Integration between Cryptography and Steganography to Hide Secret Message inside an Image

Article · January 2017

CITATIONS
0

READS
30

2 authors, including:

Ahmed Nashaat
Kirkuk University
11 PUBLICATIONS **12** CITATIONS

SEE PROFILE

Journal of Information Engineering and Applications
ISSN 2224-5782 (print) ISSN 2225-0506 (online)
Vol.7, No.4, 2017

Integration between Cryptography and Steganography to Hide Secret Message inside an Image

Ahmed Nashaat SHAKIR[1,2*] Sibel TARIYAN[3]
1. Cankaya University, Ankara, Turkey
2. Department of Software Engineering, Kirkuk University, Kirkuk, Iraq
3. Department of Computer Engineering, Cankaya University, Ankara, Turkey

Abstract
The information hiding is progression of hiding the details of a function or object or both of them. On the other hand it represents an important way that used in data security. Another name for information hiding is the steganography, which hides the data inside another data such as embed text inside image or image inside another one. The steganography techniques were used from ancient times and through uses of many mechanics and different ways, such as writing in invisible ink in the Greek Testament, while Cryptography is the process of hiding information by encrypt this data using a complex algorithms. Cryptography is used when collaborating over an untrusted intermediate such as internet. The steganography and cryptography work similarly but in different contexts. In this study, an integration of cryptography and steganography to produce an efficient and robust model has been presented. In terms of cryptography, the Data Encryption Standard (DES) algorithm is implemented, whereas in steganography, the Least Significant Bit (LSB) algorithm is used. Our results show efficient time implementation and a robust algorithm mechanism in terms of peak signal-to-noise ratio (PSNR), signal-to-noise ratio (SNR) and mean square error (MSE).
Keywords: Steganography, Cryptography, Data Hiding, Data Encryption Standard (DES), Least Significant Bit (LSB).

1. Introduction

The Internet nowadays represents an essential medium that is being used to perform most of our daily tasks in addition to connect ten millions of people. Moreover, it is increasingly being used to perform web-based operations and business, such as bank transactions, e-commerce, and online shopping.

Cryptography is one of the most effective approaches that have been taken to change plain information into ambiguous meaningless data. It has played a key role in many data security fields even before the invention of technology and the Industrial Revolution. Cryptography is a method of sending or receiving secret information in an Some experts have claimed that the appearance of cryptography techniques dates back to the invention of writing, with different perspectives ranging from diplomatic missives to war plans. In spite of the variety of cryptography techniques, cryptography itself represents the science of transforming plain (clear) text into cipher (cryptogram) text. This process of changing text, or messages, is also known as encryption. The process of reversing an encrypted text to its original state (plain text) is called decryption. Both of these processes are controlled by a single key or multiple keys [1]. Figure 1 shows the basic encryption diagram.

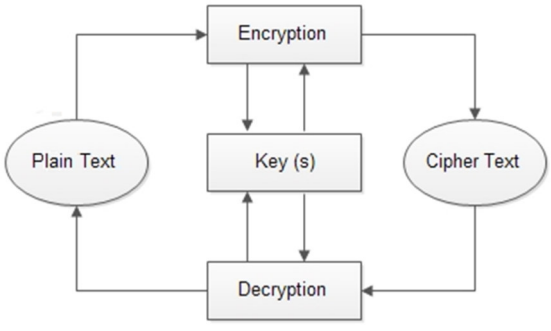


Figure 1. Basic Encryption Diagram

Steganography is implemented in a manner that makes the information unseen but existent [2]. It is a procedure that makes use of human perceptive sense of visual or aural redundancy to digital multimedia, and that embeds the secret information in the public media to transfer digital media carrying confidential information to achieve covert communications [3].

2. Related Workes

1. Data Encryption Standard (DES)

DES Operation Structure uses a 56-bit key, which is separated into eight blocks with 7 bits per block. The last bit for each block is assigned to an equity bit (0 or 1). In spite of DES using 64 bits for encryption, only 56 bits are actually used. This is for computational purposes and to ease the randomness process. Figure 2 shows DES structure

