 round respectively, are used. The AES algorithm mainly includes three aspects: round change, turns and key expand. Each round transformation is composed by the non-linear layer, the linear mixture layer, the addroundkey layer.

2.1.2 Security Requirements in WSN:

Sastry and Wagner observed the merits and limitations of security aspects of the IEEE 802.15.4 specification. The specification allows a maximum of 255 Access Control List (ACL) entries, where within ACL there is no support for group keying and pair wise keying in IEEE 802.15.4 specification. The specification suffers from IV Management, Key management problems and insufficient integrity protection. It should be noted that the IEEE 802.15.4 API indicates two clear directions. One of them is to go with the specification itself without adding more security patches, and the other is to adopt add-on security service on top of the API according to application's requirement. Nevertheless, one has to see that the combined security suite must consider some basic security requirements, AES encryption process is shown in Fig 1.

2.1.3 Realization of AES algorithm:

AES is iterative block encryption algorithm that has the variable length data block and the variable length key. After several rounds of block transform, an intermediate state of transformation is generated. The state can be expressed as a two-dimensional byte array, which has 4 line and Nb column (Nb is the data block length divided by 32). Keys can also be expressed as a two-dimensional byte array, which has 4 line and Nk column (Nk is the key block length divided by 32). The transform round number Nr is decided by Nb and Nk, as shown in Table 1.

Each round transformation contains 4 steps, which are bytesub, shift row, mix column, addroundkey respectively.

1. Bytesub is non-linear permutation based on Sbox. It is used to map each byte of input or intermediate state into another byte through a simple table look-up operation. Then elements of rows and columns of Sbox are taken out as output.
2. ShiftRow is a line-based operator of cyclic shift. The last 3 rows of ShiftRow state is followed by cyclic shift in turn. For example, when Nb=4, the 0^{th} line is cyclic left 0 bit, the 1st line is cyclic left 1 bit, the 2nd line is cyclic left 2 bit, the 3rd line is cyclic left 3 bit.
3. Mixcolumn is a replacement operation. It runs plus and multiplication operation of the value of state-byte in mathematical domain. So each byte is replaced by the result.
4. The state adjustment result is obtained by XOR of the addroundkey with a roundkey.

$$(a_{i,j})4\times 4 \otimes (k_{i,j})4\times 4 = (b_{i,j})4\times 4$$

Where $a_{i,j}$ is each byte of state data block Nb=4, $k_{i,j}$ is each byte of state round key Nk=4, $b_{i,j}$ is the result of XOR. Round key is derived from the key, including key Expansion and round key. The realization of the process is as follows [3].

1. The key is expanded to inflation key.
2. The sum of bit number of round key is equal to the value that is the block length multiplied by a round number added 1. For example, when Nb=4, Nr=12, the sum of bit number is 128× (12+l) =1664 bit.
3. The round key is taken out from inflation key. The 1st round key is composed by the first Nb characters, the 2nd round key by next Nb characters and so on [13].

