

Audio Steganography Using Least Significant Bit and AES Cryptographic Algorithm

Nurul Hidayah Ahmad Zukri^{1,*}, Nur Khairani Kamarudin¹, Mohd Enif Izraf Ishak² and Nor Aimuni Md Rashid²
¹Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Perlis, Kampus Arau, 02600 Arau, Perlis, Malaysia

²Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA Cawangan Melaka, Kampus Jasin, 77300 Merlimau, Melaka, Malaysia

*corresponding author: ¹hidayah1278@uitm.edu.my

ARTICLE HISTORY

ABSTRACT

Received
1 October 2018

Accepted
12 December 2018

Available online
30 December 2018

In the era of technology, massive amount of data have been stored and exchanged over the internet. One of the most important factors of data communication has been the security of the credential information. Cryptography is a technique for securing the secrecy of the information. Many different techniques have been developed under the cryptography umbrella. However, it is not enough to keep the contents of a message secret. Thus, another algorithm should be implemented to keep the existence of a message secret. Steganography is an interesting and effective method for concealing data throughout the history. In this paper, Least Significant Bit (LSB) algorithm has been proposed. This algorithm can help users to protect their confidential information during transmission by embedding the data inside the audio .wav file. This application implements an Advance Encryption Standard (AES) to enhance the steganography security by encrypting the information and as such, only the intended recipient can read the data. The proposed method is analyzed in terms of audio similarity analysis between the original file and the audio steganography file. The result shows about 100% of similarity between both files. File integrity is also measured and the results show the hashing value of the encrypted file was different compared to the original file.

Keywords: Audio Steganography, Cryptography, LSB, HAS, AES.

1. INTRODUCTION

The advent of the internet technology has increased the cybercrime cases especially in data stealing. The Symantec Intelligence reported that the attacker targets the valuable and confidential data to attack on [1]. The attackers do not just take randomly from a large amount of data, instead they select the target to attack or steal interest information such as document file or message string. One of the solutions which came to the rescue is the cryptography. But cryptography alone is not enough to protect the credential information because the attacker is still aware about the existence of the message. Consequently, the attacker would plan to launch the cryptanalysis attack [2]. The growing risk in cyber security need to resolve quickly as data is the important assets [3]. In order to protect data, steganography technique was introduced [4].

The word steganography is derived from the Greek words “Steganos” mean “covered or secret” and “graphy” mean “writing or drawing” defining it as “secret writing” [12]. Steganography is an art of hiding the presence of information inside other media such as image, video or an audio

file. The purpose of this technique is to protect the sensitive information by hiding the information in another media and to avoid unauthorized user from accessing the information.

In this paper AES algorithm is proposed to encrypt the secret text before embedding the data into the cover file. LSB algorithm is proposed to illustrate the security potential of the steganography. An application of the algorithm is illustrated with Waveform Audio File Format (WAV) file as a cover medium.

The rest of the paper is arranged as follows, section 2 does literature survey of the audio steganographic techniques and AES encryption. In section 3 the proposed audio steganographic technique has been described. While, the flowchart shows the overall user steps in using the application. Section 4 gives results and performance evaluation of AES encryption, audio cover similarity and secret data integrity testing. Conclusion is presented in Section 5.

2. LITERATURE REVIEW

2.1 Audio Steganography

“Steganos” means covered or secret and “graphy” means writing or drawing derived from the Greek word, hence bringing the meaning of the word steganography as a secret writing[12]. Steganography is an art of hiding the presence information inside other media. The purpose of this technique is to protect the sensitive information by hiding the information in another media as to avoid unauthorized user from accessing the information.

Audio steganography is a technique that uses audio type as a media to cover the hidden message. In this technique [15], the user will produce stego-file by embedding the secret data inside the digital cover of audio file using a key so that the attacker does not acknowledge the presence of the secret data. Only the one with the key and knowledge that a specific file in which the secret data is being embedded in, can thus extract the secret file.

In order to embed secret data inside the audio cover a lot of techniques have been introduced. The technique used ranges from a simple algorithm embedding the information inside signal noise to the advanced techniques that exploits the sophisticated signal processing technique. In [13], a new technique to improve the capacity of the spread spectrum technique has been discovered. Roy et.al [14] modified the LSB technique to increase the capacity and bit error optimization in audio steganography.

Audio steganography method use the properties of the human auditory system (HAS), according to [6] HAS is more sensitive when compared to the human visual system (HVS). It is more challenging to embed the secret data inside a digital audio file than using a digital image file as a cover media, but there are some loopholes in HAS that can be exploited.

