ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2019.0291

SECURE TRANSMISSION OF DATA USING IMAGE STEGANOGRAPHY

Sourabh Chandra[1] and Smita Paira[2]
[1]*Department of Computer Science and Engineering, Calcutta Institute of Technology, India*
[2]*Department of Computer Science and Technology, Indian Institute of Engineering Science and Technology, India*

Abstract

Data is one of the most relevant and important term from the ancient Greek age to modern science and business. The amount of data and use of data transformation for organizational work is increasing. So, for the sake of security and to avoid data loss and unauthorized access of data we have designed an image Steganographic algorithm implementing both Cryptography and Steganography. This algorithm imposed a cipher text within a cover image to conceal the existence of the cipher text and the stego-image is transferred from sender to intended receiver by invoking a distributed connection among them to achieve the data authenticity.

Keyword:

Cryptography, Steganography, RSA, RMI Architecture, Distributed connection, JPEG image

1. INTRODUCTION

As a part of information security "Steganography" is a well-known concept, literally which signifies the meaning "covered writing" [12]. Steganography imposes the secret information within a cover object termed as stego-medium to escape detection and to retain the original information with minimum distortion. This stego-medium appears like a non-secret file in the network and manages to avoid drawing the attention towards itself as a content of security.

$$\textit{secret-information} + \textit{cover-medium} = \textit{stego-medium} \quad (1)$$

Steganography had been widely used for secure communication [3]. The schemes used at this age are the physical process of Steganography. In modern digital steganography information is first encrypted. Then using an embedding algorithm in the transport layer encrypted information is embedded with the cover medium and transmitted over the network [10] [11]. Both cryptography and steganography provide data confidentiality and authenticity. In contrast to cryptography which focuses on keeping the message secret while the existence of secret message may tempt the attacker whereas Steganography hides a message as well as the very existence of secret information [5]. Cryptography ensures privacy of message and structure of the message alter whereas steganography ensures the secrecy of message and the structure of message does not alter [7] [3]. Steganography may use in conjunction with cryptography by concealing the existence of the ciphered text so that the information is more secure [4].

Media formats .JPEG, .BMP, .GIF, .MP3, .text etc. are suitable as cover medium because of their high degree of redundancy and availability and popularity over internet [6]. Depending on what type of cover-medium used, steganography is classified as audio steganography uses .WAV, .MP3 media formats, video steganography uses .MPEG, .AVI, image steganography uses .JPEG, .BMP, .GIF media formats. Audio steganography utilize the Psycho acoustical property of human auditory system (i.e. the presence of low-pitched sound is undetected in presence of a louder sound) and inserting data into digitalized audio-signals. LSB coding, phase coding, spread spectrum are some popular method of audio steganography. Video Steganography embedded the message within the video files. Due to its large size video Steganography is eligible to hide large amount of data. Image steganography technique utilize the weakness of human visual system [8] and embedded the information with a minor modification in image pixels. LSB coding, masking and filtering etc. are the image steganography method.

2. LITERATURE REVIEW

In [13], Guo and Le measured the quality factor of JPEG images by maintaining quantization tables and performed some permutations along with this scheme to transmit a hidden file. Authors in [14] combined cryptography with steganography by first encrypting a message using Vernam cipher and then embedding it with an image using LSB technique with shifting. Seethalakshmi et al. [15] implemented neural networks to identify best locations in the host image to embed the secret data. Patel and Meena superimposed dynamic cryptography with steganalysis [16]. The LSB of the picture element is modified with the MSB of it and pixel selection is done using pseudo random number.

Abood combined image cryptography with steganography [17]. Both encryption and decryption are done using RC4 stream cipher and a hash function along with RGB pixel shuffling are used for steganalysis. The authors in [18] proposed two quantum image hiding strategies. A steganography quantum approach is proposed to hide an image in another image file. Secondly, a quantum watermarking approach is used to hide a water-marked gray image to a carrier image. Qo et al. proposed a quantum steganography approach using matrix coding for quantum color images [19].