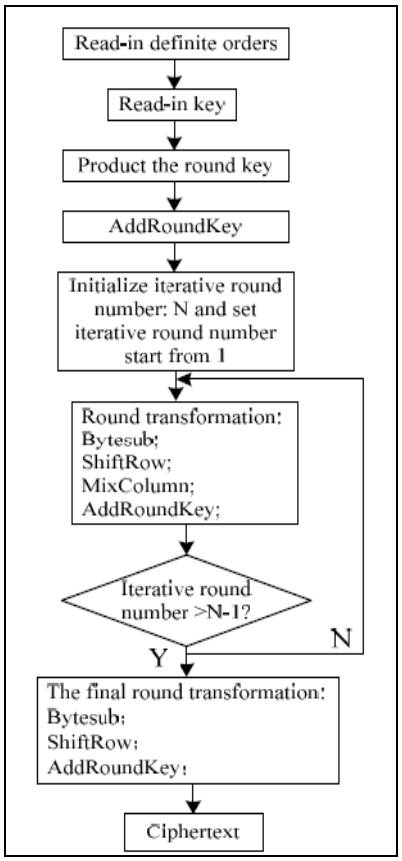

Fig 1: AES encryption process scheme

Table 1: The Relationship between the transform round number Nr and Nb, Nk

Nr	Nb=4	Nb=6	Nb=8
Nk=4	10	12	14
Nk=6	12	12	14
Nk=8	14	14	14

2.2 Spread Spectrum Method

Basically this method calculates the frequency masking threshold using psycho acoustic model, data signal is spread by a M-sequence code, and the spread signal is embedded in audio below the frequency masking threshold. M-sequence codes have good autocorrelation properties where the autocorrelation function has peaks equal to 1 at 0, N, 2N… (approximately 1/N elsewhere). Because of these periodic peaks, the M-sequence code is self-clocking, so the receiver can easily synchronize the data frame and retrieve the embedded data by de-spreading with the same M-sequence code. Fig.2. shows an example of the original audio spectrum and the frequency masking threshold. The frequency masking threshold is calculated at each critical band based on the psycho acoustic model [14].

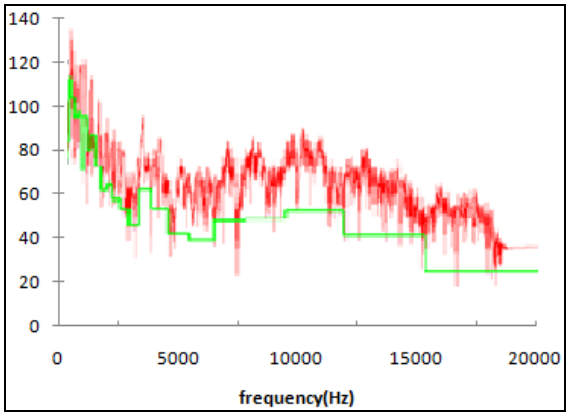

Fig.2. Frequency Masking Threshold

Next, data signal is spread by an M-sequence code. The spectrum of the spread signal is shown as Fig.3. The spectrum is almost flat in the wide frequency range. Compared to the masking threshold, some bands have higher spectrum of the spread data signal than the masking threshold, other bands have much lower spectrum than the threshold. The higher spread signal is audible to human ears, and the lower spread signal can be still increased up to the threshold. The spread signal of the flat spectrum is filtered with the masking filter that has the same amplitude response as the masking threshold. The spectrum of the output signal has the same spectrum as the masking threshold. Then the output signal is power-adjusted below the masking threshold and added to the original audio.

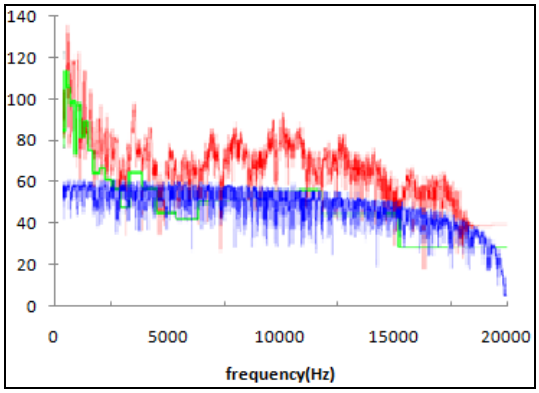

Fig.3. Spread Data Signal.

Fig.4 shows the spectrum of the output signal. In this way, the power of the spread signal can be maximized below the frequency masking threshold.