2.1 LSB Encoding Technique

Low bit encoding also known as least significant bit (LSB), is the earliest method used to cover the existence of the information [7]. This technique works by using the least significant bit of cover file to hide the secret information.

The most right bit of the sample audio is the least significant bit. Given an example of the text 'HEY' as a secret message, the LSB technique will embed the binary value of 'HEY' inside the least significant bit in the audio file.

This technique is easy to implement and has a high embedded capacity but this technique has a low robustness level. If the cover file was modified, the secret data inside it will be destroyed [7]. Besides that the hidden data could be easily extracted by the attacker, since the data was only embedded on the least significant bit.

2.1.1 Human Auditory System

According to [8] the human auditory system is a sensory system in human to detect noise. All parts in the ear is a part of HAS. Human auditory ranges are between 15 Hz to 20000 HZ and as humans age the range will get smaller. The ability for human to distinguish the two (2) different sounds is about 3 Hz to 4Hz for the frequency between 15 Hz to 2000Hz and it means in between those ranges human can only differentiate 600 frequency. While for above 2000Hz it needs 0.3% of the frequency sound and for the range between 2000 Hz to 16000 Hz humans can only differentiate 720 frequencies.

According to [9] "HAS have a large dynamic range, that has a fairly small differential range", means a loud sound tends to cover up the quiet sounds. There are also some distortions that the HAS ignores. Moreover, audio signal has the characteristics of redundancy and unpredictable nature that makes it perfect to cover the secret message inside it.

2.2 Advanced Encryption Standard

The Advanced Encryption Standard (AES) is a technique to encrypt the information. It was developed in 2001 by Vincent Rijman and Joan Daeman. According to [10] AES algorithm was used by the United States government to protect sensitive information. In AES it uses the same key that is known as a symmetric key to encrypt and decrypt the information, . Each block depends on the key size and has a different round to perform encryption and decryption. While for the 128 bit it will be 10 rounds the 192 bit will be 12 rounds, and 14 rounds for 256 bit. This means that the longer key is used, the more secure the data will be. However, the process to encrypt and decrypt consume more time because of the additional round in the process.

An example of AES process is on the 128 bit key size. These keys are applied along with mathematical operation to make it different from another. The first step requires the process of adding a round key. After that, in each round the process follows the same steps except for the last one. The processes that happen in round 1 to round 9 is substituting byte, shifting row, mixing column and adding a round key. The final 3 processes include substituting byte, shifting rows and adding a round key. To decrypt the same functions are applied except that they are done in a reverse mode.

Robustness is a capability of a secret data to withstand the attack intentionally or unintentionally. The LSB technique has the highest capacity to be compared with other technique but has a low robustness level [16,17]. This makes the research fraternity interested

in combining AES algorithm and LSB technique. Both techniques implementation will be discussed in the next section.

3. METHODOLOGY

This section will discuss about the AES encryption and decryption process involved in the secret information of the audio file. It also discusses the LSB technique to be applied in the process of hiding the secret text.

