IMPLEMENTATION OF INFORMATION SECURITY USING OBJECT ORIENTED CONCEPT

Dr.J.P.Patra[1]
[1]Asso. Professor. B.E., Computer Science and Engineering (CSVTU), India
patra.jyotiprakash@gmail.com[1]

Karishma Rathod[2]
[2]Computer Science and Engineering (CSVTU), India
karishma.rathod@ssipmt.com[2]

Udita Chauhan[3]
[3] Computer Science and Engineering (CSVTU), India
udita.chauhan@ssipmt.com[3]

Abstract— Steganography is the art and science of sending hidden messages such that the presence and nature of such a message is only known by the sender and intended recipient.. In our project, we are introducing an object oriented concept for providing security to the information .This paper provide a brief knowledge about implementation of LSB algorithm using Java methodology. This urges the researcher's to invent new data hiding techniques through Steganography principle to protect and secure the data.

Keywords- Steganography, Data hiding, Encryption, Decryption, Object oriented.

I. INTRODUCTION

Steganography comes from the combination of two Greek words Stegano means closed and Graphy refers to writing means secret writing. Steganography is the art of embedding personal information into other data by using some rules and techniques [3].Steganography is also accomplished through hiding information in other information, thus hiding the existence of the communicated information. As a result, unauthorized users are not able to understand and recognize the embedded information.Many different carrier file formats can be used, but digital images are the most popular. In the domain of digital images, many different image file formats exist, most of them for specific applications. We are using improved LSB(least significant bit) replacement method for producing a secret embedded image that is totally indistinguishable from the original image by the human eye[1].

II. METHODOLOGY

Java technology is both a programming language and a platform. The Java programming language is a high-level language that can be characterized by the buzzwords like simple, object oriented, distributed, robust, secure, portable, dynamic etc. It is an open source, so users do not have to struggle with heavy license fees each year. It is used for project purposes because of the following reasons as it is Platform independent, Java API's can easily be accessed by developers etc. Another advantage of JAVA is that, once the program is written in java we can run it anywhere means that application developed through Java is platform independent. JAVA based enterprise applications perform well because stable JAVA standards help developers to create multilevel applications with a component based approach.

In our project, the programming language used is java and the mechanism used for the encryption and decryption purpose is LSB algorithm.

Data Hiding by LSB: Various techniques about data hiding have been proposed in literatures. One of the common techniques is based on manipulating the least-significant-bit (LSB) [2-5] planes by directly replacing the LSBs of the cover-image with the message bits. LSB methods typically achieve high capacity but unfortunately LSB insertion is vulnerable to slight image manipulation such as cropping and compression.

III. IMPLEMENTATION OF LSB TECHNIQUE

There are many steganography techniques which are capable of hiding data within an image. These techniques can be classified into two categories based on their algorithms: (1) spatial domain based techniques; (2) transform domain based techniques [14]. The spatial domain based steganography technique use either the LSB or Bit Plane Complexity Segmentation (BPCS) algorithm. The most widely used technique to hide data is the usage of the LSB [10]. The existing techniques are mainly based on LSB (Least Significant Bit) where LSBs of the cover file are directly changed with message bits. A significant number of methods have been proposed for LSB steganography [6-9], [11-13]. Masud et al. [2] has proposed a LSB technique for RGB true color image by enhancing the existing LSB substitution techniques to improve the security level of hidden information.

It is a simple approach for embedding message into the image. In this method some information from the pixel of the carrier image is replaced with the message information so that it can't be observed by the human visual system, therefore it exploits some limitations of the human visual system.The Least

Significant Bit insertion varies according to number of bits in an image[15]. For an 8-bit image,the least significant bit i.e. the 8^{th} bit of each byte of the image will be changed by the 1-bit of secret message The least significant bit is the lowest bit in a series of binary number[15]. In LSB substitution the least significant bits of the pixels are displaced by the bits of the secret message which gives rise to an image with a secret message embedded in it. The method of embedding differs according to the number of bits in an image (different in 8 bit and 24 bit images)

Consider an 8-bit grayscale bitmap image where each pixel is stored as a byte representing a grayscale value. Suppose the first eight pixels of the original image have the following grayscale values:

11010010 01001010 10010111 10001100 00010101 01010111 00100110 01000011

To hide the letter C whose binary value is 10000011, we would replace the LSBs of these pixels to have the following new grayscale values:

11010011 01001010 10010110 10001100 00010100 01010110 00100111 01000011.

ALGORITHM:

A. Embedding phase

The embedding process is as follows.

