Volume 3, Issue 8, August 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

LSB and MSB Based Steganography for Embedding Modified DES Encrypted Text

Prashanti .G* , Sandhya Rani.K, Deepthi.S
Department of Computer Science
VLITS, India

Abstract: ***Now a day's Secure Transmission of data is very important because of high rates of data theft and interception. This work deals with one of the encryption methodology i.e. modified data encryption standard algorithm that encrypts original text into insensible text (cryptography)and to make data more secure, the text is embedded into an image(steganography) .This means the text will not be visible to any one that intercepts this image. The modified DES here introduces a new method to enhance the performance of the Data Encryption Standard (DES) algorithm. This is done by replacing the 8×32 S-Box instead of 6×4 S-Box. The output of each S-Box undergoes AND and XOR operation before going to the permutation P. In this paper we also proposed a new operation Addition modulo instead predefined XOR operation applied during the 16 round of the standard algorithm which results a Cipher text. This cipher text is embedded into an image using the two techniques, LSB (Least Significant Bit) based steganography and MSB (Most Significant Bit) based steganography. The main advantage of this method is that if someone (hacker) wants to get the information at first the hacker should extract the cipher text from the image. This step is itself quite difficult and there is another security barrier of decrypting the cipher text to original text.***

Index Terms—DES, Encryption, Decryption, asymmetric cryptography, symmetric cryptography, image embedding, steganography, LSB and MSB based steganography, grayscale images.

I. Introduction

Cryptography is about concealing the content of the message or in other words it is the practice and study of techniques for secure communication in the presence of third parties (called adversaries). More generally, it is about constructing and analyzing protocols that overcome the influence of adversaries and which are related to various aspects in information security [15] such as

1. Confidentiality: Information in computer transmitted is accessible only for reading by authorized parties.
2. Authentication: Origin of message is correctly identified with an assurance that identity is not false.
3. Integrity: Only authorized parties are able to modify transmitted or stored information.
4. Non-Repudiation: Requires that neither the sender, nor the receiver of message be able to deny the transmission.
5. Access Control: Requires access may be controlled by or for the target system.
6. Availability: Computer system assets are available to authorized parties when needed.

Modern cryptography intersects the disciplines of mathematics, computer science, and electrical engineering. Applications of cryptography include ATM cards, computer passwords, and electronic commerce. Cryptography prior to the modern age was effectively synonymous with encryption, the conversion of information from a readable state to apparent insensible. Decryption is conversion from insensible to a readable state. Data encryption standard algorithm is one of the encryption algorithms that convert readable form of the text (Plain Text) into insensible text (Cipher Text). This cipher text is then embedded into a medium in such a manner that it can't be easily detected which is known as steganography. Combinational use of cryptography and steganography [3] provides best security than usage of individual (fig.1).

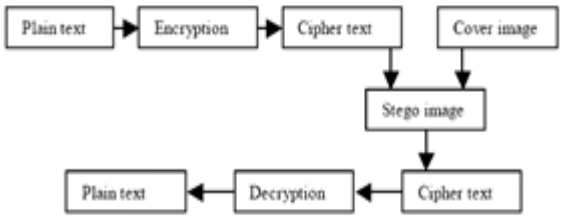

Figure 1: Combination of cryptography and steganography

II. Literature Review

A. The Scope of Cryptography

Cryptography encodes information in such a way that nobody can read it, except the person who holds the key. More advanced crypto techniques ensure that the information being transmitted has not been modified in transit. There is some difference in cryptography and steganography, in cryptography the hidden message is always visible, because information is in plain text form but in steganography hidden message is invisible.

A.1 Data Encryption Standard

In 1974, IBM proposed "Lucifer", an encryption algorithm using 64-bit keys. Two years later (1977), NBS (now NIST) in consultation with NSA made a modified version of that algorithm into a standard DES [14].The DES encryption scheme can be explained by the following figure.2