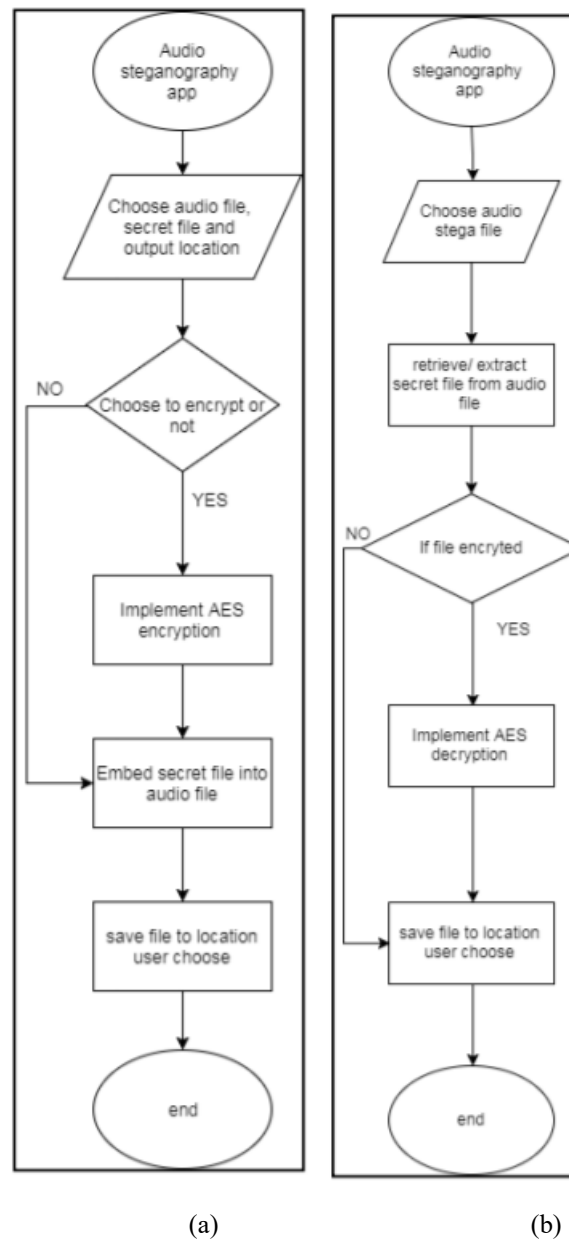


Figure 1: Flowchart of Audio Steganography Application for (a) hiding and (b) retrieval features.

3.1 Flowchart of the proposed project

The encryption process will convert the plaintext to ciphertext, while steganography will hide the secret data. Figure 1 shows the user steps in using the proposed application. For hiding the text, the user will need to choose audio file as a cover file and secret file to embed inside it. User The location of the file and file name also needs to be specified by the user. In addition, the user can choose to implement encryption or not to strengthen the security for audio steganography cover.

Meanwhile in the Figure 1 (b) the flowchart to retrieve data option is shown. The user needs to choose the cover file to retrieve data from it. If the cover file is implemented with encryption, the user needs to key-in the password in order to retrieve the file. Beforehand the password is sent to the intended recipient via the email. So, only the authenticated user will be able to retrieve the secret file.

3.2 Implementation of the LSB technique and AES Encryption

The LSB technique uses the secret file and the cover file for embedding purpose. The application supports the 8 bit, 16 bit, 32 bit, and 64 bit .wav audio file. This algorithm will first read all the bit inside the audio file and then calculate the available size to embed the secret file. It also includes the identification code to differentiate between the cover file bit with the secret file bit. If the user prompts for an encryption, AES algorithm will encrypt the data and help in padding the password. It also implements a random initialization vector key and salt key to enhance the security of the secret data.

In order to retrieve the data, the user needs to choose the cover file and key-in the password only if the cover file is implemented with the encryption. The password is used to decrypt the secret file cipher text to readable text. The algorithm works by reading the entire bit stream to find the identification code that is embedded on the beginning and ending of the secret file. The identification code is used to indicate if there are any secret files and to keep the file integrity check.

As testing purposes, the “Similarity” application was used to identify the similarity between those files. The “Similarity” application identifies the audio file similarities by using an audio fingerprint algorithm.

On the other hand, Secret file Integrity testing was used to measure the integrity of the secret file. In this test, “HashmyFile” application will have to be used. This application is used to generate hashing value between original secret file and extracted secret file, by comparing the hashing value from both of the files which thus conclude the file integrity. This testing method also verifies the encryption on the secret file by analysing the hashing value on the stega-file.

4. RESULT AND ANALYSIS

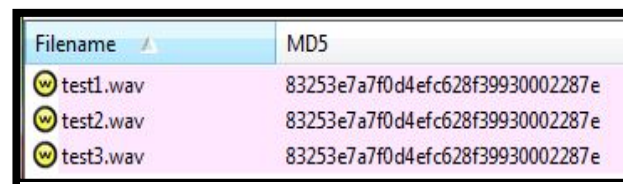
The result and analysis for the proposed project is given in this section. AES encryption result and decryption result is given in section 4.1. The audio similarity result is given in the section 4.2, while the file integrity result is given in section 4.3.

4.1 AES Encryption and Decryption Result

Secure data using audio steganography was tested using same audio file and secret file for control variable. While the carrier file format used is in the form of a .wav file format the secret file is in a .txt file format. The testing involves embedding without AES encryption and embedding with AES encryption.

4.1.1 Embedding without AES Encryption

Embedding without AES encryption was done without password. It was tested for three times using the same carrier file and secret file. The result was analysed using “HashMyFile” tool to calculate the hashing to measure the data similarity.