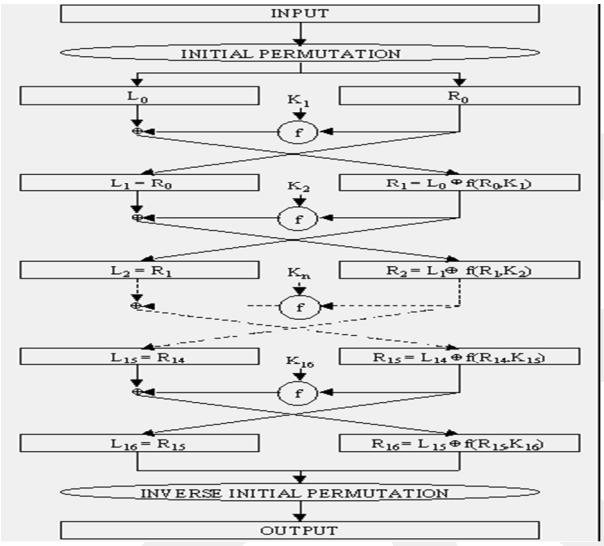


Figure 2. DES structure

After this, DES assigns 64 bits for the plain text with 16 rounds of replacements (permutation) steps, and substitution. The overall initial steps of DES are as follows:

A. To encrypt a 64-bit block, it should be assigned to an initial permutation (IP). In this step, each bit is transferred to a new position; for instance, moving the sixth, seventh and eighth bits to the 55th, 56th and 48th positions respectively.

B. All 64 bits, which are produced from the previous step, are divided into two blocks of 32 bits, known as left and right, each of which are assigned to an initial value, namely L_0 and R_0.

C. After block division is completed, a specific formula is implemented in 16 rounds, for both L an R. The formula can be seen as follows:

$L_n = R_n - 1$ .....……..(1)

$R_n = L_n - 1\ XOR\ f(R_{n-1,}K_n)$……. (2)

Where *n* represents the number of rounds (i.e. 1 to 16). For each step in the process, each L block is simply taken from the previous block (R block). Then, the new value of R is calculated by applying a bit-by-bit XOR of the L block within the outputs of applying the DES algorithm (f) to the prior R and K_n, where K_n represents a value of 48 bits derived from the 64-bit DES key. For each round, every K_n differs from the other according to the standard key schedule algorithm.

D. The output of the last round (L16, R16) is merged to produce 64 bits and is reproduced in an inverse IP to be (IP-1). After that, the position of the bits is reordered to the original values. In order to do that, the 55th, 56th and 48th are set back to the values of 6th, 7th, and 8th respectively [4,5,6 and 7] .

2. Least Significant Bit (LSB)

Least Significant Bit (LSB) this is one of the most popular techniques used to hide information. LSB is a simple method to perform steganography (B.S Champakamala, K. Padmini, .D. K. Radhika) [8].

LSB embeds data into an image so that the data cannot be detected or observed by a normal observer. This technique replaces some of the image pixels with hidden information inside an image. Although other techniques embed data inside images, LSB does the same work with least significant bits. This process contributes to reducing color variation such that it makes the changes almost undetectable. Such as change is a color numeric value by one and the second change by two and so on. Secret Key Cryptography (SKC) In (Zomaya Y., Seredynski F., and Bouvry P., 2003) [9].

3. Combining encryption and data hiding

Encryption and data hiding can be combined into one model to enhance the performance of data security. In the following section, a number of studies, and research that deals with a crypto-stegano approach, will be

demonstrated. Using cryptography or steganography separately can be combined with some issues. A study presented in (Lyu S., and Farid H., 2006) [10], implemented both steganography and cryptography. The method evolved and iterative process design. In addition, the encoding properties were tested to ensure its functionality and performance. A number of breaking methods were applied to the proposed method and the method presented good performance in terms of security. A new project of integrating cryptography and steganography was developed in (Narayana S., and Prasad G., 2010) [11].

In this paper, we propose a new Cryptography and Steganography method to produce efficiency for hiding secret data in image. In this method, we propose a mechanism that suggests integration between the improved Data Encryption Standard (DES) cryptography algorithm by using an irrational number (to increase the randomness of the sub key generation used in the DES algorithm), and a data hiding technique by using the Least Significant Bit (LSB) steganography algorithm. Firstly, we will encrypt the data with the key using the improved Data Encryption Standard (DES). We then take this encrypted data and put them into the host image by using the Least Significant Bit (LSB) steganography algorithm. In addition, our proposed mechanism covers both the encryption and decryption algorithm for both the Data Encryption Standard (DES) and the Least Significant Bit (LSB). Advanced data protection can be obtained by combining both cryptography and steganography into one model.

In section 2. we have related works. In section 3, the algorithm has been described. The experiments results are given in section 4. At last, in section 5, conclusions has been presented.

3. Proposed Approach

3.1 Proposed algorithm

In this section, we are going to discuss both the Data Encryption Standard (DES) and the Least Significant Bit (LSB), and how our mechanism works along with implementations and the test part for each algorithm.

