Hiding Encrypted Text Files In Multimedia Files

Milu Mary Rajan
Dept of CSE
GEC, Thrissur

Ajay James
Assistant Professor
Dept of CSE
GEC,Thrissur

Abstract

The security and confidentiality of secret data has become of prime and supreme importance and concern because of the explosive growth of internet and fast telecommunication techniques in recent years. Different methods like cryptography, hashing, steganography, authentication have been developed and in practice today for protecting data from unauthorised access. Our main goal in this paper is to give new insights and directions on how to improve existing methods of hiding secret messages, possibly by combining steganography and cryptography. We start by describing three powerful cryptographic algorithms namely AES, Twofish, Serpent and one commonly used steganography method, ' LSB Substitution'. This proposed system encrypts the secret data with these crypto algorithms and then embeds the encrypted data in a cover file. The cover file can be either an image file(data format used is .bmp) or an audio file(data format used is .wav).

1. Introduction

Cryptography and steganography are well known and widely used techniques that manipulate secret data in order to cipher or hide their existence respectively. Steganography is the art and science of hiding the fact thatcommunication is taking place whereas cryptography is the practical art of converting messages into a different form. In cryptography, the sender transmits the secret message through an insecure channel after scrambling it using an encryption key. The reconstruction of the original secret message is possible if the receiver has the appropriate decryption key. Steganography on the other hand, will embed the secret message in another message, thus the existence of message is unknown. In this paper we are focusing to develop one system, which uses both cryptography & steganography for better confidentiality and security.

Presently we have very secure methods for both cryptography and steganography [4]. We have used three such crypto algorithms namely AES(Advanced Encryption Standard), Twofish, serpent and most common LSB(Least Significant Bit) sustitution method for steganography. We know that hiding data is better than moving it shown and encrypted. Even the intruder extracts the data it will be in encrypted form. But still there is a chance that the intruder can break the code. In our system we will be using the following approach for ensuring more security

- We encrypt the data using three cryptographic algorithms instead of using a single algorithm [3].
- Embed the secret message in different types of cover objects.
- Use three keys for both encryption as well as for decryption.

So our final goal is to develop a new system that cascades AES ,Twofish, Serpent algorithms for encryption and hides the encrypted data using LSB sustitution [2].

2. Cryptography Module

Cryptography is an important element of any strategy to address message transmission security requirements. Cryptography is the study of methods of sending messages in disguised form so that only the intended recipients can remove the disguise and read the message. It is the practical art of converting messages or data into a different form, such that no-one can read them without having access to the 'key'. The message may be converted using a 'code' or a 'cypher' or 'cipher'.

The following sections discuss about three powerful cryptography algorithms that are used in the proposing system.

2.1. AES Algorithm

AES is a symmetric block cipher. The block and key can in fact be chosen independently from 128, 160, 192, 224, 256 bits and need not be the same. However, the AES standard states that the algorithm can only accept a block size of 128 bits and a choice of three keys - 128, 192, 256 bits. If the key size used is 128 then the number of rounds is 10. Here we use 128 bit data blocks and 128 bit key.

AES algorithm[5] begins with an Add round key stage followed by 9 rounds of four stages and a tenth round of three stages. This applies for both encryption and decryption with the exception that each stage of a round the decryption algorithm is the inverse of its counterpart in the encryption algorithm.

The four stages are as follows:

- Substitute Bytes
- Shift Rows
- Mix columns
- Add Round Key

The tenth round simply leaves out the Mix Columns stage. The first nine rounds of the decryption algorithm consist of the following:

- Inverse Substitute Bytes
- Inverse Shift Rows
- Inverse Mix columns
- Inverse Add Round Key

Again, the tenth round simply leaves out the Inverse Mix Columns stage.