Filename	MD5
test1.wav	83253e7a7f0d4efc628f39930002287e
test2.wav	83253e7a7f0d4efc628f39930002287e
test3.wav	83253e7a7f0d4efc628f39930002287e

Figure 2: Hashing Result for Embedding without Encryption

As shown in Figure 2 the hashing using MD5 algorithm, the results show that all the stego-file has the same hashing. From the result we can conclude that the attacker is able to detect a stego-file only if the attacker has the original carrier file and deep knowledge in steganography. Such conditions can thus ease the attacker/hacker to extract the secret information from the stego file because the original carrier and stego-file can be analysed to compare the differences and therefore results in the information being extracted.

4.1.2 Embedding with AES Encryption

Embedding with AES encryption was done using password and encryption. It was tested for three times using the same carrier and secret file. The result was analysed using “HashMyFile” tool to calculate the hashing to measure the data similarity.

Filename	MD5
test1.wav	a7ef354168d9885c1c97385664c56a5d
test2.wav	de85876b497bc66a549d53d90f4f0fc5
test3.wav	ac9d27f60fd2ba136f5cc433db9465d3

Figure 3: Hashing Result for Embedding with Encryption

As shown in the Figure 3, even when the sample and secret file were the same, the hashing algorithm was different. This is because the encryption was used in padding the initialization vector on both password and ciphered text to create a randomness in making the data a secret. Without knowing the value of initialization vector and secret file size the attacker could not decrypt the data.

4.2 Audio Similarity Analysis

The results were collected and then analyzed. From the result of the audio steganography application, it shows that the application works well in developing audio stega-file. Though there are some limitations on the function proposed, it serves its purpose well and is able to do what it is intended for.

As shown in Figure 4, the test was done using the audio similarity tool. The original audio file name is a Sample.wav and the audio file with “test” file name is a stega-file. From the result of the original audio file and the audio stegafile, it was shown that the file content was 100 % same with the original as well with precise comparison. Similarity tool compares the audio file using an acoustic fingerprint method to detect the similar audio file. The result showed that the original audio file has the same sound as a stega-file.

File	% content	% tags	% precise
E:\backup\D...\test3.wav	100.0%	66.7%	100.0%
E:\backup\D...\test2.wav	100.0%	66.7%	100.0%
E:\backup\D...\test1.wav	100.0%	66.7%	100.0%
E:\...Sample.wav	100.0%	11.1%	100.0%
E:\...\Encrypted test3.wav	100.0%	86.7%	100.0%
E:\...\Encrypted test2.wav	100.0%	86.7%	100.0%
E:\...\Encrypted test1.wav	100.0%	86.7%	Premium only

Figure 4: Audio Similarity Result

4.3 File Integrity Analysis

The stega-file and the secret file were collected and imported to the “HashmyFile” application to be processed and to generate the hashing value respectively.

Filename	MD5
Encrypted test1.wav	a7ef354168d9885c1c97385664c56a5d
Encrypted test2.wav	de85876b497bc66a549d53d90f4f0fc5
Encrypted test3.wav	ac9d27f60fd2ba136f5cc433db9465d3
Sample.wav	283cd4e6b0906e8ac71306e35db569fd
test1.wav	83253e7a7f0d4efc628f39930002287e
test2.wav	83253e7a7f0d4efc628f39930002287e
test3.wav	83253e7a7f0d4efc628f39930002287e

Figure 5: Hashing Value of the original audio file and audio stega-file

Figure 5 shows the hashing value for the original audio file and audio stega-file respectively. In this test, the secret file and audio file are used as a control variable. The result for test1.wav, test2.wav and test3.wav shows the same hashing value even after embedding the secret file. However, the stega- file that uses the AES encrypted shows a different hashing value. This result proves that the AES encryption works well in this application.

Filename	MD5
TestOriginal.txt	8f838f5c64d2bea0140cb8761a27b3c1
Encrypted test1.wav.e...	8f838f5c64d2bea0140cb8761a27b3c1
Encrypted test2.wav.e...	8f838f5c64d2bea0140cb8761a27b3c1
Encrypted test3.wav.e...	8f838f5c64d2bea0140cb8761a27b3c1
test1.wav.extract.txt	8f838f5c64d2bea0140cb8761a27b3c1
test2.wav.extract.txt	8f838f5c64d2bea0140cb8761a27b3c1
test3.wav.extract.txt	8f838f5c64d2bea0140cb8761a27b3c1

Figure 6: Hashing Value of the retrieved secret file

As shown in Figure 6, the result of hashing value for all retrieved secret file including the original secret file has the same hashing value. This test result shows that the file was not being altered and the file integrity was assured.

5. CONCLUSION

From the result and analysis it can be concluded that the embedding with AES encryption is more secure because the secret information could be extracted without knowing the original secret file. Additionally, the data was combined with multiple information to make it discrete and unreadable.