3.1.1 Improved DES Cryptography Algorithm by Using the Irrational Number

DES Operation Structure (DES) normally uses a 64-bit structure distributed as follows: a 56-bit key, which is separated into 8 blocks with 7 bits per block. This means that DES actually uses only 56 bits with 8 bits being used in randomness processes. Overall, DES assigns a 64-bit key for the plain text with 16 rounds of replacement (permutation) steps and substitution. It is clear that this method is not practical or efficient in the present day. To solve this problem, we suggest using the improved data encryption standard (DES) 64-bit key cryptography algorithm with an irrational number. An irrational number is used here for two reasons. First, it extends the key space in the DES algorithm and increases the probabilities of the sub-keys in each group. A normal DES algorithm is shown in Figure 3. Figure 4 shows the DES algorithm based on an irrational number.

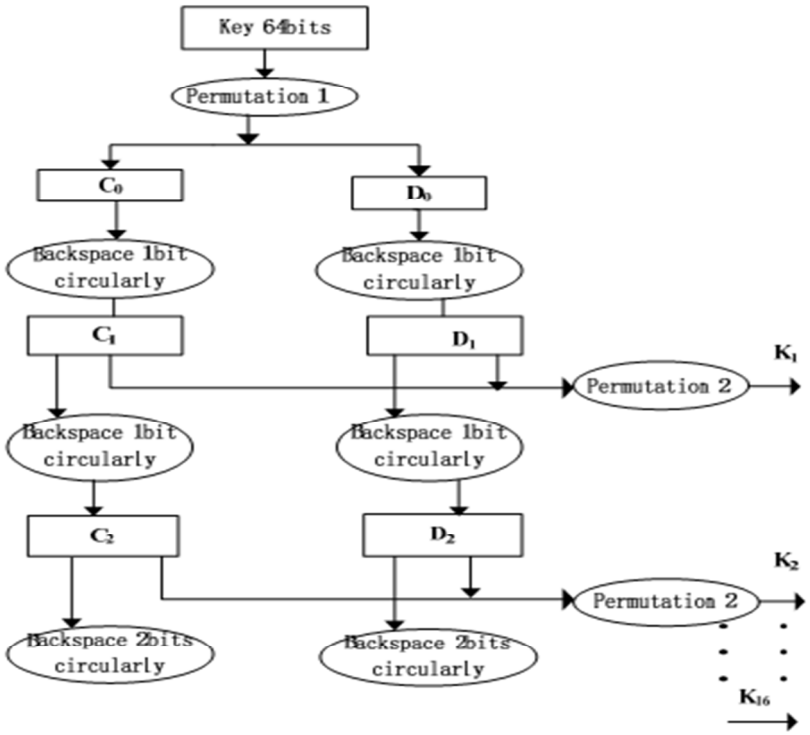


Figure 3. Normal DES algorithm sub-key generation

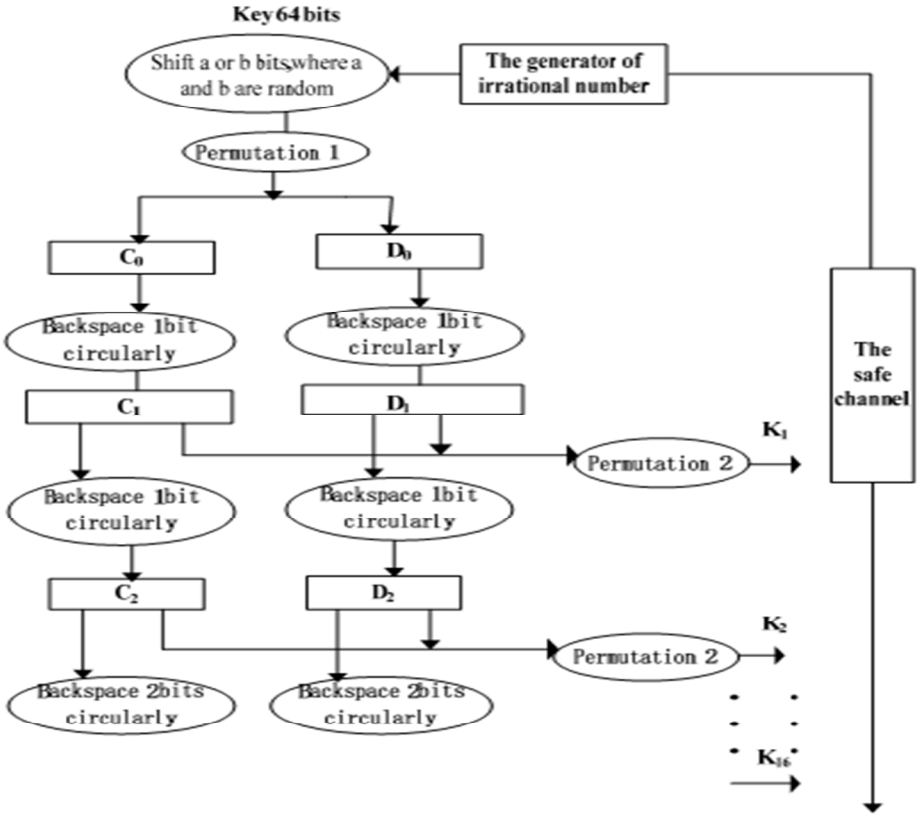


Figure 4. DES Algorithm Based on an Irrational Number

As shown in the figure above, 'a' and 'b' are measured based on an irrational number. This information of 'a' and 'b' is used in the shifting process of the production key. The irrational number will be part of this key, which will increase the arbitrariness in the DES sub key, and consequently, increase the robustness of the DES algorithm. Our sub key operates as follows: when two-digit numbers are nominated after two irrational number points in π arbitrarily, 'a' and 'b' will be measured based on the following: if 'a' is equivalent to the first irrational number selected, and 'b' is equivalent to the other irrational number, this will allow an exclusive-or (X-OR) circuit to produce an (X-OR) between these two numbers and another two numbers selected in the same way. If we obtain an odd result, it means it will shift 'a' bits; if we obtain an even result, 'b' bits will be shifted.

