Image Assisted Data Security using Key Encrypted File

Sarath S Pillai
Software Engg. & Researcher
EXOTEL , Bangalore , INDIA

Syam S Pillai
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Gopikrishnan U
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Sruthi George
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Rintu Joseph
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Aswathy G S
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Manjusha Nair S
Asst. Professo,
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Reby John
Asst. Professor,
Department Of Computer Science
College Of Engineering Chengannur
Kerala, INDIA

Abstract*— In this paper a new technique for image based data hiding with a binary file is proposed. In traditional methods the image based data hiding referred as steganography. The basic idea of Steganography is that it distributes the message uniformly throughout the image. But it will cause degradation to the image. In this paper the image based data hiding is performed using the binary file that ensures more security and confidentiality. Here the image is divided into blocks of equal sizes and the message is then embedded into the central pixel of the block using cyclic combination of 6th, 7th, 8th bit. The block of the image is chosen randomly using Array shuffling algorithm similar to pseudo random number generator seeded with an array shuffling key. In proposed method the image is divided into 256 blocks and the message is uniformly distributed in the image but there will be no change in the image pixels. The user keeps a secret binary file along with the image that contains the bits corresponding to the message. The binary file that contains bit values corresponding to the message will provides more security. The binary file is then encrypted with the same key or another key and converted to base64 encoding. And it is send to the receiver. The receiver can download the image from anywhere and the information about the image is shared by sending the URL or it may be known to the receiver. The array shuffling key is shared with the receiver by using any of the symmetric key exchange technique. Then receiver can retrieves the message by analyzing the binary file by decrypting and converting to normal utf-8 encoding from base64. In this method the image quality is maintained as such as the original image, ie the pixel value of the image does not changed. The main advantage is that it does not limit the size of message that can be distributed in the image. The message can be retrieved only with the help of image, binary file, and array shuffling key. So the security is assured.

***Keywords*— Image, Block shuffling, double encryption**

I. INTRODUCTION

In recent years, everyone is moving towards digital world. With the rapid development of the internet technologies, digital media needs to be transmitted conveniently over the network. Attacks, unauthorised access of the information over the network become greater issue.

In this paper, image assisted data retrieval using binary file is proposed. The binary file contains the bits referring the changes to the pixel value of the image with respect to the message. This binary file is send to the receiver along with the array shuffling key which will be discussed later. The receiver is made known about the image. Thus he can retrieve the message. Here security of message is guaranteed as the receiver can retrieve message only if he knows the binary file, image and the shuffling key.

Usually Cryptography and Steganography are the solutions to hide data. Steganography is an art and science of hiding the data in some covered writing. Steganography is different from Cryptography which is about concealing the content of message whereas Steganography is about concealing the existence of message itself [1].

Steganography techniques uses different media like image files, audio files, video files and text files for secret communication. Depending upon the cover media we can classify the Steganography into many parts;

1. Text Steganography
2. Image Steganography
3. Audio Steganography
4. Video Steganography

There are many parameters that affect Steganography techniques. These parameters include hiding capacity, perceptual transparency robustness, complexity, survivability, capability and deductibility.

1. *Hiding Capacity*

Hiding capacity is the size of information that can be hidden relative to the size of the cover. A large hiding capacity allows the use of a smaller cover for a message of font size, and thus decreases the bandwidth required to transmit the stego-image.

2. *Perceptual Transparency*

The act of hiding the message in the cover necessitates some noise modulation or distortion of the cover image. It is important that the embedding occur without significant degradation or loss of perceptual quality of the cover. In a secret communication application, if an attacker notices some distortion that arouses suspicion of the presence of hidden data in stego-image, the Steganography encoding has failed even if the attacker is unable to extract the message.

3. *Robustness*

Robustness refers to the ability of embedded data to remain intact if the stego-image undergoes transformations such as linear and non-linear filtering, addition of random noise, cropping or decimation, lossy compression and conversion back to digital form.

4. *Tamper Resistance*

Beyond robustness to destruction, tamper resistance refers to the difficulty for an attacker to alter or forge a message once it has been embedded in a stego-image, such as a pirate replacing a copyright mark with one claiming legal ownership.

II. RELATED WORKS

Steganography is implemented before centuries .It has been explained before inventing the computer .But after the invention of computer and networking the secure data flood over networks and the need of providing the security become more necessary. Encryption and Cryptography are the common methods to provide security. There are some existing systems of Steganography.

1. *LSB Method [2]*

In this method, least significant bit of pixel value is used for insertion of message. This method is easy to implement but it has many disadvantages associated with it.

- Message can be easily recovered by an unauthorized person as message is in LSB
- As message is hidden in LSB, so intruder can modify the LSB, so intruder can modify the LSB of all the image pixels in the way the hidden message can be destroyed.
- LSB is the most vulnerable to hardware imperfections or quantization of noise.

2. *6th, 7th Bit Method [2]*

In this method Parvinder et al used the 6th, 7th bit for the interaction of message. They didn’t use any LSB. They overcome the disadvantages associated with LSB method. But this method also has its own disadvantages. The main disadvantage associated with it is that this method provides only the 5o percent chances of message insertion at a pixel value.