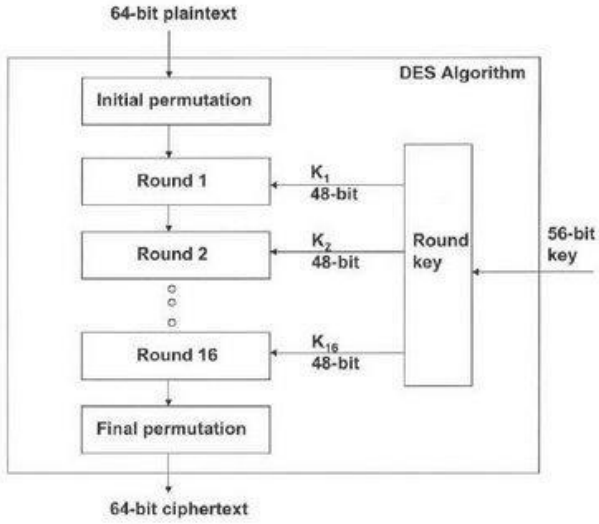

Figure 2: original DES encryption

The plain text is 64 bits in length and the key in 56 bits in length. Longer plain text amounts are processed in 64-bit blocks [15]. The main phases in the processing of the plain text are, Initial Permutation (IP*):* The plaintext block undergoes an initial permutation. 64 bits of the block are permuted. A Complex Transformation*:* 64 bit permuted block undergoes 16 rounds of complex transformation. Subkeys are used in each of the 16 iterations. 32-bit swap*:* The output of 16th round consists of 64bits that are a function of input plain text and key.32 bit left and right halves of this output is swapped. Inverse Initial Permutation (IP^{-1}*):* The 64 bit output undergoes a permutation that is inverse of the initial permutation.

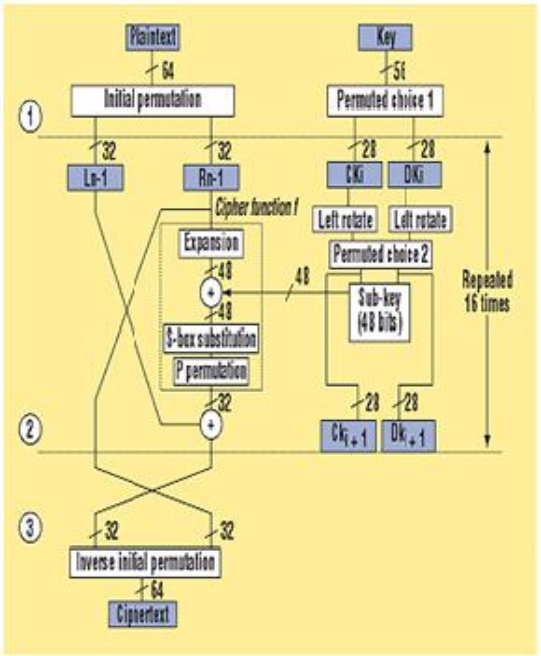

Figure 3: details of single round

The internal structure of a single round is shown above. The left and right halves of each 64-bit intermediate value are treated as separate 32-bit quantities, labeled L (left) and R (right). As in any classic Feistel cipher, the overall processing at each round can be summarized in the following formulas:

Li = Ri-1.

Ri = Li-1$\oplus$ F(Ri-1, Ki) Where Function F can be described as P(S(E(R(i-1)) K(i))).

The round ke$\oplus$$X_i$ is 48 bits. The R input is 32 bits. This R input is first expanded to 48 bits by using a Expansion Permutation (E) table that defines a permutation plus an expansion that involves duplication of 16 of the R bits. The resulting 48 bits are XORed with K_i. This 48-bit result passes through a substitution function (S-Box) that produces a 32-bit output, which is permuted as defined by Permutation Function (P). The role of the S-boxes in the function F is illustrated in figure below. The substitution consists of a set of eight S-boxes, each of which accepts 6 bits as input and produces 4 bits as output. For example, in S_1 for input 011001, the row is 01 (row 1) and the column is 1100 (column 12). The value in row 1, column 12 is 9, so the output is 1001.