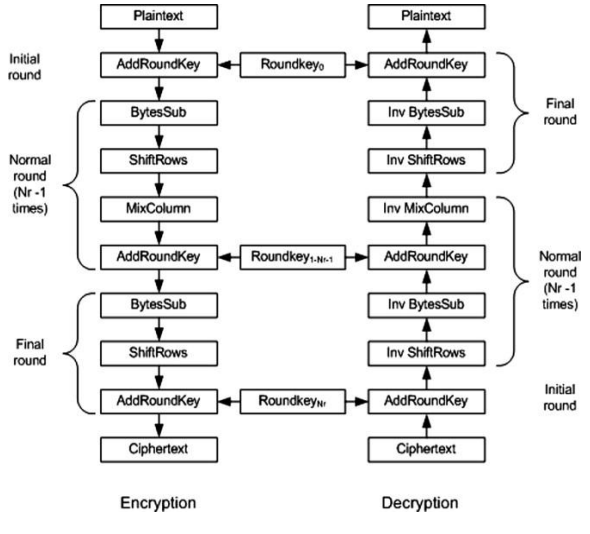

Figure 1: Overall sturcture of AES

Substitute Bytes stage is simply a table lookup using a 16 x 16 matrix of byte values called an s-box. This matrix consists of all the possible combinations of an 8 bit sequence (28 = 16 x 16 = 256). Shift rows transformation use simple permutation on each row. In Mix column transformation each byte of a column is mapped into a new value that is a function of all four bytes in the column. The 128 bits of state are bitwise XOR ed with the 128 bits of the round key in Add Round Key stage.

2.2. Twofish Algorithm

Figure 2 shows an overview of the Twofish block cipher. A closer look at the Twofish cipher reveals these design elements:-

- The key dependent S-boxes
- The Maximum Distance Separable (MDS) matrix
- Pseudo Hadamard Transform (PHT)
- Fiestel networks

Twofish[6] uses a 16-round Feistel like structure with additional whitening of the input and output. The only non-Feistel elements are the 1-bit rotates. The rotations can be moved into the F function to create a pure Feistel structure, but this requires an additional rotation of the words just before the output whitening step. The plaintext is split intofour 32-bit words. In the input whitening step, these are XOR ed with four key words. This is followed by sixteen rounds.

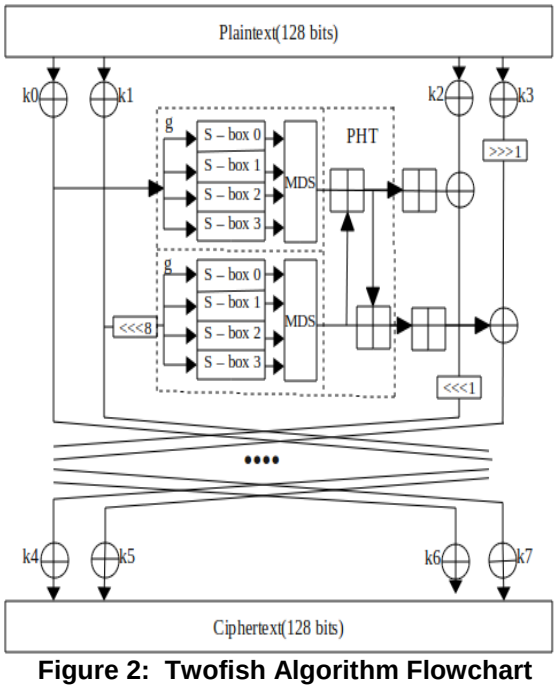

Figure 2: Twofish Algorithm Flowchart

- The two words on the left are used as input to the g functions. (One of them is rotated by 8 bits first.)

- The g function consists of four byte-wide key-dependent S-boxes, followed by a linear mixing step based on an MDS matrix.

- The results of the two g functions are combined using a Pseudo Hadamard Transform (PHT), and two keywords are added.)

- These two results are then XOR ed into the words on the right (one of which is rotated left by 1 bit first, the other is rotated right afterwards).

- The left and right halves are then swapped for the next round.

- After all the rounds, the swap of the last round is reversed, and the four words are XOR ed with four more key words to produce the cipher text.