3. *PVD (Pixel Value Differencing Method)[2]*

The pixel value differencing method proposed by Wu and Tsai can successfully provide both high embedding capacity and outstanding imperceptibility for the stego-image. The pixel value differencing (PVD) method segments the cover image into non overlapping blocks containing two connecting pixels and modifies the pixel differences in each block (pair) for data embedding. A large difference in the original pixel value allows a greater modification.

4. *Cover Region and Parity Bits Method [3]*

In this technique, the image is divided in a minimum of L(m) contiguous and disjoint regions and their use are defined by a pseudo-random number generator (PRNG).

$$P(I)=\sum_{j\in i} LSB(C_j) \bmod_{2.} ----------(1)$$

It is necessary only one LSB flipping of any pixel of the region to change the parity region value.

III. PROPOSED SYSTEM

In the basic Steganography method the message is uniformly distributed throughout the image. But in this paper instead of hiding the message in the image, we simply keep a binary file along with image. The image can be downloaded from internet. Initially the image is divided into blocks of equal size. The size of the block depends on the size of image and the length of message. The file contains the binary data corresponding to the message. The basic difference with the classic Steganography is that there is no limit to the message to be send, because here we keep the stegnographic data into a file, such that the size of the binary file can be vary with respect to the size of message. And it is password protected that provides additional feature security. Since we keep the bit changes in the file, there is no change to the pixel values of the image. After dividing the image into blocks, central pixel of selected block is calculated. The block is selected using Pseudo Random Number Generator (PRNG) which is seeded with a secret key. Now the corresponding bits are inserted into the file based on the cyclic combination of the last three bits. Cyclic Combination of last three bits is used separately for the insertion of 0 & 1 in the following manner (given in the figure 1).

The proposed system consists of three phases;

Phase 1: Insertion Algorithm

Steps:

i) Dividing the image into 256 blocks. Each block size will be calculated by dividing the image size by 256.

ii) Number each block from one to 256.

iii) Shuffle the block numbers according to the numeric key value by using proposed array shuffling algorithm.

iv) Calculate the central pixel of each block of image according to the shuffled array. The central pixel is calculated using the formula given.

$$C(i)=Abs\left[\frac{B(F)\times(2i-1)+1}{2}\right].------(3)$$

v) Calculate the red value of the central pixel by checking the attribute of the central pixel.[eg:using ‘enumerate’ in python code.].Or check the 3rd last bit of LSB.

vi) Compare the message bit with the 1st bit of the central pixel value and 2nd bit of message with 2nd bit red value of central pixel and so on

vii) If the message bit and central pixel bit are different then input a ‘0’ into the secret file we are keeping with us. If the message bit and central pixel bit are same, then put a ‘0’ into the secret file

viii) If the number of message bit is greater than 256, then shuffle the already shuffled array of blocks. i.e., if message bit reaches 256 then the last block is used. So we need to get another array of blocks. For that reshuffling applied.

Phase 2: Array Shuffling Algorithm.

i) Calculate the greatest prime number below the integer key value.

ii) Perform modulo division on the square of length of array by the prime number and result is updated as the next divisor

iii) The result is divided by 3 and the value of divisor updated.

iv) The element at the divisor valued index of the input array is inserted into the output shuffled array

v) Then the corresponding element inserted is popped from the input array

vi) Length of array is then decremented by one and the divisor value doubled

vii) Repeat the steps 2-6 until the divisor value become zero

viii) Repeat the steps from 2-7 until the length of array become zero

Phase 3: Secret Key Encryption.

i) Construct new key where newkey = modular division of i by key index of original key, where I = 0 to length (message)

ii) Find the ASCII values of data bit and newkey

iii) Add the ASCII value and modular division of sum by 256.

iv) Append the value into an array.

v) Repeat the steps i to vi n times, where n = length(message)

vi) Encode the array with python base64 encoding algorithm.

Phase 4: Secret Key Decryption

i) Encode the input with python base64 decryption method.

ii) Construct new key where newkey = key[i% len (key)],where i is from 0 to len(message)

iii) Subtract the ASCII values of the newkey from the ASCII value of encoded data bit.

iv) Add 256 to the result and modular division by 256

v) Append the character value of the above result to the output file

vi) Repeat the steps n times where n = 0 to length(message)

Phase 5: Retrieval Algorithm

i) Dividing the image into 256 blocks. Each block size will be calculated by dividing the image size by 256.

ii) Number each block from one to 256.

iii) Shuffle the block numbers according to the numeric key value by using proposed array shuffling algorithm.

iv) Calculate the central pixel of each block of image according to the shuffled array. The central pixel is calculated using the formula given below.

$$C(i)=Abs\left[\frac{B(F)\times(2i-1)+1}{2}\right].------(3)$$

v) Calculate the red value of the central pixel by checking the attribute of the central pixel.[eg:using ‘enumerate’ in python code.].Or check the 3rd last bit of LSB.

vi) Compare the secretfile bit with the 1st bit of the central pixel value and 2nd bit of secret file with 2nd bit red value of central pixel and so on

vii) If the secret file bit is one then we need to change the corresponding central pixel value.ie., if central pixel bit is 1,then it need to change to 0 in output file. Else if central pixel is 0, in output file insert 1.

viii) If the secret file bit is 0, then no need to change the corresponding central pixel bit. If it’s 1, then insert one into output file. Else insert 0 to output file.

ix) If the number of secretfile bit is greater than 256, then shuffle the already shuffled array of blocks.