3. CONTRIBUTION

To make a secure and imperceptible transaction of information, we have proposed a technique to hide the encrypted text within an image by combining the cryptography and steganography process. In our proposed method we first encrypt the plain text using public key cryptography algorithm RSA of 50bit. Then choose a JPEG image as cover and read the header-footer of this image in an array buffer. We generate the Stego-image by including the cipher text at the end of the footer.

3.1 METHODS USED

Among various methods of Cryptography and Steganography we have used RSA algorithm and Image Steganography method.

3.1.1 RSA Algorithm:

Rivest et al. invented RSA [1] algorithm and it is widely used public key cryptography algorithm having two algebraic structures: a public key $R = Z_n + X$ and a private group $G = Z(\emptyset(n))^*X$. In RSA algorithm two prime numbers (p and q) are taken initially and their products are used to generate the public key and private key. Public key consists of a value n and e, called modulus and public exponent respectively. Private key termed as d is called private exponent. The public key and private key generation of RSA algorithm is as follows:

Step 1: Choose two large, random prime number p and q, such that $p \neq q$.

Step 2: Compute modulus n as $n = p \times q$.

Step 3: Compute the Euler's totient for n as $Ø(n) = (p\text{-}1)\times(q\text{-}1)$.

Step 4: Select the public exponent e, where $1 < e < Ø(n)$ and e is a co-prime of $Ø(n)$.

Step 5: Calculate the private exponent d as $d = e^{-1} \text{mod} Ø(n)$.

The encryption operation in RSA for message P is done by the exponentiation to the eth power modulo n:

$$C = P^e \ \text{mod}n \tag{2}$$

Decryption of ciphered text C is the exponentiation to the d^{th} power modulo n:

$$P = C^d \ \text{mod}n \tag{3}$$

Explanation with example:

1. Choose two prime numbers, $p = 61$ and $q=53$
2. Compute modulus $n = pq$, where $n = 61\times53 = 3233$
3. Compute Euler's totient $Ø(n) = (p\text{-}1)\times(q\text{-}1)$, $Ø(n) = 3120$
4. Choose e co-prime to $Ø(n)$ where $1< e < Ø(n)$, $e = 17$
5. Compute $d = e^{-1}\text{mod}Ø(n)$, $d = 17\text{-}1\text{mod}(3120) = 2753$

Let the message to encrypt $P = 123$, we calculate $C = 12317$ mod(3233) = 855 and to decrypt $C = 855$, we calculate $P = 8552753\text{mod}(3233) = 123$

3.1.2 Image Steganography:

Image Steganography uses digital image as cover object to hide the information as it contains huge amount of redundant bits [9]. Images are divides into a matrix of pixels and each pixel is represented by bit pattern. Generally images are represented by 24bit or 8bit pixel either in the form of a binary file where each pixel represents three colors Red, Green and Blue (RGB) for color image or as a gray scale image [9]. In image steganography information is hidden within an image with a minor modification in the image pixels. In the specific domain of digital image various image formats exist for different applications such as JPEG (Joint Photographic Experts Group), GIF (Graphical Interchange Format), BMP etc. [3].

In this paper, we have used a .jpeg file as a cover medium. The original text is first encrypted using the well-known RSA cryptography algorithm. The generated cipher text is hidden behind the .jpeg image file for secure transmission of the data to the receiver. The encrypted text is appended at the end of the image file and finally the stego file is exposed through an RMI architecture for different Client-Server actions.

3.1.3 RMI Architecture:

Remote Method Invocation [2] allow programmers to develop a distributed object application. RMI creates a Client-Server relationship. The Server application creates an object and makes it accessible remotely. A Client application receives a reference to the object on the server and then invokes the method on it. The interface uses by the Client and Server objects to create a remote connection is provided through stubs and skeleton, remote reference and transport layer protocol. A simple Client/Server application is created by using the following step.