2.3. Serpent Algorithm

The Serpent algorithm is a 32-round Substitution-Permutation (SP) network operating on four 32-bit words. The Serpent algorithm[7] consists of three main components

- Initial Permutation IP

- Thirty-two rounds consisting of a Round Function that performs Key Masking, S-Box Substitution, and (in all but the last round) data mixing via a Linear Transformation

- Final Permutation FP

Major steps are:

- Bit-wise XOR with the 128-bit Round Key K

- Substitution via thirty-two copies of one of eight S-Boxes

- Data mixing via a Linear Transformation

These operations are performed in each of the thirty-two rounds with the exception of the last round. In the last round, the Linear Transformation is replaced with a bit-wise XOR with a final 128-bit key. One of a total of eight different S-Boxes is used per round, where each S-Box performs a 4-bit to 4-bit substitution operation. The S-Box used is the round number modulo eight: round 9 uses S-Box 1, round 18 uses S-Box 2, etc. Each round requires thirty- two copies of the appropriate S-Box to operate on the 128-bit input data. The thirty-two 4-bit S-Box outputs form the 128-bit data that is input to the Linear Transformation.

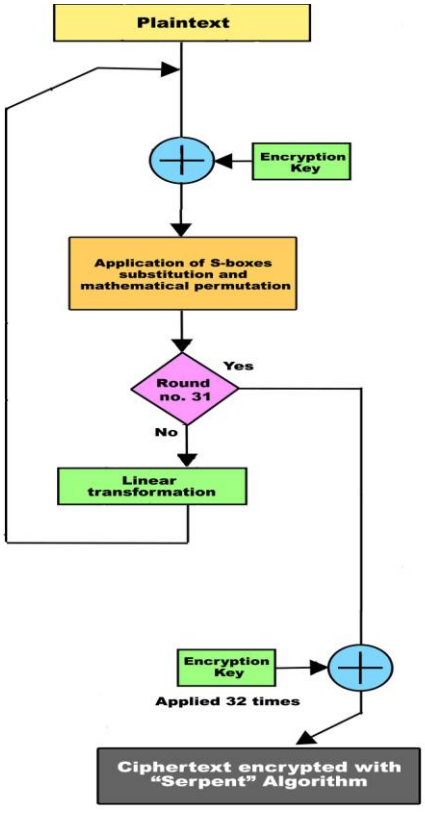

Figure 3: Serpent Algorithm Flowchart

The Linear Transformation mixes the four 32-bit blocks of data, denoted by X0, X1,

X2, and X3 based on the equations below. Note that <<< denotes a left rotation and << denotes a left shift .

input:= X0 , X1 , X2 , X3

X0 := X0 <<<13

X2 := X2 <<<3

X1 := X1 ⊕X0 ⊕X2

X3 := X3 ⊕X2 ⊕(X0<<3)

X1 := X1<<<1

X3 := X3<<<7

X0 := X0 ⊕X1 ⊕X3

X2 := X2 ⊕X3 ⊕(X1<<7)

X0 := X0<<<5

X2 := X2<<<22

output := X0, X1, X2, X3

3. Steganography Module

Data hiding is a method of hiding secret messages in to a cover-media such that an unintended observer will not be aware of the existence of the hidden messages. Here jpg image files and wav audio files are selected as the cover- media. These images and audios are referred to as cover-files. Cover-files with the secret messages embedded in them are called stego-files. For data hiding methods, the image/audio quality refers to the quality of the stego-file.

There are many techniques for hiding data in image as well as audio. One of the common techniques is based on manipulating the Least-Signicant-Bit (LSB) planes by directly replacing the LSBs of the cover-image with the message bits. LSB methods typically achieve high capacity. Least significant bit (LSB) coding is the simplest way to embed information in image as well as in digital audio file.

3.1. Hiding data in images by simple LSB substitution

LSB substitution is the process of using least significant bit pixels of the carrier image for data hiding purpose. It is the most simplest method for embedding secret data in to the image. The least significant bit insertion varies according to number of bits in an image. For an 8 bit image, the least significant bit ie, the 8^{th} bit of each byte of the image is changed to the bit of secret message. For 24 bit image, the colors of each component like RGB(red,green and blue) are changed. LSB is effective in using bmp images since the compression in bmp is lossless. But for hiding the secret message inside an image of bmp file using LSB algorithm it requires a large image which is used as a cover.

3.2. Hiding data in audio by simple LSB substitution

Among many different data hiding techniques

proposed to embed secret message within audio file, the LSB data hiding technique is one of the simplest methods for inserting data into digital signals in noise free environments, which merely embeds secret message-bits in a subset of the LSB planes of the audio stream.