Inputs: Image file and the text file

Output: Text embedded image

Procedure:

Step 1: Extract all the pixels in the given image and store it in the array called Pixel-array.

Step 2: Extract all the characters in the given text file and store it in the array called Character- array.

Step 3: Extract all the characters from the Stego key and store it in the array called Key-array.

Step 4: Choose first pixel and pick characters from Key- array and place it in first component of pixel. If there are more characters in Key- array, then place rest in the first component of next pixels, otherwise follow Step (e).

Step 5: Place some terminating symbol to indicate end of the key. '0' has been used as a terminating symbol in this algorithm.

Step 6: Place characters of Character- Array in each first component (blue channel) of next pixels by replacing it.

Step 7: Repeat step 6 till all the characters has been embedded.

Step 8: Again place some terminating symbol to indicate end of data.

Step 9: Obtained image will hide all the characters that input.

B. Extraction phase

The extraction process is as follows.

Inputs: Embedded image file

Output: Secret text message

Procedure:

Step 1: Consider three arrays. Let they be Character-Array, Key-array and Pixel-array.

Step 2: Extract all the pixels in the given image and store it in the array called Pixel-array.

Step 3: Now, start scanning pixels from first pixel and extract key characters from first (blue) component of the pixels and place it in Key-array. Follow Step3 up to terminating symbol, otherwise follow step 4.

Step 4: If this extracted key matches with the key entered by the receiver, then follow Step 5, otherwise terminate the program by displaying message "Key is not matching".

Step 5: If the key is valid, then again start scanning next pixels and extract secret message characters from first (blue) component of next pixels and place it in Character array. Follow Step 5 till up to terminating symbol, otherwise follow step 6.

Step 6: Extract secret message from Character-array. The primary motivation of the current work is to increase PSNR. For this purpose we employ the approach which hide secret image in to cover image with the help of logic gates.

Algorithm

Step1: Read the image to be embedded

Step 2: Read the image inside which message is embed

Step 3: set numSignificantBits = n ; where n= 1,2.....8

Step 4: size1 = size(secret); and size2 = size(coverImage);

Step 5: set the "numSignificantBits"n significant bits of each byte of cover image to zero by using bit by AND operation on cover and size1 matrix

Step 6: embedd the "numSignificantBits" most significant bits of secret image to create the stego image by using stego= (cover zero+ secret)/28-n

Step 7: recover the embedded image, by using bit by shift operation

Step 8: Display Figure of cover image, Image to be hidden, stego image and recover image