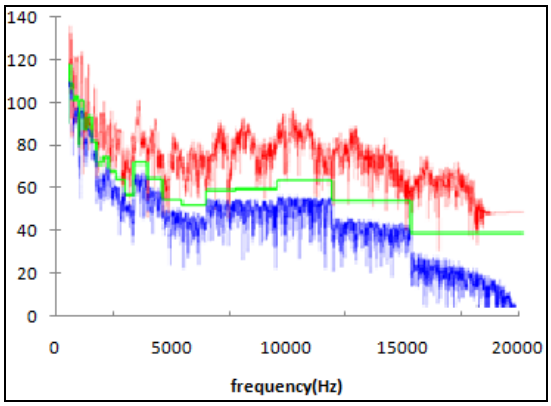

Fig.4. Colored Spread Signal.

3. METHODOLOGY

3.1 Problem Definition

Enhanced Audio Steganography is a method of hiding the message in the audio file of any formats. EAS provides an easy way of implementation of mechanisms when compared with audio steganography. Apart from the encoding and decoding in Audio steganography, EAS contain extra layers of encryption and decryption. The four layers in EAS are:

- Encoding
- Decoding
- Encryption
- Decryption

Encoding is a process of hiding the message in the audio. Modified LSB (Least Significant Bit) Algorithm is used to encode the message into audio. It performs bit level manipulation to encode the message. The following steps are:

a. Receives the audio file in the form of bytes and converted in to bit pattern.
b. Each character in the message is converted in bit pattern.
c. Replaces the LSB bit from audio with LSB bit from character in the message.

Powerful encryption algorithm is used to encrypt the message before encoding for further security purpose. The following steps are used to encrypt the message:

- Adding all ASCII values of characters in the key given by user.
- Converting the sum into bit pattern
- Performing logical operation to the bit pattern.
- Adding to the encoded character.

For more security enhancement the encoding is done only when the byte which is received from the audio is 254 or 255.This selection of particular bytes for encoding will reduce the lack in quality of audio after encoding. It can be proved by seeing the frequency chart indicating the deviations happened after encode. Thought it shows bit level deviations in the chart as a whole the change in the audio cannot be determined.

The above mentioned details in the problem definition is about the existing system which has few disadvantage like degradation in sound quality, on the other hand the algorithm used to encrypt the data is easily broken.

When hiding information inside Audio files the technique usually used is low bit encoding which is somewhat similar to LSB that is generally used in Images. The problem with low bit encoding is that it is usually noticeable to the human ear, so it is a rather risky method for someone to use if they are trying to mask information inside of an audio file.

3.2 Solution

In the proposed system first we are encrypting the data using the Advanced Encryption Standard algorithm which is highly sophisticated and for cryptographers, a cryptographic "break" is anything faster than a brute force - performing one trial encryption for each key. Thus, an attack against a 256-bit-key AES requiring 2200 operations (compared to 2256 possible keys) would be considered a break, even though 2200 operations would still take far longer than the age of the universe to complete. The largest successful publicly-known brute force attack against any block-cipher encryption has been against a 64-bit RC5 key by distributed.net [15].

A. Algorithm

```
import java.security.*;
import javax.crypto.*;
import javax.crypto.spec.*;
import java.io.*;

public class AES{
   public static String asHex(byte buf[]){
   StringBuffer strbuf = new StringBuffer(buf.length*2);
   int i;
   for(i=0; i<buf.length; i++){
      if(((int)buf[i] & 0xff)<0x10)
                  strbuf.append("0");
      strbuf.append(Long.toString((int)buf[i]&0xff, 16));

      } // end of for loop
   return strbuf.toString();

   }// end of asHex()

public static void main(String args[]){
   String message = "Audio Steganography";

   // Get the key generator
   KeyGenerator kgen =KeyGenerator.getInstance("AES");
   kgen.init(128);
   // Generate the secret key spaces.
   SecretKey skey = kgen.generateKey();
   byte[] raw = skey.getEncoded();
   SecretKeySpec skeySpec = new SecretKeySpec(raw,
   "AES");

   // Instantiate the cipher
   Cipher cipher = Cipher.getInstance("AES");
   cipher.init(CipherENCRYPT_MODE, skeySpec);
   byte[] encrypted = cipher.doFinal((args.length= = 0 ?
   "This is just an example":args[0]).getBytes());
   System.out.println("Encrypted String" +
   asHex(encrypted));
   cipher.init(Cipher.DECRYPT_MODE, skeySpec);
   byte[] original = cipher.doFinal(encrypted);
   String originalString = new String(original);
   System.out.println("Original String" + originalString+ ""
   + asHex(original));

   } // end of main()
} // end of class
```