The low robustness weakness in LSB embedding technique is also the strength in this project. This is because when even one bit data are being altered in the stego-file, all information will be destroyed. With this reason, the secret information is saved from being leaked.

Even though the unencrypted data are able to be extracted it is still a challenging task to extract the information. This is because every steganography has a different signature and the attacker needs to retrieve and reconstruct the data as original to make it able to be read. If the data is not constructed properly the data will be corrupted and unable to be read.

The author said there are no universal steganalysis tool to detect and extract the information from stego-file, due to the fact that the steganography tool uses a different algorithm, padding size and signed ID [11].

The future direction of this project is to be able to increase the hidden capacity and create a higher robustness technique to hide the file. Moreover, it hopes to implement a higher security level to secure any secret information.

ACKNOWLEDGEMENT

The authors would like to thank Universiti Teknologi MARA (UiTM) Melaka and Universiti Teknologi MARA (UiTM) Perlis, Malaysia.

REFERENCES

- [1] Ahmad, S. S. (n.d.). Steganography for Inserting Message on Digital Image Using Least Significant Bit and AES Cryptographic Algorithm.
- [2] Meghanathan, N., & Nayak, L. (2010). Steganalysis Algorithms for Detecting the Hidden Information in Image, a Video and Network Security, 2(1), 43–55.
- [3] Jain, R., & Boaddh, J. (2016). Advances in Digital Image Steganography, (Iciccs), 163–171.
- [4] Putra, M. S. A., Budiman, G., & Novamizanti, L. (2014). Implementation of steganography using LSB with encrypted and compressed text using TEA-LZW on Android. Proceeding - 2014 International Conference on Computer, Control, Informatics and Its Applications: “New Challenges and Opportunities in Big Data”, IC3INA 2014, (1), 93–98. <http://doi.org/10.1109/IC3INA.2014.7042607>
- [5] Stobert, E., & Biddle, R. (2014). A Password Manager that Doesn't Remember Passwords. Proceedings of the 2014 Workshop on New Security Paradigms Workshop, 39–52. <http://doi.org/10.1145/2683467.2683471>
- [6] Tanwar, R., & Bisla, M. (2014). Audio steganography. ICROIT 2014 - Proceedings of the 2014 International Conference on Reliability, Optimization and Information Technology, 322–325. <http://doi.org/10.1109/ICROIT.2014.6798347>
- [7] Djebbar, F., Ayad, B., Meraim, K. A., & Hamam, H. (2012). Comparative study of digital audio steganography techniques. EURASIP Journal on Audio, Speech, and Music Processing, 2012(1), 1–16. <http://doi.org/10.1186/1687-4722-2012-25>
- [8] Zwicker, E., & Fastl, H. (2013). Psychoacoustics: Facts and models (Vol. 22). Springer Science & Business Media.
- [9] Gupta, N., & Sharma, N. (2014). Dwt and LSB based Audio Steganography. ICROIT 2014 - Proceedings of the 2014 International Conference on Reliability, Optimization and Information Technology, 428–431. <http://doi.org/10.1109/ICROIT.2014.6798368>
- [10] Bhanot, R., & Hans, R. (2015). A review and comparative analysis of various encryption algorithms. International Journal of Security and Its Applications, 9(4), 289–306. <http://doi.org/10.14257/ijisia.2015.9.4.27>
- [11] Raggio, M., & Hosmer, C. (2012). Data Hiding: Exposing Concealed Data in Multimedia, Operating Systems, Mobile Devices and Network Protocols. Data Hiding: Exposing Concealed Data in Multimedia, Operating Systems, Mobile Devices and Network Protocols. <http://doi.org/10.1016/C2011-0-05181-5>
- [12] Mohan, C. (2015). Steganography based information security with high embedding capacity, 17–21.
- [13] Zhang, P., Li, Y., Ma, X., Fan, Y., & Chen, X. (2015). Efficient Audio Data Hiding via Parallel Combinatory Spread Spectrum, (2015), 814-818. <https://doi.org/10.1109/CISP.2015.7407989>

- [14] Roy, S., Parida, J., Singh, A. K., & Sairam, A. S. (2012). Audio steganography using LSB encoding technique with increased capacity and bit error rate optimization. *Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology - CCSEIT '12*, (October), 372–376. <https://doi.org/10.1145/2393216.2393279>
- [15] Kamred, U.,S., (2014). A Survey on Audio Steganography Approaches. *International Journal of Computer Applications*. 95(14)