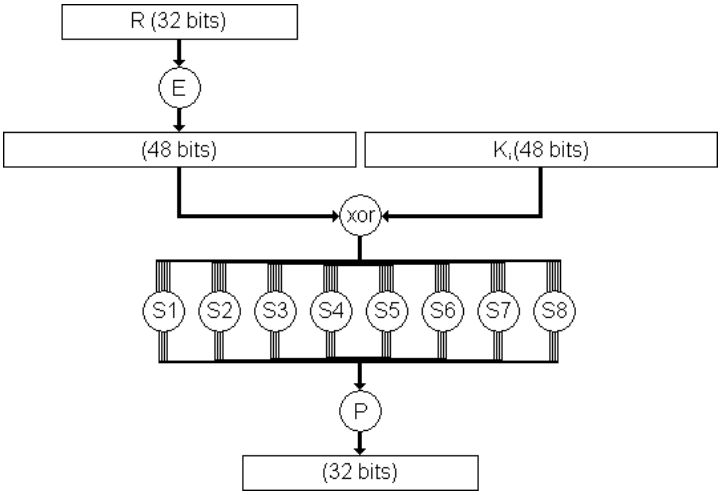

Figure 4: S-Boxes in function F

64-bit key is used as input to the algorithm. The bits of the key are numbered from 1 through 64; every eighth bit is ignored.. The key is first subjected to a permutation governed by a table labeled Permuted Choice One as shown in figure 3. The resulting 56-bit key is then treated as two 28-bit quantities, labeled C_0 and D_0. At each round, C_{i-1} and D_{i-1} are separately subjected to a circular left shift, or rotation. They also serve as input to Permuted Choice Two which produces a 48-bit output that serves as input to the function F (R_{i-1}, K_i).

As with any Feistel cipher, decryption uses the same algorithm as encryption, except that the application of the sub keys is reversed.

B. Steganography

Steganography is the art of hiding the fact that communication is taking place, by hiding information in other information[2].The goal of Steganography1 is to mask the very presence of communication making the true message not discernible to the observer.

Steganography is concern with the hiding of text in information like image, text, audio, and video. There are 4 different types of steganography [5]-

Text steganography: using digital files is not used very often since text files have a very small amount of redundant data.

Audio/Video steganography: is very complex in use.

Image steganography: is widely use for hiding process of data. Because it is quite simple and secure way to transfer the information over the internet.

Image steganography has following types:

a. Transform domain
- Jpeg

b. Spread spectrum
- Patch work

c.. Image domain
- LSB and MSB in BMP
- LSB and MSB in JPG

It is most efficient (in term of data hiding) method of image steganography. Because the intensity of image is only change by 1 or 0 after hiding the information. Change in intensity is either 0 or 1 because the change at last bit.

Methods of hiding data in digital images:

There are two types of methods in digital images:-

1. LSB (Least Significant Bit)

LSB (Least Significant Bit) is a method for embedding data into cover image. The least significant bit of each pixel of an image is altered to a bit of a message that is to be hidden.

2. *MSB (Most Significant Bit)*

This method considers the value of the MSB of the Pixels of the image for data hiding. The MSB bits of each pixels of an image are changed to a bit of a secret message that is to be hidden.

A. Steganography Versus Cryptography

The comparison between steganography and cryptography [3] is illustrated from the following table 1.

Steganography	Cryptography
Unknown message passing	Known message passing
Steganography prevents discovery of the very existence of communication	Encryption prevents an unauthorized party from discovering the *contents* of a communication
Little known technology	Common technology
Technology still being developed for certain formats	Most of algorithm known by all
Once detected message is known	Strong current algorithms are currently resistant to attack, larger expensive computing power is required for cracking
Steganography does not alter the structure of the secret message	Cryptography alter the structure of the secret message

Table 1: Comparison

III. PROPOSED ALGORITHM

A. Algorithm for encryption:

A new improvement to the DES algorithm is made which makes the use of the new operation known as addition modulo 2^{32}(+) instead of the XOR operation in the original des algorithm and also 8×32 S-box is used instead of 6×4 S-box of original DES algorithm.