3.1.1.1 DES Encryption Process

Inputs: Plain Text Message 256 bit, Key 64-bit, Irrational number.

Output: Cipher Text Message

1) Read the plain text message from the input window.
2) Read the host key from the input window.
3) Read the irrational number generated by the code.
4) Process of the DES rounds and initial permutation.
5) Process of the cipher function, E-XOR operation with the round key data (E-XOR operation is fed into an S-Box array).
6) E-XOR operation between the key data and round data.
7) Inverse initial permutation.
8) Cipher text message is produced.

Figure 5 shows our work diagrams for the DES algorithm.

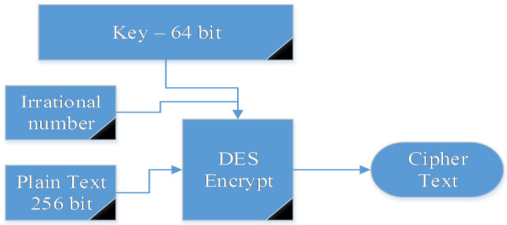


Figure 5. DES Encryption Process

3.1.1.2 DES Decryption Process
Inputs: Cipher Text Message, Key 64-bit, Irrational number.
Output: Plain Text Message 256 bit

1) Read the Cipher Text Message from the input window.
2) Read the Host Key from input window.
3) Read the irrational number generated by the code.
4) Reverse the process of the DES Rounds and the initial permutation.
5) Process of Cipher Function, E-XOR operation with the round key data (E-XOR operation fed into an S-Box array).
6) E-XOR operation between the key data and the round data.
7) Inverse Initial Permutation.
8) Plain text message is produced.

Figure 6 shows the DES decryption process.

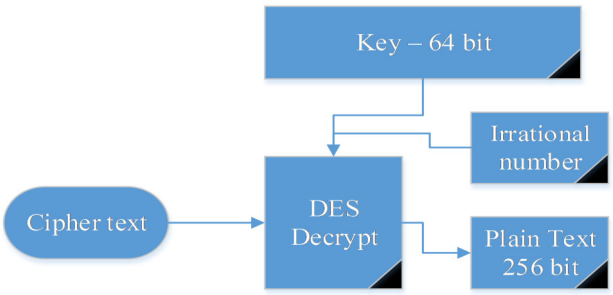


Figure 6. DES Decryption Process

3.1.2 Least Significant Bit (LSB) Steganography Algorithm
We used a 256×256 bitmap as a cover image RGB component and hide the 256-bit cipher text in a mutable location within a cover image. The main purpose of using both techniques is to increase the robustness of our algorithm.
Figures 7 and 8 show the working mechanism of the least significant bit algorithm.