Step 1: Enter and Compile the Source Code: This application uses four source files- first file, which defines the remote interface that is provided by the server, second source file implements the remote interface, third one contains the main program for the server machine and fourth one implements the client side of this distributed application.

Step 2: Generate Stubs and Skeleton: Stubs reside on the client machine to present the same interfaces as the remote server.

Step 3: Install files on Client and Server

Step 4: Start the RMI registry on server machine

Step 5: Start the server

Step 6: Start the Client

3.2 PROPOSED ALGORITHM

This section presents a step-by-step solution to the problem described above. The encryption algorithm at the Sender's end and decryption algorithm at the Receiver's end are detailed below.

3.2.1 Encryption Algorithm (Sender's end):

Step 1: Select the text file where the original message has been written.

Step 2: Encrypt the content of the text file using the RSA algorithm with the public key of the receiver.

Step 3: Select an appropriate cover image (.jpeg format).

Step 4: Read the header and footer of the selected image in an array buffer.

Step 5: Add the encrypted data at the end of image footer.

Step 6: Sender and receiver are connected to the network.

Step 7: Sender provides the receiver's IP address and then send the Stego-image if the IP address is valid.

3.2.2 Decryption Algorithm (Receiver's end):

Step 1: Receive the Stego-image.

Step 2: Extract the encrypted message from the end of the stego-image by reading the image footer.

Step 3: Generate the private key and decrypt the extracted message and then create a text file.

Step 4: Save the text file at the desired location.

3.2.3 Key Used:

The privacy of any cryptography or steganography algorithm depends on the size of the key used. In the proposed algorithm, we have used the RSA algorithm for encrypting the text data. In RSA, two large random prime numbers are generated and are processed to create the private and public keys. These prime numbers need to be kept secret. The length of the RSA key depends on the number of bits used in the modulus function. In this algorithm, we have used keys of size 2048 bits.