Example: x and y are the Inputs

X=1100 1000

Y=1000 1111

Addition modulo is obtained as (x+y) mod2^8

(1100 1000+1000 1111) mod2^8

=(1 0101 0111) mod 2^8

s=0101 0111

To find original x value perform following operation

$X=x^1+(-y)$

To obtain (-y) = 2^8-y =>256-143=113

Perform X^1+ (-y) which results Original x

0111 0001

0101 0111

1100 1000 ← original x value

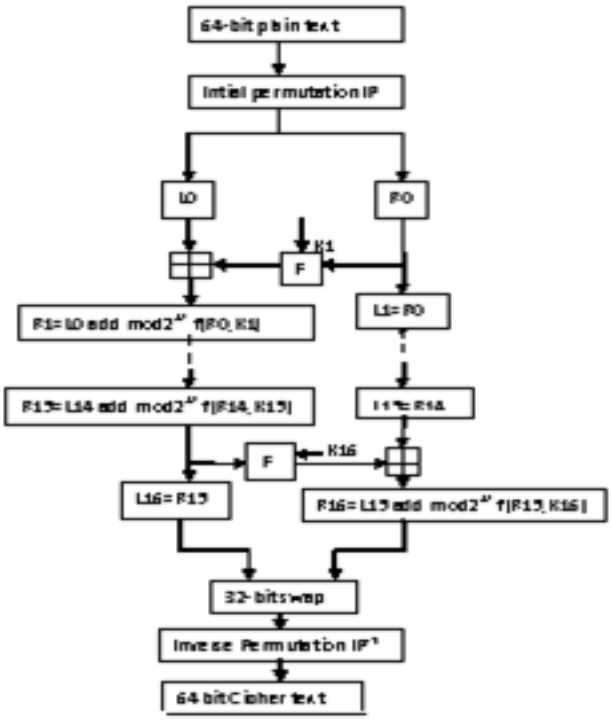

Figure 5: The proposed Des algorithm

Algorithm of modified data encryption standard with addition modulo operation:
INPUT: plaintext p1 . . . p64; 64-bit key K=k1 . . . k64 (includes 8 parity bits).
OUTPUT: 64-bit cipher text block C=c1 . . .c64.

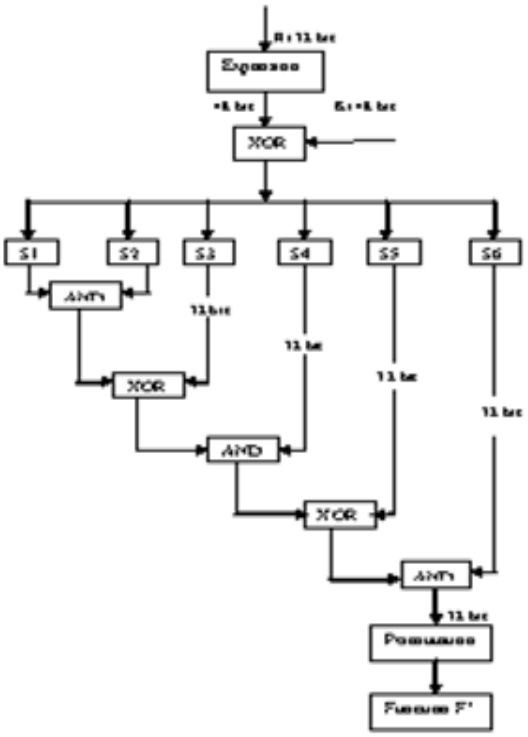

Figure 6: Function F Design for proposed Algorithm.

1. (key schedule) Compute sixteen 48-bit round keys Ki, from K.
Note: Where k=64 bits out of which 8 parity bits are discarded outcome is 56 bits, after Left circular shift and PC which results 48 bit key.
2. (L0, R0) □□IP (p1, p2,. . .p64) (Use IP Table to permute bits; split the result into left and right 32-bit halves L0=p58p50 . . . p8,R0=p57p49 . . . p7)
3. (16 rounds) for i from 1 to 16, compute Li and Ri as follows:
    3.1. Li=Ri-1
    3.2. Ri = Li-1 additionmodulo2^{32} f (R i-1, Ki)

where f(Ri-1, Ki) = P(S(E(Ri-1) XOR Ki)), computed as follows:
(a)Expand Ri-1 = r32r1r2 . . . r32 r1 from 32 to 48 bits, M□E(Ri-1).
(b)M' □M XOR Ki. Represent M ' as eight 6-bit character strings: M'= (B1 . . . B8)
(c)M" □ F' where function F'= ((((s1Λs2) XOR s3) Λ s4)XOR s5) Λ s6). Here si(Bi) maps to the 8/32 S-Box that consist of 256 entries.
(d)M"' □P (M"). (Use P per table to permute the 32 bits of M"=m1m2 . . . m32, yielding m16m7 . . . m25)
4. b1b2 . . . b64 □ (R16, L16). (Exchange final blocks L16, R16.)
5. C □□IP^{-1} (b1b2 . . . b64).
6**.** End.