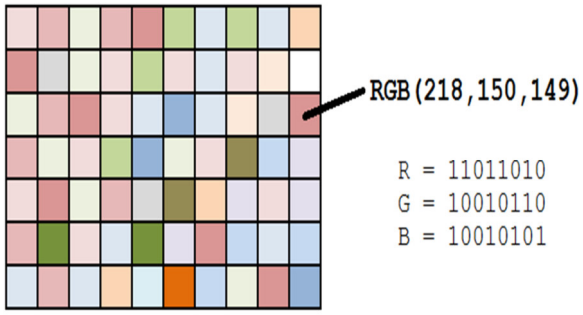


Figure 7. The Working Mechanism of Least Significant Bit Algorithm

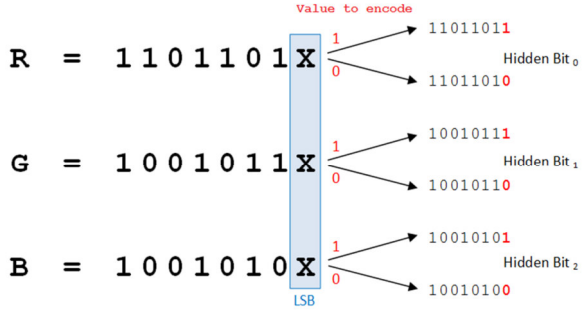


Figure 8. The Selection Mechanism of the Least Significant Bit Algorithm

We assume that we have cover image “I (i, j)” and we want to create a steganography image “IS (i, j)” and our message bit will be “Mb”. Then the message embedding process is as follows:
IS (i, j) = I (i, j) - 1, if LSB (I (i, j)) = 1 and Mb = 0…(1)
IS (i, j) = I (i, j), if LSB (I (i, j)) = Mb …… (2)
IS (i, j) = I (i, j) + 1, if LSB (I (i, j)) = 0 and Mb = 1 .. (3)

The result will produce a steganography image "IS (i, j)." The extraction process of the message is the reverse process of the above equation, as follows:

IS (i, j) = I (i, j) +1, if LSB (I (i, j)) = 1 and Mb = 0 … (4)

IS (i, j) = I (i, j), if LSB (I (i, j)) = Mb …….. (5)

IS (i, j) = I (i, j) - 1, if LSB (I (i, j)) = 0 and Mb = 1 …. (6)

3.1.2.1 LSB Embedding Algorithm

Inputs: Host image (256×256), and Cipher Text Message

Output: Steganography image (256×256)

1) Read cipher text message from text file that results from the previous encryption process by the improved DES algorithm.
2) Read host cover image.
3) Read the RGB component from the host cover image where the message should be embedded.
4) Read the last bit in each pixel of the host cover image in order to embed our cipher text message.
5) Embed the cipher message bits into the LSB location of each host cover message pixel.
6) Production of the steganography image.

The LSB embedding process is shown in Figure 9 below.

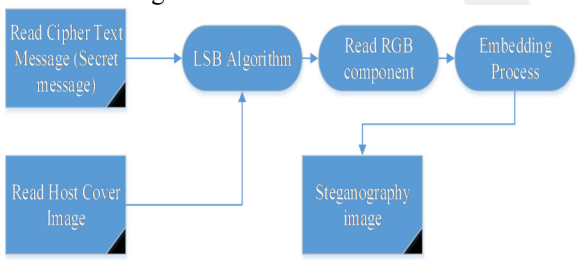


Figure 9. LSB embedding process

3.1.2.2 LSB Extraction Algorithm

Input: Steganography image (256×256)

Output: Cipher Text Message (Secret message)

1) Read Steganography image (256×256).
2) Read the RGB component for each pixel in the steganography image.
3) Extract the last bit of each pixel in the steganography image.
4) Convert each pixel bit into a decimal value.
5) Extract message value to the secret message text file.

The LSB extraction process is shown in Figure 10 below.

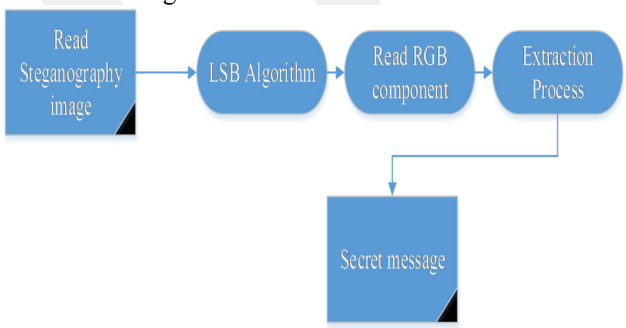


Figure 10. LSB Extraction Process

4. Experimental Result

4.1 Overall Workbench

The integration between the improved Data Encryption Standard (DES) cryptography algorithm by using an irrational number, and the data hiding technique by using the Least Significant Bit (LSB) steganography algorithm is shown in Figure 11.