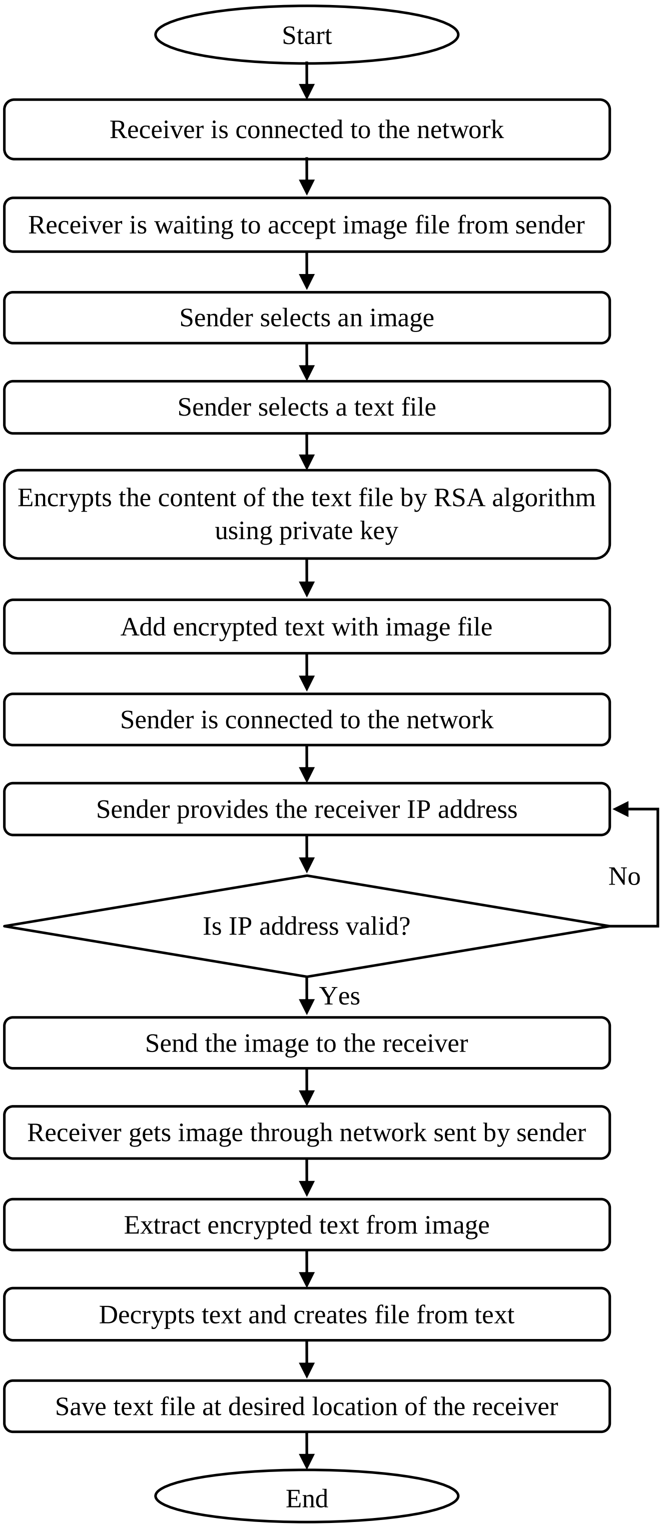

Fig.1. Flowchart of proposed algorithm

The flow diagram (shown in Fig.1) of the proposed work has been explained in this section. Initially receiver asks the sender for connection establishment over the internet. Once they are connected, the sender starts the file transfer process. For secure transmission of the text file, sender applies an integrated approach by combining cryptography with image steganography.

The sender first encrypts the text file using RSA cryptography algorithm and then hides the cipher text behind an image for secure propagation. After validation of the receiver's IP address, the image stego file is sent to the receiver. The latter upon reception extracts the encrypted text message from the image file and decrypts it to get the original message.

3.3 DATA FLOW DIAGRAM

This section illustrates the proposed technique through data flow diagrams. The Fig.2 and Fig.3 presents the context level and first level of the data flow diagram for the proposed strategy. In the context level, entire method is covered by a single process leveled 0. It performs five major operations-encryption of the original text file, hiding the cipher text behind an image file, transmission and reception of the stego file, extraction of the encrypted text from the image file at the receiver's end and finally decryption of the cipher text to get the original message.

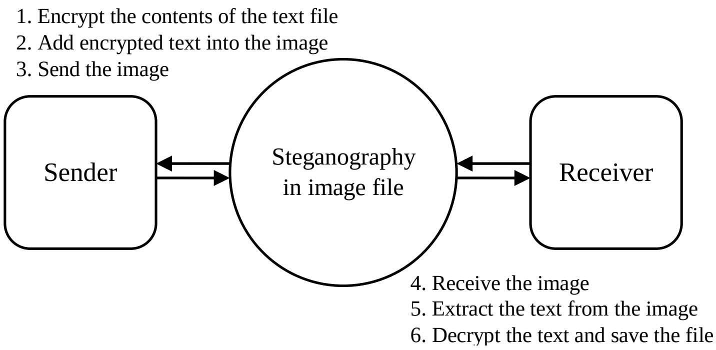

Fig.2. Context Level DFD

The Fig.3 presents the first level of the data flow diagram where the five major operations performed by the steganography process in Level 0 are shown.

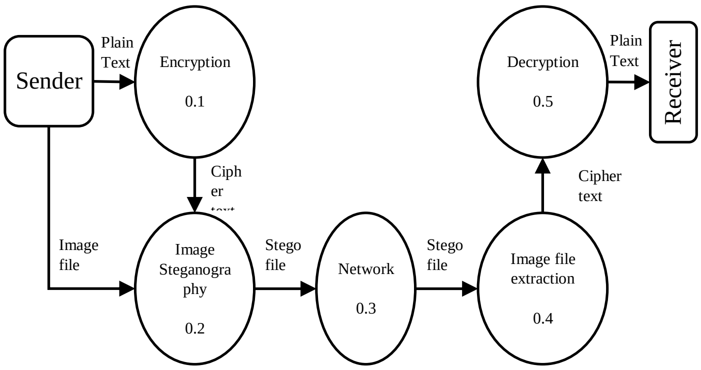

Fig.3. 1st Level DFD

3.4 ILLUSTRATION OF PROPOSED METHOD

The proposed method involves the process of encrypt the text file into an image, extraction of original text from stego image and final output.