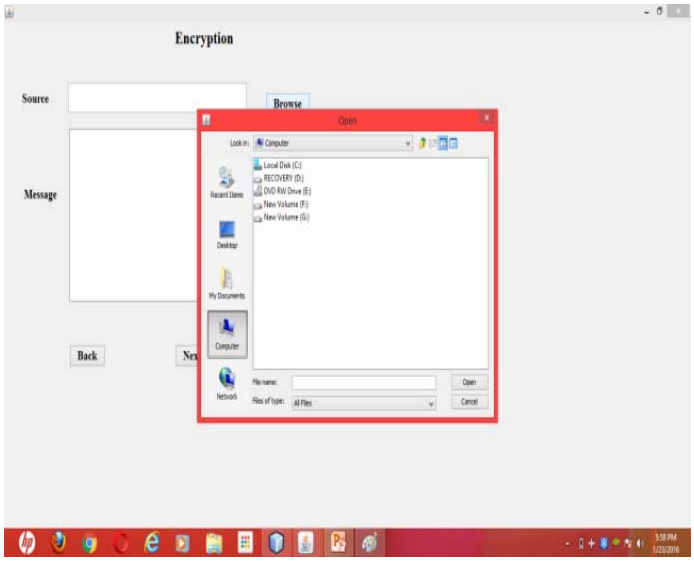
Step 9: End

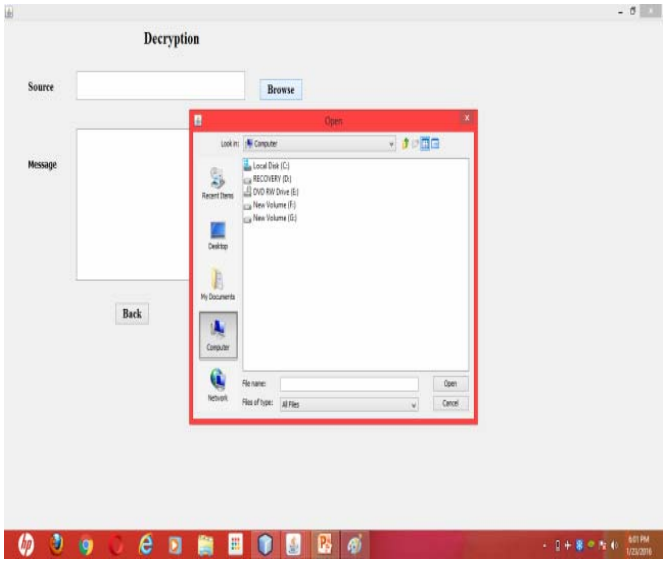
IV. WORKING

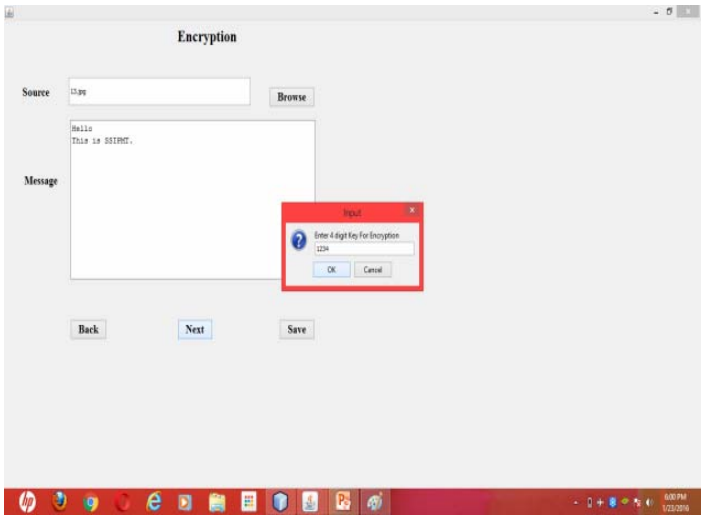
The following snapshots will show how the steganography project will work. The first two figures are used for encryption purpose. In the first figure, the user will provide the necessary information i.e. image behind which the message is to be passed (path of the image), the message to be passed. And in the second figure, it will ask for entering a passkey.

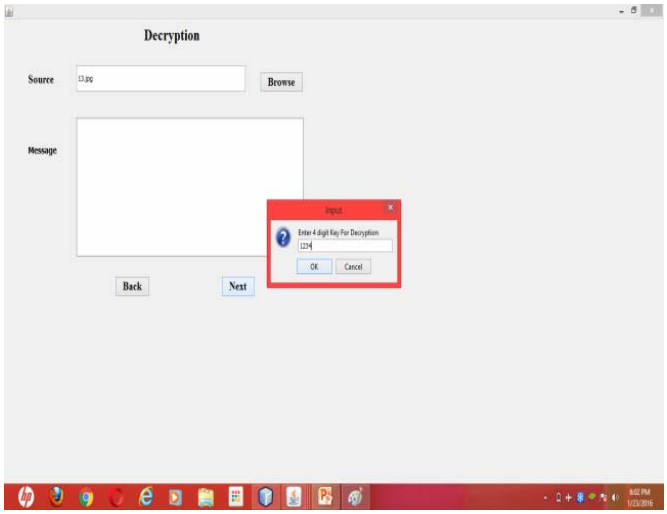
Now for decryption, i.e. third figure, the user is asked to provide the path of the image and after it, when it will proceed, then it will ask for entering the passkey again. If passkey is correct as entered same by the sender, then it will show the message. Otherwise the message is not shown.

The figures are as follows-









The above two figures are performed while the encryption process is carried on. In the first figure, the user will provide the necessary information i.e. image behind which the message is to be passed (path of the image) and the it enters the message. And in the second figure when the user clicks on the next button, then it asks to enter a passkey which is of four digit. This passkey should be provided to the receiver so that he/she can use it for decryption.

The above two figures are performed when the decryption process is carried on. In the first figure, the user will provide the path from where the image is get. And in the second figure when the user clicks on the next button, then it asks to enter a passkey which is of four digit. If this passkey matches with the sender's passkey, then it will show the desired message which is sent by the sender.

V. RESULT

After performing the above procedure, the encryption and decryption is done and thus the security is provided as the message is hidden in the image and only the person who knows the passkey can be able to see the message. Following is the image before and after applying the steganography-

Image before steganography

Image after steganography

VI. REFERENCES

[1] An LSB method of image steganographic Techniques by Jyotiprakash Patra, Lalit Kumar Jain and Himanshu Kumar Gawhade. International Journal of Engineering Research and Application on ISSN:2248-9662 vol5. Issue- 4 April 2015 Page 62-65.