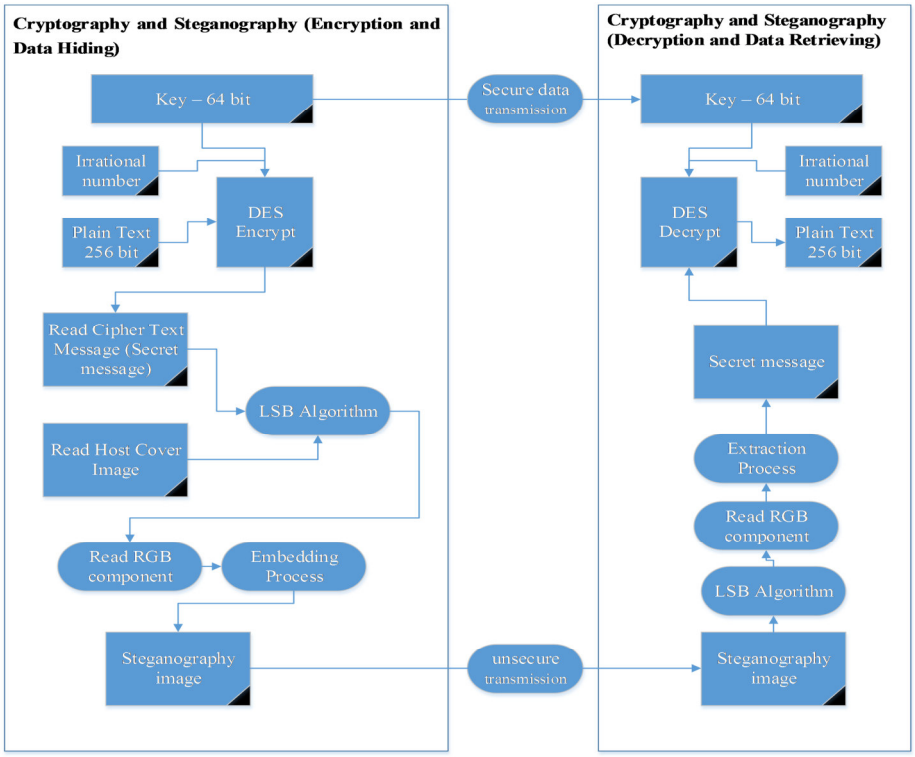


Figure 11. Overall Workbench for the Proposed Algorithm

The proposed algorithm has been implemented by using three interfaces as shown in Figures 12, 13 and 14 below.

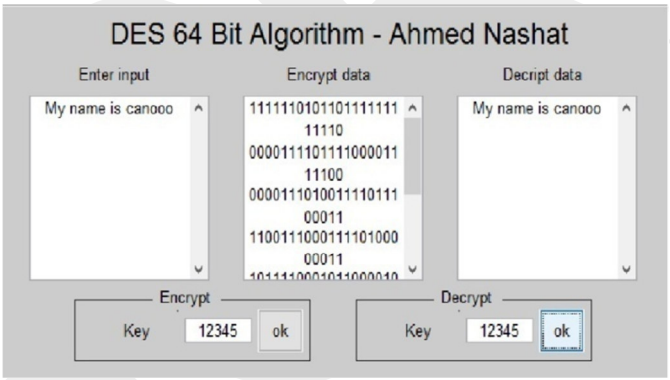


Figure 12. DES Algorithm Interface

Program code windows above show, our main DES application, include following box windows, first input (the text message along with key box entry), second is encryption data box (text message after DES apply) , third is decrypt data box (present the original text message after decrypt DES apply along with key box).

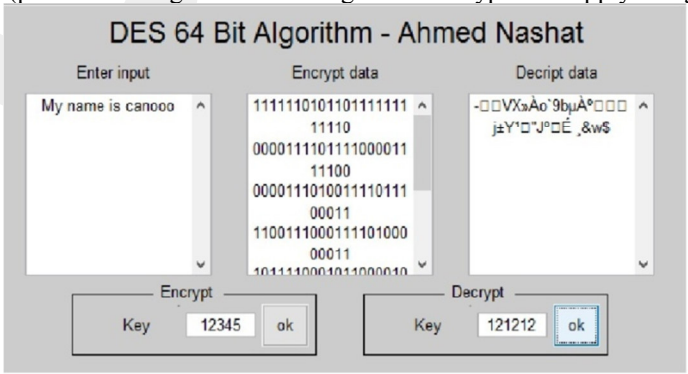


Figure 13. Examination Placebo Key

Program code windows above show, our main DES application, when try to enter incorrect key in decrypt key box, her we use key (12345678), while we inter incorrect key for decrypt process (12121), the result

show we cannot read the encrypt message without having the right key.