3.4.1 Process of Encrypting the Text File into an Image:

As we stated to enhance the security level at first we have encrypted the plain text then we apply the Steganography method on this encrypted text. First we take a .txt or .doc file as plain text and read it. Then encrypt this plain text using RSA algorithm of key size 50 bits. We have chosen JPEG image as cover image. Then read the header and footer of this image and stored into an array buffer. We have added the ciphered text at the end of footer of the selected cover image and generated the stego image. The stego image is then sent to the intended receiver through a distributed connection between them, which we have created using the RMI architecture.

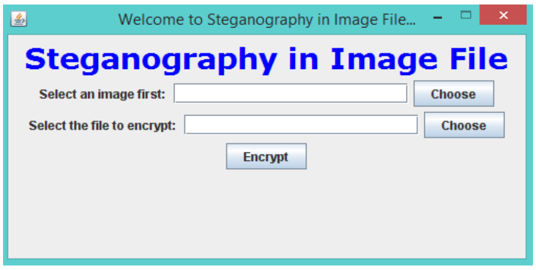

Fig.4. Interface for the Sender

The Fig.4 presents the sender's interface through which he/she selects the text file to be sent and the image file for embedding the encrypted text file. The sender then selects "Encrypt" button to start the crypto-stego integrated approach.

3.4.2 Extracting Original-Text from Stego-Image:

After receiving the Stego image, receiver first extract the Cover Image and Cipher Text. Then de-cipher cipher text and create a text file which contains the plain text. Then save the text file to its preferable location. The Fig.5 presents the interface for the Receiver where upon selection of the "Extract and Decrypt" button the entire process of cipher text decryption and original message extraction is done.

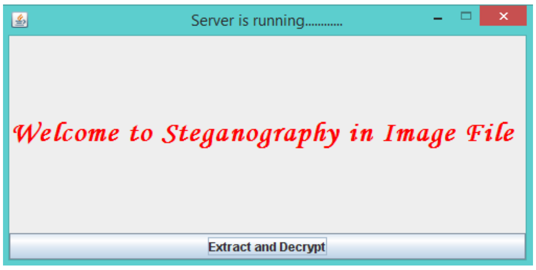

Fig.5. Interface for the Receiver

3.4.3 Final Output:

The Fig.6 shows the original image we have used as cover image to hide the text and Fig.7 shows the same image after imposing the cipher text within it. Interestingly, both original and stego images are indistinguishable. Camouflaging an encrypted pattern to an exactly natural image is intended for fooling eavesdroppers. The quality factors of the original image is managed by embedding each pixel of the stego image to a specific region.

Fig.6. Original Image

Fig.7. Stego Image

4. CONCLUSION AND FUTURE SCOPE

At this age of civilization exchanging data for communication through the network is an integral part of every organization and every sector of society. Our proposed algorithm is to secure this communication with a secure communication system by creating a distributed connection. This algorithm imposed an encrypted text which has been encrypted by using the RSA algorithm within a JPEG image and then the image file is send over the network i.e. we combining the concept of Cryptography and Steganography to make an illusion to the hacker that the sender sends an unsuspicious media file to the receiver. As an image file appear in the network as an innocent media file so it does not attract the hacker as a content of security. In this algorithm Cryptography makes the data secure as a cipher text and Steganography makes this cipher text disguise so that no one other than the intended receiver can know the existence of the cipher text within the image file. Here we have embedded the encrypted text at the footer of the chosen JPEG image file. In future our work will be focused on to embed the text within the image pixel and we will work to eradicate the problem of lossy compression related to the JPEG image and extend our work on other image formats like .BMP, .GIF etc..