Here, using this proposed algorithm an example is solved . Our Input Message is 0123456789ABCDEF using this proposed algorithm the plain text is finally converting into cipher text of E46037E66BA9FEB8. For the input given above the following TABLE.II shows 16 rounds and resulting output.

Input: 0123456789ABCDEF

Output: E46037E66BA9FEB8

TABLE II Result of proposed DES 16 Rounds

Round	Sub Key(Ki)	Left Bits(Li)	Right Bits(Ri)
IP		CC00CCFF	F0AAF0AA
1	1B02EFFC7072	F0AAF0AA	CE620D18
2	79AEB9DBC9EF	CE620D18	22D1F6C2
3	55FC8A426F99	22D1F6C2	DE6B4E29
4	72ADD6DB351D	DE6B4E29	AB5BFB24
5	7CEC07EB53AE	AB5BFB24	E4977040
6	63A53E507B2F	E4977040	2BB88B30
7	EC84B7F618BC	2BB88B30	2531F060
8	F78A3AC13BFB	2531F060	D0B8BCB2
9	E0DBEBEDE781	D0B8BCB2	7436155C
10	B1F347BA464F	7436155C	56FFC0E2
11	215FD3DED386	56FFC0E2	D66AAA00
12	7571F59467E9	D66AAA00	587FC4EA
13	97C4D1FABA41	587FC4EA	A6FFD004
14	5F43B7F2E73A	A6FFD004	FAA3CA30
15	BF918D3D3F0A	FAA3CA30	E9FFF05C
16	EB3D8B0E17F5S	E9FFF05C	5BC44D34

A. Algorithm of steganography

After the converting our information into ciphertext we need to embed that data in the image which is our second step of security.

B.1. LSB Based Steganography for grayscale

Grayscale is a range of shades of gray without apparent color. The darkest possible shade is black, which is the total absence of transmitted or reflected lightFor every pixel in a red-green-blue (RGB) grayscale image, R = G = B[8]. The lightness of the gray is directly proportional to the number representing the brightness levels of the primary colors. Black is represented by R = G = B = 0 or R = G = B = 00000000, and white is represented by R = G = B = 255 or R = G = B = 11111111. Because there are 8 bits in the binary representation of the gray level, this imaging method is called 8-bit grayscale[9].

For example 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 ciphertext E46037E66BA9FEB8 obtained from modified DES algorithm first we have to convert the ciphertext into decimal numbers.suppose if we take The first two hexa-decimal numbers of our ciphertext is E4 which is first converted to binary form as

E4 → 11100100 we would replace the LSBs of these pixels to have the following new grayscale values:

11010011
01001011
10010111
10001100
00010100
01010111
00100110
01000010.

Note that, on average, only half the LSBs need to change. The difference between the cover (i.e. original) image and the stego image will be hardly noticeable to the human eye. Figure 7.1(a), (b) that show a cover image and a stego image (with data is embedded); there is no visible difference between the two images.

Algorithm to embed text message using Grayscale Image

Step 1: Read the cover image and text message which is to be hidden in the cover image.
Step 2: Convert text message in binary.
Step 3: Calculate LSB of each pixels of cover image.
Step 4: Replace LSB of cover image with each bit of secret message one by one.
Step 5: Write stego image

Algorithm to retrieve text message using Grayscale Image

Step 1: Read the stego image.
Step 2: Calculate LSB of each pixel of stego image.
Step 3: Retrieve bits & convert each 8 bit into character.

Images of LSB based steganography**:**

Figure 7.1(a) Original Image

Figure 7.1(b) Stego Image

B.2 MSB Based Steganography for grayscale:

In computing, the most significant bit (msb also called the high-order bit) is the bit position in a binary number having the greatest value. The msb is sometimes referred to as the left-most bit due to the convention in positional notation of writing more significant digits further to the left [4]. MSB Steganography is inserting secret message in Most Significant bit of the pixel of image.

For example 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 ciphertext E46037E66BA9FEB8 obtained from modified DES algorithm first we have to convert the ciphertext into decimal numbers.suppose if we take The first two hexa-decimal numbers of our ciphertext is E4 which is first converted to binary form as

E4 → 11100100 we would replace the MSBs of these pixels to have the following new grayscale values:

11010011
11001010
10010110
00001100
00101000
11010110
00100111

01000011.continue this process until the entire ciphertext is embedded .Here embedding data in the most significant bits produces detectable results fig 7.2 (a), (b) shows the original image and stego image, where there is a detectable difference between the two images.