Implementation

Initially user can select the image and divide the image into block of equal size shown in the figure. Each block has a central pixel value and a block number.

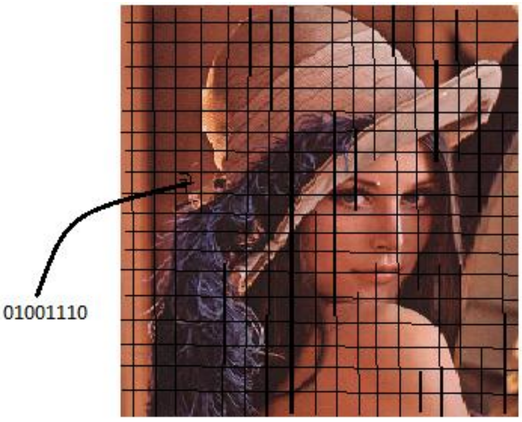

Figure: Image divided into blocks

Obtain a sequence of block numbers using a random number generator. The input is our password. For inserting the message, If there is a change in the last bit of central pixel and the message, then rather than changing the bit value of image, we just insert a bit 1 to the file. If no change of bit is needed then insert 0 to the file. Consider the example;

In the figure the central pixel value is 01001110 and if the message bit is 0, then there is no change between the message and last bit of pixel value, so insert 0 to the binary file. If the message is 1, then insert 1 to the binary file. Then the binary file is encrypted using secret key encryption algorithm.

The binary file is send and at the receiver side the image is downloaded from anywhere and the message extracted with the help of binary file that was decrypted using secret key decryption algorithm. The file is send to the user. The data can retrieve only if the password is provided to him. And the user downloads the image from internet. At the time of retrieval the pixel values of image is needed to change if the corresponding bit in the file is 1, else no change is needed.

IV. RESULT AND DISCUSSIONS

In our proposed system data or information is not directly hidden in the image. Rather we keep an associated binary file to store the corresponding bits 0 or 1 , denoting changes in the pixel value of image. The image can be downloaded directly from internet. And the method preserves the quality of the image and it does not limit the size of message that can be distribute in the image. The decryptor can retrieve information only if both the file and secret key is known to him. Thus security of information is achieved to a great extend compared to existing system.

V. CONCLUSION AND FUTURE SCOPE

The main goal of this paper was to introduce a new method for image Steganography that ensures more secure transmission of data and information. Here there is no need for sending the image, it can be downloaded from anywhere. Only the file and secret key is to be sent. And the pixel value of the actual image is not changed, ie the picture quality do not changes. The change are made in the binary file which is password protected, that provides additional security to the system. The main advantage is that it does not limit the size of information that can be sent.

The main advantages of proposed system are:

- Security of data can be achieved.
- Do not change the pixel value of image.
- Rather than changing the pixel value, corresponding bits 0 or 1 is stored in the file.
- Image can be downloaded anywhere from the Internet.
- Retrieval of data cannot be done without the file and the secret key.
- The size of the data encrypted is independent of the size of image.
- Encryption within encryption. i.e., the secret binary file obtained is also encrypted and converted to base64.

VI. ACKNOWLEDGMENT

We extend our sincere gratitude to our project coordinator Prof. Manjusha S Nair and guide Miss. Reby John, College of Engineering, Chengannur, for motivating suggestions, insightful advice, valuable guidance, help and support in successful completion of this paper.

VII. REFERENCES

[1] A. Gutub& M. Faltani (2007), “A Novel Arabic Text Steganography Method Using LetterPoints and Extension”, WASET International Conference on Computer Information and SystemScience and Engineering (ICCISSE), Vienna, Austria, May 25-27.

[2] RJ Anderson & FAP Petitcolas (1998), “On the Limits Of Steganography”, IEEE Journal onselected Areas in Communications, Vol. 16 No 4, pp 474-481.

[3] R. Chandramouli& N.D. Memon (2003), “Steganography capacity: A steganalysis perspective”,Proc. SPIE Security and Watermarking of Multimedia Contents, Special Session onSteganalysis.

[4] S.K. Pal, P.K. Saxena& S.K. Muttoo (2004), “Image steganography for wireless networks usingthe handmaid transform”, International Conference on Signal Processing & Communications(SPCOM).

[5] M. T. Parvez& A. Gutub (2008), “RGB Intensity Based Variable-Bits Image Steganography”,APSCC 2008-Proceedings of 3rd IEEE Asia-Pacific Services Computing Conference, Yilan,Taiwan, 9-12 December.