The following steps are performed : -

• Receives the audio file in the form of bytes and converted in to bit pattern.

• Each character in the message is converted in bit pattern.

• Replaces the LSB bit from audio with LSB bit from character in the message.

Low-bit encoding embeds secret data into the least significant bit (LSB) of the audio file.

4. Proposed Method

This paper aims at enhancing the performance of existing combination of algorithms. Here the secret message is encrypted using three efficient cryptographic algorithms namely AES, Twofish, Serpent algorithms. The system receives three keys from the sender for this triple encryption process. These keys should be known to the intended receiver, otherwise he cannot decrypt the message. Then hide the encrypted data in either image file or audio file. This doubly protected data is then sent to the intended receiver.

The hidden encrypted data is extracted out from the stego file at the receiver end. The encrypted data is then converted to original secret message by applying decryption algorithms in reverse order. The receiver should use the encryption keys itself for proper decryption.

The system design can be divided in to three modules:

• Embedding program design

• Extracting program design

The detailed design of each module is given below.

4.1. Embedding program design

This following section encrypts data using AES, Twofish, Serpent algorithms in the mentioned order and hides it in the selected file.

Step0: start

step1: Secret message and cover file for hiding is available at the sender.

Step2: backup = secret message

step3: convert first password in to 128 bit key

step4: encrypted data = perform AES encryption on backup

step5: if AES is successful then backup= encrypted data else go to step 3

step6: convert second password in to 128 bit key

step7: encrypted data= perform Twofish encryption on backup

step8: if Twofish is successful then backup= encrypted data else go to step 6

step9: convert third password in to 128 bit key

step10: encrypted data = perform Serpent encryption on backup

step11: if Serpent is successful then backup = encrypted data. Else print error message and go to step 9

step12: Hide backup in the cover file

step13: if Hiding is successful then go to step 14 else go to step 12

step14: Stop

4.2. Extracting program design

This following section unhides the data and decrypt it using Serpent, Twofish, AES algorithms in the mentioned order.

step0: start

step1.a: stego file is entered. backup = stego file

step1.b: decrypt data = unhide the encrypted data from backup

step2: if ! successful go to step 1.b

step3: convert first password to 128 bit key

step4: decrypt data = perform Serpent decryption on decrypt data

step5: if Serpent decryption failed go to step 13

step6: convert second password to 128 bit key

step7: decrypt data = perform Twofish decryption on decrypt data

step8: if Twofish decryption failed go to step 13

step9: convert third password to 128 bit key

step10: decrypt data = perform AES decryption on decrypt data

step11: if AES decryption failed go to step 13

step12: print secret message go to step 14

step13: print error message

step14: stop

5. Conclusion

This paper focuses on providing secure data transmission based on the concept of cryptography and steganography, by the cascaded encryption of secret data and hiding this encrypted data in a cover file. Considering the issues of unauthorized access and other security issues on data transfer, we propose the idea of "Hiding encrypted data in multimedia files". The concept of cascading AES, Twofish, Serpent algorithms in the encryption phase and hiding this encrypted data in multimedia files using simple LSB substitution method, helps to reduce the vulnerability of ciphers and hence the secret data will get secured.

10. References

[1] ADhawal Seth, L Ramanathan and Abhishek pandey, Security Enhancement: "Combining Cryptography and Steganography", *International Journal of computer Applications* .

[2] Chi-Kwong Chan and L M Cheng, ""Hiding data in images by simple LSB Substitution", *The Journal of the Pattern Recognition Society.*

[3] Aseel M Al-Anani, M H Abdallah, Randa A Al-Dallah and Rola I Al-Khalid, "Multimedia Multilevel Hiding Technique", *European Journal of Scientific Research.*

[4] A Joseph Raphael and Dr. V sundaram, "Cryptography and Steganography- A Survey", *International Journal for Computer Applications.*

[5] *National Institute of Standards and Technology (NIST):* ":Advanced Encryption Standard (AES)", FIPS-197 (2001)

[6] Purnima Gehlot, Richa Sharma, S. R. Biradar : "VHDL Implementation of Twofish Algorithm "

[7] Ross Anderson1, Eli Biham2, Lars Knudsen3: "Serpent: A Proposal for the Advanced Encryption Standard"