Algorithm to embed text message using Grayscale Image

Step 1: Read the cover image & text message, which is to be hidden in the cover image.
Step 2: Convert text message into binary.
Step 3: Calculate MSB of each pixel of cover image.
Step 4: Replace MSB of cover image with each bit of secret message one by one.
Step 5: Write stego image.

Algorithm to retrieve text message using Grayscale Image

Step 1: Read the stego image.
Step 2: Calculate MSB of each pixel of stego image.
Step 3: Retrieve bits & convert each 8 bit into character
Images of MSB Based Steganography:

Figure 7.2(a) Original Image

Figure 7.2(b) Stego Image

B.3. LSB Based Steganography for color Images

In a computer, images are represented as arrays of values. These values represent the intensities of the three colors R (Red), G (Green) and B (Blue), where a value for each of three colors describes a pixel. Each pixel is combination of three components(R,G and B)[6]. Here we take the result of modified DES algorithm discussed in proposed algorithm and the cipher text obtained is

E46037E66BA9FEB8

The first two hexa-decimal numbers of our ciphertext is E4 which is first converted to binary form as

E4 → 11100100

This is to be embedded into an image by using the LSB algorithm which replaces least significant bits of pixels of an image. For example a grid for 3 pixels of a 24-bit image can be as follows:

00101101 00011100 11011100
10100110 11000100 00001100
11010010 10101101 01100011

When the number E4, which binary representation is 11100100, is embedded into the least significant bits of this part of the image, the resulting grid is as follows:

00101101 00011101 11011101
10100110 11000100 00001101
11010010 10101100 01100011

Likewise entire cipher text is going to be embedded into the LSB bits of Pixels of the Image fig 8.1(a),(b) shows a original image and a stego image(with data is embedded): there is no visible difference between the two images.

Algorithm to embed text message using color Image

1.read the pixels of a given image and store in an array called image-array
2.Convert the message that is to be embedded into binary message.
3.read this binary message into an array called message-array.
3.choose the pixel from the image-array and pick the characters from the message- array and place it in the LSB of pixel .
4. repeat step4 till all characters are embedded.
5.obtained image will be stego image that has hidden data.

Images of LSB Based Steganography: with color Image

Figure 8.1(a) Original Image

Figure 8.1 (b) Stego Image

B.4. MSB Based Steganography for color Images:

Here we take the result of modified DES algorithm discussed in proposed algorithm the cipher text obtained form that is

E46037E66BA9FEB8

The first two hexa-decimal numbers of our ciphertext is E4 which is first converted to binary form as

E4 →11100100

This is to be embedded into an image by using the MSB based stegnography which replaces most significant bits of pixels of an image.

For example a grid for 3 pixels of a 24-bit image can be as follows:

00101101 **0**0011100 **1**1011100
10100110 **1**1000100 **0**0001100
11010010 **1**0101101 **0**1100011

When the number E4, which binary representation is 11100100, is embedded into the Most significant bits of this part of the image, the resulting grid is as follows:

10101101 **1**0011100 **1**1011100
00100110 **0**1000100 **1**0001100
01010010 **0**0101101 01100011

Likewise entire cipher text is going to be embedded into the MSB bits of Pixels of the Image Fig 8.2(a),(b) shows the original image and stego image(that contains data embedded in the most significant bit). This produces a detectable results between the two images

Algorithm to embed text message using color Image

1.read the pixels of a given image and store in an array called image-array
2.Convert the message that is to be embedded into binary message.
3.read this binary message into an array called message-array.
3.choose the pixel from the image-array and pick the characters from the message- array and place it in the MSB of pixel .
4. repeat step4 till all characters are embedded.
5.obtained image will be stego image that has hidden data.

Images of MSB Based Steganography: with color Image

Figure 8.2(a) Original Image

Figure 8.2 (b) Stego Image

IV. Conclusion

Cryptography deals with taking a message and making it appear to random noise, unreadable to an outside world. It does nothing to hide the presence of message to itself. Often stegnography is used in conjunction to cryptography so that message remains unreadable even if detected. This paper is to create across platform that can effectively encrypt a message and hide it

inside a digital image file. The encryption algorithm is the modified DES algorithm which converts readable message into unreadable form (cipher text) and then this is embedded into the image using LSB and MSB methods of stegnography. As there are many application of combining cryptography and stegnography like it allows for two parties to communicate secretly and covertly. It allows for some morally conscious people to safely whistle blow on internal actions. One of the other main uses is for the transportation of high level or top secret documents between international governments.