REFERENCES

[1] Behrouz A. Forouzan, "*Cryptography and Network Security*", McGraw Hill, 2007.

[2] Herbert Schidlt, "*Java the Complete Reference*", 8th Edition, McGraw Hill, 2011.

[3] N.F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen", *Computer*, Vol. 31, No. 2, pp. 26-34, 1998.

[4] S. Song, J. Zhang, X. Liao, J. Du and Q. Wen, "A Novel Secure Communication Protocol Combining Steganography and Cryptography", *Procedia Engineering*, Vol. 15, pp. 2767-2772, 2011.

[5] A.J. Raphael and V. Sundaram, "Cryptography and Steganography-A Survey", *International Journal of Computer Technology and Applications*, Vol. 2, No. 3, pp. 626-630, 2016.

[6] S.A. Laskar and K. Hemachandran, "An Analysis of Steganography and Steganalysis Techniques", *Assam University Journal of Science and Technology*, Vol. 9, No. 2, pp. 83-103, 2012.

[7] F.A.P. Petitcolas, R.J. Anderson and M.G. Kuhn, "Information Hiding-A Survey", *Proceedings of International Conference on Protection of Multimedia*, pp. 1062-1078, 1999.

[8] Y.S. Huang, Y.P. Huang, K.N. Huang and M.S. Young, "The Assessment System of Human Visual Spectral Sensitivity Curve by Frequency Modulated Light", *Proceedings of IEEE International Conference on Engineering in Medicine and Biology*, pp. 263-265, 2005.

[9] T. Morkel, J.H.P. Eloff and M.S. Oliver, "An Overview of Image Steganography", *Proceedings of 5th Annual Conference on Information Security*, pp. 111-116, 2005.

[10] T. Handel and M. Sandford, "Hiding Data in the OSI Network Model", *Proceedings of 1st International Workshop on Information Hiding*, pp. 1-7, 1996

[11] R. Doshi, P. Jain and L. Gupta, "Steganography and its Applications in Security", *International Journal of Modern Engineering Research*, Vol. 2, No. 6, pp. 4634-4638, 2012.

[12] M. Rouse, "Steganography", Available at: https://searchsecurity.techtarget.com/definition/steganography

[13] Jing-Ming Guo and Thanh Nam Le, "Secret Communication using JPEG Double Compression", *IEEE Signal Processing Letters*, Vol. 17, No. 10, pp. 879-882, 2010.

[14] Kamaldeep Joshi and Rajkumar Yadav, "A New LSB-S Image Steganography Method Blend with Cryptography for Secret Communication", *Proceedings of* 3rd *International Conference on Image Processing*, pp. 86-90, 2015.

[15] K.S. Seethalakshmi, B.A. Usha and K.N. Sangeetha, "Security Enhancement in Image Steganography using Neural Networks and Visual Cryptography", *Proceedings of International Conference on Computational Systems and Information Systems for Sustainable Solutions*, pp. 396-403, 2016.

[16] Nikhil Patel and Shweta Meena, "LSB Based Image Steganography using Dynamic Key Cryptography", *Proceedings of International Conference on Emerging Trends in Communication Technologies*, pp. 448-457, 2016.

[17] May H. Abood, "An Efficient Image Cryptography using Hash-LSB Steganography with RC4 and Pixel Shuffling Encryption Algorithms", *Proceedings of Annual Conference on New Trends in Information and Communications Technology Applications*, pp. 86-90, 2017.

[18] A.A.A. El-Latif, B. Abd El Atty, M.S. Hossain, M.D. A. Rahman, A. Alamri and B.B. Gupta, "Efficient Quantum Information Hiding for Remote Medical Image Sharing", *IEEE Access*, Vol. 6, pp. 21075-21083, 2018.

[19] Z. Qu, Z. Cheng and X. Wang, "Matrix Coding-Based Quantum Image Steganography Algorithm", *IEEE Access*, Vol. 7, pp. 35684-35698, 2019.