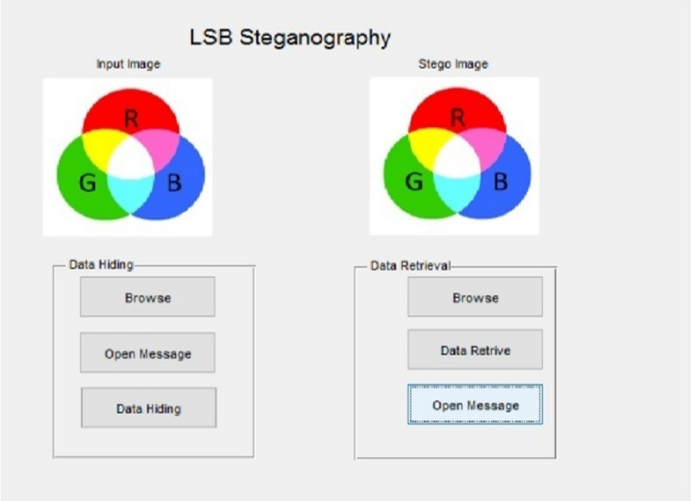


Figure 14. LSB Algorithm Interface

Program interface above show our LSB algorithm interface, include two part , the first part is input image, her we should select host image and encrypted message come from applying DES algorithm and finally applying our LSB algorithm. While the second part is concerning with message data retrieve, here we should select our steganography image and applying invers LSB algorithm to get our data retrieved.

4.2 Performance Test

In our proposed algorithm, the initial input is plain text and our final output is a steganography image. In this case, we should use many concepts to test our proposed algorithm, such as testing the performance of the improved DES algorithm by comparing it with normal DES implantation, Peak Signal-to-Noise Ratio (PSNR), Signal-to-Noise Ratio (SNR), and Mean Square Error (MSE) used to test the quality of the steganography image. Table 1 shows the comparison made between the time implementation of the normal DES algorithm and the improved DES algorithm for both the encryption and decryption processes. The results were obtained by applying (T) mutable time implementations in the same plain text and with the same key for each initial round.

Table 1. DES encryption and decryption performance test

Encryption in Second time	Encryption in Second time		Decryption in Second time	Decryption in Second Time
DES	Improved DES		DES	Improved DES
T	0.7809	0.7805	0.7784	0.7779
T	0.7730	0.7677	0.7323	0.7329
T	0.7730	0.7073	0.7820	0.7817

The Peak Signal-to-Noise Ratio (PSNR), Signal-to-Noise Ratio (SNR), and Mean Square Error (MSE) have been used to test the quality of the image. A high quality image should go over more than 30 dB [12 and 13]; this means a higher value for PSNR and SNR means higher quality for the image and quality algorithm, as we assume we have host image “I (i, j)”, and a steganography image “ IS (i, j)”, where “i” and “j” are the dimensions of the image. Then the PSNR, SNR and MSE can be calculated from the following:

PSNR = 10 log10 (Isignal 2/ MSE) …..…….. (7)

SNR = 10 log10 (Isignal / Inoise) ……..…… ..(8)

In addition, the mean square error (MSE) can be calculated from the following:

MSE = 1/I*J ((IS-I) 2) ……..… (9)

Table 2 shows the PSNR and SNR test value obtained from different test images (I), i.e., after applying the above equation to calculate the differences between the host image and the final steganography image.

Table 2. PSNR and SNR (differences between the host image and the steganography image)

No.	Cover Image	Secret Message	Steganography Image	SNR(dB)	PSNR(dB)	MSE
I1	RBG	Text	RBG pixel	46.01	47.34	0.045
I2	RBG	Text	RBG circle	46.67	48.99	0.027
I3	RBG	Text	Microsoft Logo	45.98	46.02	0.023

Table 3 shows the visual image test for both the host image and the steganography image.

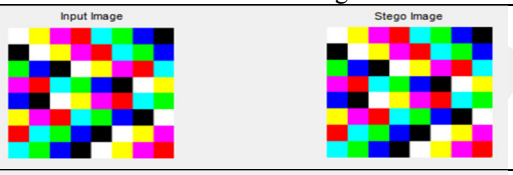
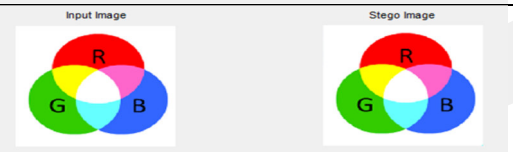
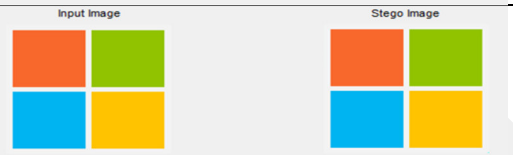
Image	Result Image
RBG pixel	
RBG circle	
Microsoft Logo	