[2] Y. Hsiao. C.C. Chang. and C.-S. Chan. Finding optimal least significant- bit substitution in image hiding by dynamic programming strategy. Pattern Recognition, 36:1583–1595, 2003 [3] C.K. Chan. and L.M. Cheng. Hiding data in images by simple lsb substitution. Pattern Recognition, 37:469–474, 2004.

[4] Y. K. Lee. and L. H.Chen. High capacity image steganographic model. IEE Proc.-Vision, Image and Signal Processing, 147:288–294, 2000

[5] C.F. Lin. R.Z. Wang. and J.C. Lin. Image hiding by optimal lsb substitution and genetic algorithm. Pattern Recognition, 34:671–683, 2001.

[6]S. M. Masud Karim, Md. Saifur Rahman, Md. Ismail Hossain "A New Approach for LSB Based Image Steganography using Secret Key", International Conference on Computer and Information Technology (ICCIT), Pages No. 286 –291,22-24 Dec., 2011.

[7]Kousik Dasgupta, J. K. Mandal, Paramartha Dutta, "Hash Based Least
Significant Bit Technique for Video Steganography (HLSB)", International Journal of Security, Privacy and Trust Management (IJSPTM), Vol. 1, Issue No. 2, April, 2012.

[8]Mamta Juneja, Parvinder Singh Sandhu, "Designing of Robust Image Steganography Technique Based on LSB Insertion and Encryption", International Conference on Advances in Recent Technologies in Communication and Computing, Pages No. 302 –305,27-28 Oct., 2009.

[9]Swati Tiwari, R. P. Mahajan, "A Secure Image Based Steganographic Model Using RSA Algorithm and LSB Insertion", International Journal of Electronics Communication and Computer Engineering (IJECCE), Vol. 3, Issue No. 1, 2012.

[10]Wien Hong, Tung-Shou Chen, "A Novel Data Embedding Method Using Adaptive Pixel Pair Matching",IEEE Transactions on Information Forensics and Security, Vol. 7,Issue No. 1,Pages No. 176 -184, Feb., 2012.

[11]Weiqi Luo, Fangjun Huang, Jiwu Huang, "Edge Adaptive Image Steganography Based on LSB Matching Revisited", IEEE Transactions on
Information Forensics and Security, Vol. 5, Issue No. 2, Pages No. 201 –214, June, 2010.

[12]Min-Wen Chao, Chao-hung Lin, Cheng-Wei Yu, Tong-Yee Lee, "A High Capacity 3D Steganography Algorithm",IEEE Transactions on Visualization and Computer Graphics,Vol. 15, Issue No. 2, Pages No. 274 –284, March-April, 2009.

[13]Mohammad Tanvir Parvez, Adnan Abdul-Aziz Gutub, "RGB Intensity Based Variable-Bits Image Steganography", Asia-Pacific Services Computing Conference, IEEE,Pages No. 1322 –1327, 9-12 Dec., 2008.

[14]Jing-Ming Guo, Thanh-Nam Le, "Secret Communication Using JPEG Double Compression", Signal Processing Letters, IEEE, Vol. 17, Issue No. 10, Pages No. 879 –882, Oct., 2010.

[15]Dr.Ekta Walia, Payal Jainb, Navdeep, "An Analysis of LSB & DCT based Steganography", Global Journal of Computer Science and Technology, Vol. 10, Issue No. 1, April, 2010.

AUTHOR'S PROFILE

	Dr.Jyotiprakash Patra obtained his B.E degree in Computer Science & Engg. from Biju Pattnaik Technical University, Orissa in the year 2004. He acquired M.E in Computer Science and Engineering from CSVTU, Bhilai, in 2008. MATS University, Raipur awarded him with the Ph.D. degree for his work in the field of Soft Computing in 2015. He has served in various positions in different Engineering colleges as Associate Professor and Head. Currently he is working with Shri Shankaracharya Institute of Professional Management & Technology, Raipur as Associate Professor in Department of Computer Science & Engineering. He has contributed Two books .His research interests include Algorithms and Cryptography.
	Karishma Rathod Pursuing Bachelor of Engineering in Computer Science from Shri Shankaracharya Institute of Professional Management and Technology. Area of interest: Cryptography.
	Udita Chauhan Pursuing Bachelor of Engineering in Computer Science from Shri Shankaracharya Institute of Professional Management and Technology. Area of interest: Cryptography.