After the encryption of the data we embed the data which is encrypted in an audio file then we encrypt the audio file using the Spread Spectrum technique.

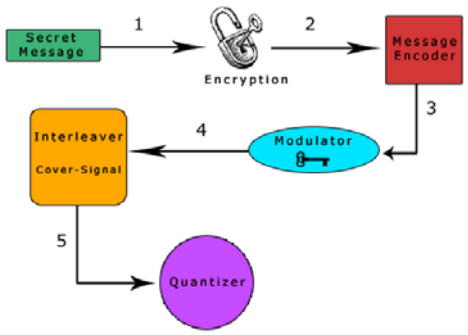

Fig: 5 Illustration of Spread Spectrum Process

1. The secret message is encrypted using a symmetric key, k_1.
2. The encrypted message is encoded using a low-rate error-correcting code. This step increases the overall robustness of the system.
3. The encoded message is then modulated with a pseudorandom signal that was generated using a second symmetric key, k_2, as a seed.
4. The resulting random signal that contains the message is interleaved with the cover-signal.
5. The final signal is quantized to create a new digital audio file that contains the message.
6. This process is reversed for message extraction.

3.3 Simulation

The proposed technique is simulated in java with the data and audio files. The data is encrypted and is embedded into the audio file then that audio file is encrypted. By using this technique we find no difference in the size and the quality of the audio before encryption and after encryption. The tool we have used to

develop this application is NetBeans IDE, The base IDE includes an advanced multi-language editor, debugger and profiler integration, file versioning control, and unique developer collaboration features.

4. EXPERIMENTAL RESULTS

4.1 Comparison of the embedded and encrypted audio file after and before

In Fig: 6 it is shown that an audio file is selected before we are embedding the confidential encrypted data into it and the frequency of the audio file before embedding into audio file is taken which results in Fig:7. Fig: 8 shows the selected audio file in which encrypted confidential is embedded and it is played, after playing the audio file we recorded the frequency of the audio which results in Fig:9.

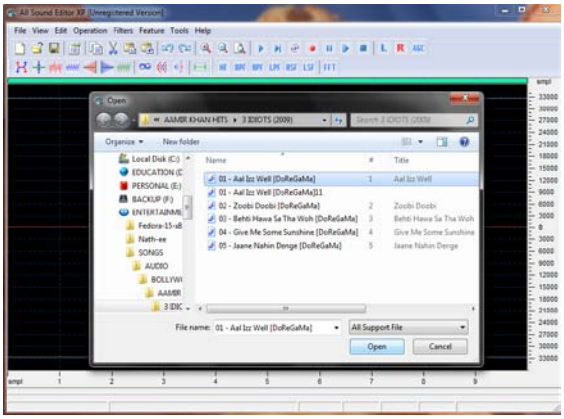

Fig: 6 Selection of the file before embedding data and encrypting the audio file

Fig: 7 The frequency of the audio file before encrypting and embedding the data into it.