Acknowledgment

We take this opportunity to acknowledge those who have been great support and inspiration through the research work. Our sincere thanks to Mrs.B.RenukaDevi Head of the department of CSE to her diligence and motivation. Special thanks to Vignan's Lara Institute of Science and Technology, for giving us such a nice opportunity to work in the great environment and for providing the necessary facilities during the research and encouragement from time to time. Thanks to our colleagues who have been a source of inspiration and motivation that helped to us. And to all other people who directly or indirectly supported and help us to fulfill our task. Finally, we heartily appreciate our family members for their motivation, love and support in our goal.

References

[1]. "Use of image to secure text message with the help of LSBreplacement "Saurabh Singh, Gaurav Agarwal Invertis institute of Engineering and Technology, Bareilly, India

[2]. "Cryptography and Steganography – A Survey" A.Joseph , Raphael,Dr.V Sundaram, Int. J. Comp. Tech. Appl., Vol 2 626-6302

[3]. Mr. Rohit Garg," Comparison Of Lsb & Msb Based Steganography In Gray-Scale Images "International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 8, October – 2012 ISSN: 2278-0181

[4]. Mr . Vikas Tyagi*1, Mr. Atul kumar2, Roshan Patel, Sachin Tyagi, Saurabh Singh , Gangwar "Image steganography using least significant bit with cryptography"

[5]. Dr. Ekta Walia, Payal Jain,Navdeep " An Analysis of LSB & DCT based Steganography

[6]. Joshua Michael Buchanan,"Creating a robust form of steagonography"

[7]. Vijay kumar sharma ,vishal shrivastava" a steganography algorithm for hiding image in Image by improved lsb substitution by imizeDetection

[8]. V. Lokeswara Reddy, Dr. A. Subramanyam**"** Implementation of LSB Steganography and its Evaluation for Various File Formats" Int. J. dvanced Networking and Applications 868 Volume: 02, Issue: 05, Pages: 868-872 (2011)

[9]. Neeta Deshpande, Kamalapur Sneha, Daisy Jacobs, "Implementation of LSB Steganography and Its Evaluation for various Bits‖ Digital Information Management", 2006 1st International Conference on. 06/01/2007; DOI: 10.1109/ICDIM.2007.369349

[10]. Mamta Juneja, Parvinder S. Sandhu & Ekta Walia "Appliaction of LSB Based Steganography Technique for 8- bit Color Images".

[11] Jessica Fridrich, Miroslav Goljan and Rui Du State, "Detecting LSB steganography in Color and Gray-scale Image".

[12] "New Approach of Data Encryption Standard Algorithm Shah Kruti R., Bhavika Gambhava."

[13] J.Orlin Grabbe ―The DES Algorithm Illustrated‖

[14] W. Stallings, Cryptography and Network Security: Principles and Practices, 5th ed., Prentice Hall, 1999.

Authors Profile :

Prashanti.G is Assistant Professor at Department of Computer Science and Engineering, Vignan's Lara Institute of Technology and Science, Vadlamudi, Guntur, Andhra Pradesh, India. She has received her M.Tech degree from Vignan Engineering College, Vadlamudi,, Guntur, Andhra Pradesh, India

Deepthi.S is Assistant Professor at Department of Computer Science and Engineering, Vignan's Lara Institute of Technology and Science, Vadlamudi, Guntur, Andhra Pradesh, India. She has received her B.Tech degree from K.L College of Engineering, Guntur, Andhra Pradesh, India in
2008. She has received her M.Tech degree from Nalanda Institute of Engineering and Technology, Guntur, Andhra Pradesh, India.

Sandhya Rani.K is Assistant Professor at Department of Computer Science and Engineering, Vignan's Lara Institute of Technology and Science, Vadlamudi, Guntur, Andhra Pradesh, India. She has received her M.Tech degree from Vignan Engineering College, Vadlamudi,, Guntur, Andhra Pradesh, India