Table 3. Visual image test

5. Conclusions

We have discussed the most important types and methods of encryption techniques used in the present day in addition to effective steganography techniques. Peak Signal-to-Noise Ratio (PSNR), Signal-to-Noise Ratio (SNR), and Mean Square Error (MSE) have been used to test the quality of steganography images in addition to visual test images. Our test lab showed the following results: the average time in the implementation for both the encryption and decryption process of the improved DES algorithm is better than the implementation of the DES. In addition, the suggestion of using an irrational number along with the DES algorithm provided us with improved security and advanced encrypting efficiency. In addition, it extended the key space without any extra running time, which can be considered to be promising in the field of information and communication technology today. Although the Least Significant Bit (LSB) steganography algorithm uses a part of the host cover image information which is changed slightly in an attempt to conceal information inside it, our visual image lab results show that both images (i.e. host image and steganography image) cannot be visually differentiated. In addition, the integration between the improved Data Encryption Standard (DES) and the Least Significant Bit (LSB) steganography algorithm provides us with two layers of protection, the first of which is the encryption with an improved DES algorithm and the second being the hiding of this cipher text in multiple locations in the cover host image. Therefore, even if an attacker succeeds in finding hidden information, he will find encrypted information which cannot be deciphered without the use of a key. In addition, in the reverse situation, if an attacker was to steal, or successfully sniff, the secret key, he would not be able to know the location of the message due the use of the Least Significant Bit (LSB) steganography algorithm.

References

[1] Robling Denning, D. E. (1982). Cryptography and data security: Addison-Wesley Longman Publishing Co., Inc.
[2] Lu, C.-S. (2004). Multimedia Security: Steganography and Digital Watermarking Techniques for Protection of Intellectual Property: Steganography and Digital Watermarking Techniques for Protection of Intellectual Property: Igi Global.
[3] Ibrahim, R., &Kuan, T. S. (2011). Steganography algorithm to hide secret message inside an image. arXiv preprint arXiv:1112.2809.
[4] Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A. (1996). Handbook of applied cryptography: CRC press.
[5] Grabbe, J. O. (1992). The DES algorithm illustrated. Laissez Faire City Times, 2(28), 12-15.

[6] Russell, D., &Gangemi, G. (1991). Computer security basics: " O'Reilly Media, Inc.".
[7] Al-Dmour, H., & Al-Ani, A. (2016). A steganography embedding method based on edge identification and XOR coding. Expert Systems with Applications, 46, 293-306.
[8] Champakamala .B.S, Padmini.K, Radhika .D. K Asst Professors, "Least Significant Bit algorithm for image steganography" , Department of TCE, Don Bosco Institute of Technology,Bangalore, India.
[9] Zomaya, A. Y., Seredynski, F., &Bouvry, P. (2003, 14-18 July 2003). Secret key cryptography with cellular automata. Paper presented at the Computer Systems and Applications, 2003. Book of Abstracts. ACS/IEEE International Conference on
[10] Lyu, S., &Farid, H. (2006). Steganalysis using higher-order image statistics. Information Forensics and Security, IEEE Transactions on, 1(1), 111-119.
[11] Narayana, S., & Prasad, G. (2010). Two new approaches for secured image steganography using cryptographic techniques and type conversions. Signal & Image Processing: An International Journal (SIPIJ) Vol, 1.
[12] Ammar Jameel Hussein , SedaYuksel, ErsinElbasi, " Dynamic Binary Location based Multi-watermark Embedding Algorithm in DWT " (Improved) , Journal of Theoretical and Applied Information Technology 20th August 2015 -- Vol. 78. No. 2 – 2015.
[13] Ammar Jameel Hussein,Et al.," SVD and DWT Techniques for Copyright Protection", 3rd Global Conference on Computer Science, Software, Istanbul, Turkey, November 2015.

Ahmed Nashaat SHAKIR received the B.S. degree in Software Engineering from the technical college 2005 Kirkuk, Iraq and the M.S. degrees in Computer Engineering from Çankaya University 2016 Turkey. Since 2007 he has been working for the faculty of engineering of Kirkuk University. His research interests include Steganography, Cryptography and Data Hiding.
Sibel TARIYAN ÖZYER received her BSc. (2004) and MSc.(2007) degrees in Computer Engineering from Cankaya University and PhD. (2012) degree in Modeling and Design of Engineering Systems from Atilim University. She is currently teaching at the Computer Engineering Department of Cankaya University. Her research interests are computer networks, wireless sensor networks and data communications.