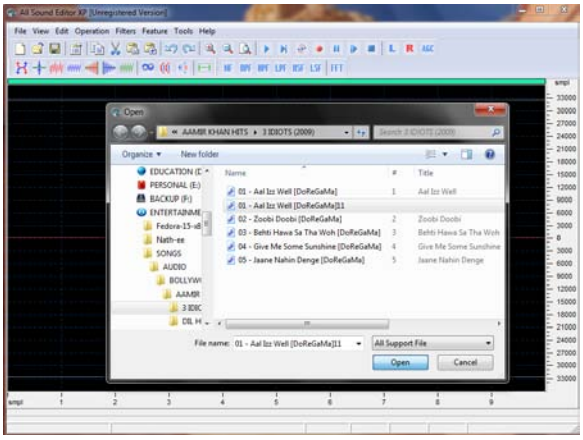

Fig: 8 Selection of the file after embedding data and encrypting the audio file

Fig: 9 The frequency of the audio file after encrypting and embedding the data into it.

To check the change in the audio before and after the encryption of the audio file which contains the encrypted data, we use the tool frequency generator the above results in Fig: 7 and Fig: 9 shows that there is no difference in the frequencies of the audio files.

By the spread spectrum technique of audio steganography we are proving that even though the audio file is encrypted and embedded with the confidential data it is very difficult to trace and at the same time it will take the universe time to hack the encrypted data.

5. CONCLUSION

This proposed system is to provide a good, efficient method for hiding the data from hackers and sent to the destination in a safe manner. This proposed system will not change the size of the file even after encoding and also suitable for any type of audio file format. Encryption and Decryption techniques have been used to make the security system robust.

6. REFERENCES

[1] Ahmad Delforouzi, Mohammad Pooyan, "Adaptive Digital Audio Steganography Based on Integer Wavelet Transform".

[2] 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.

[3] N. Cvejic, T. Seppanen, "Increasing the capacity of LSBbased audio steganography." *IEEE Workshop on Multimedia Signal Processing*, pp. 336-338, 2002.

[4] S.S. Agaian, D. Akopian, O. Caglayan, S. A. D'Souza, "Lossless Adaptive Digital Audio Steganography," In Proc. IEEE Int. Conf. Signals, Systems and Computers, pp. 903-906, November 2005.

[5] N. Cvejic, T. Seppanen, "A wavelet domain LSB insertion algorithm for high capacity audio steganography," In Proc. IEEE Digital Signal Processing Workshop, *Callaway Gardens*, GA, p. 53–55, October 2002.

[6] K. Gopalan, "Audio steganography using bit modification," Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, Vol. 2, pp. 421-424, April 2003.

[7] P. Bao and X. Ma, "MP3-Resistant Music Steganography based on Dynamic Range Transform**,"** *IEEE Int. Sym.* Intelligent Signal Processing and Communication Systems**,** pp. 266-271, Nov. 18-19, 2004, Seoul, Korea.

[8] R. A. Santosa, P. Bao, "Audio-to-image wavelet transform based audio steganography," IEEE Int. Symp. pp. 209-212, June 2005, Zadar, Croatia.

[9] 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.

[10] R SRIDEVI, DR. A DAMODARAM, DR. SVL.NARASIMHAM "Efficient Method or Audio Steganography by modified LSB Algorithm and Strong Encryption Key with Enhanced Security". In Proc. JATIT. PP: 768-771, 2005-2009.

[11] Sterling, M.; Titlebaum, E.L.; Xiaoxiao Dong; Bocko, M.F.; "An adaptive spread-spectrum data hiding technique for digital audio". Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). Publication Year: 2005 , Page(s): v/685 - v/688 Vol. 5

[12] Westlund, Harold B. (2002). "NIST reports measurable success of Advanced Encryption Standard". Journal of Research of the National Institute of Standards and Technology.

[13] Hsiang H., and ShihW. (2009). Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment. *Computer Standards & Interfaces*, doi: 10. 1016j. csi.2008.11.002.

[14] J.Johnston, "Transform Coding of Audio Signals Using Perceptual Noise Criteria", IEEE Journal on Selected Areas in Communication, vol.6, pp.314-323, February, 1988.

[15] Ou, George (April 30, 2006). "Is encryption really crackable?". Ziff-Davis. Archived from the original on August 7, 2010. Retrieved August 7, 2010.