Research Journal of Information Technology

ISSN 1815-7432

Academic Journals Inc.

www.academicjournals.com

Research Journal of Information Technology 5 (3): 456-461, 2013
ISSN 1815-7432 / DOI: 10.3923/rjit.2013.456.461

An Efficient Steganographic Method by using Image Fragments With High Security

B. Elangovan and S. Mohana
School of Computing, SASTRA University, Thanjavur, Tamilnadu, India

Corresponding Author: B. Elangovan, School of Computing, SASTRA University, Thanjavur, Tamilnadu, India

ABSTRACT

Picture quality and undetectability are the key aspects of steganography. In this study, the proposed framework uses a novel approach to enhance the steganographic scheme with optimized picture quality and higher anti-steganalysis capability. To achieve this, the secret image is hidden in the carrier image by creating layers in Stegoimage. Differing from previous works, this method retains the quality of the carrier image and it doesn't depend on the size of the carrier image and shows better imperceptibility. To abate to the damage in the reconstructed carrier image, the secret image is divided into two segments, one with odd bytes and the other with even bytes. These are embedded into the carrier image in the form of layers rather than embedding it on the carrier image itself. A pass-key is used for security purpose. This is done after pointing out the related approaches and then highlighting this model's contributions with respect to the embedding and extraction processes.

Key words: Image steganography, image processing, information security, encryption

INTRODUCTION

A steganographic system conceals secret data in a carrier image so as not to arouse a suspicion apart from the sender and receiver. According to Stefan and Fabin (2000), Steganography is secret way of communicating by hiding the existence of communication itself. Cryptograohy is the well known cousin of steganography where the message is scrambled to prevent the message from hackers (Salem *et al*., 2011; Amirtharajan *et al*., 2011, 2012).

In the present digital world, a steganographic system keeps the presence of secret data imperceptible (Amirtharajan and Rayappan, 2012a-d; Cheddad *et al*., 2010; Janakiraman *et al*., 2012a, b). But steganographic system tends to be detectable in the cover medium. Even if secret data is not detected, the existence of it modifies the carrier image's statistical properties. This leads to eavesdroppers to easily identify the changes in the properties of Stegoimage (Amirtharajan *et al*., 2012).

This study introduces a new steganographic system of hiding the secret data with higher anti-steganalysis capability by adding the fragmented data files of secret image as layers to the carrier image.

RELATED WORK

In steganography, embedding the secret bits in the redundant bits of the stego object and transmits as a carrier image (Karthikeyan *et al*., 2012; Padmaa *et al*., 2011; Rajagopalan *et al*.,

2012; Thanikaiselvan *et al.*, 2011; Thenmozhi *et al.*, 2012). Presently, most of the steganographic processes depend on the size of the carrier image. Thus a large sized secret data cannot be embedded in a carrier image of smaller size. Moreover there are chances for the carrier image to overflow when certain crypto algorithms are used during encryption. In such cases, a part of the secret image will be lost.

Considering these as the barriers in steganographic system this proposed model overcomes these difficulties.

PROPOSED WORK

As told earlier this idea mainly focuses on undetectability. So this framework begins by splitting the secret image into two segments of odd and even bytes. These are embedded into the carrier image in layers sequentially along with the encrypted text. This forms the Stegoimage. This is sent to the receiver. The receiver first isolates the carrier image and the hidden data (two segments of hidden image and encrypted text). Then the receiver uses the pass-key and retrieves the original hidden image and plain text from the encrypted text. The detailed implementation of this idea is as follows.

HIDING PROCESS

The sender first loads the secret image. This image is divided into two segments. The first segment consists of all the odd bytes and the second segment consists of all the even bytes. These are stored as separate data files. Then the binary data of the carrier image that is selected is followed by the sentinel string which is used to distinguish the carrier image from the rest of the layers. Then the above two data files are embedded sequentially with pass-key as the delimiter between them. This pass-key is also used to encrypt the text using Data Encryption Standard. This encrypted text is also embedded into the carrier image. Thus the Stegoimage consists of carrier image followed by a sentinel string which separates the carrier image from rest of the layers. This sentinel string is followed by the odd bytes data file, the even bytes data file and the encrypted text being embedded sequentially in layers with pass-key as the delimiter. This Stegoimage is sent to the receiver. The implementation of hiding process is presented in the form of an algorithm as follows.

Hiding algorithm:

Input: A secret image S_i to be hidden, where $_i$ corresponds to the number of bytes. Carrier image C, text to be hidden T, pass-key for encryption P_k.

Output: Stegoimage C'

Step 1: Load the secret image S_i and split into two segments with odd bytes in odd data file $O_j(S)$ and even bytes in even data file $E_k(S)$

Step 2: Load the carrier image C, odd bytes data file ($O_j(S)$) and even bytes data file ($E_k(S)$). The binary data of carrier image is followed by the sentinel string

Step 3: Load the text to be hidden T

Step 4: Get the P_k from sender with a minimum of eight characters

Step 5: Embed $O_j(S)$ after the sentinel string. This is followed by embedding P_k and $E_k(S)$

Step 6: The text to be hidden T is encrypted by using DES with pass-key. Then the encrypted text is added after $E_k(S)$ with P_k as the delimiter

This forms the Stegoimage C' and this is sent to the receiver.

RETRIEVING PROCESS

The receiver loads the Stegoimage after which the receiver isolates the carrier image and the hidden data (secret image and encrypted text) with sentinel string as the delimiter. From this hidden data the receiver extracts the (1) odd bytes data (2) even bytes data (3) encrypted text separately by using pass-key as the delimiter. The retrieving process is carried out only if the receiver's pass-key matches with the pass-key in the Stegoimage. Then the hidden image is formed by merging the odd byte file and even byte file alternatively. Finally the encrypted text is decrypted by using Data Encryption Standard algorithm with pass-key.

Retrieving algorithm

Input: Stegoimage C', Pass-key P_k for decryption

Output: Plain text T from encrypted text, Secret image S_i

Step 1: Load the received Stegoimage C'

Step 2: Separate C and hidden data with sentinel string as delimiter

Step 3: From the hidden data the odd bytes ($O_j(S)$), even bytes ($E_k(S)$) and encrypted text are extracted

Step 4: The bytes from $O_j(S)$ and $E_k(S)$ are merged alternatively to produce the original secret image S_i

Step 5: Using P_k decrypt the encrypted text by using DES

ALGORITHM IMPLEMENTATION

As an implementation of the hiding process, Fig. 1 acts as the secret image which is segmented into two data files ($O_j(S)$-odd bytes and $E_k(S)$-even bytes). Figure 2 acts as the carrier image (C) which will be loaded with two data files $O_j(S)$ and $E_k(S)$ and the encrypted form of text to be hidden using pass-key. This process produce the StegoImage C'.

In Fig. 3, the process of hiding secret image and the text with Carrier Image is shown.

The retrieving process loads the Stegoimage C' and checks the pass-key with the hidden data, if it matches, it extracts the secret image (S_i) and the decrypted text.

In Fig. 4 the process of extracting the secret image (Fig. 5) and the hidden text with the use of correct pass-key (P_k) is shown.

Fig. 1: Sample secret image (S_i) to be segmented into two data files

Fig. 2: Sample carrier image (C) which will be loaded with two data files

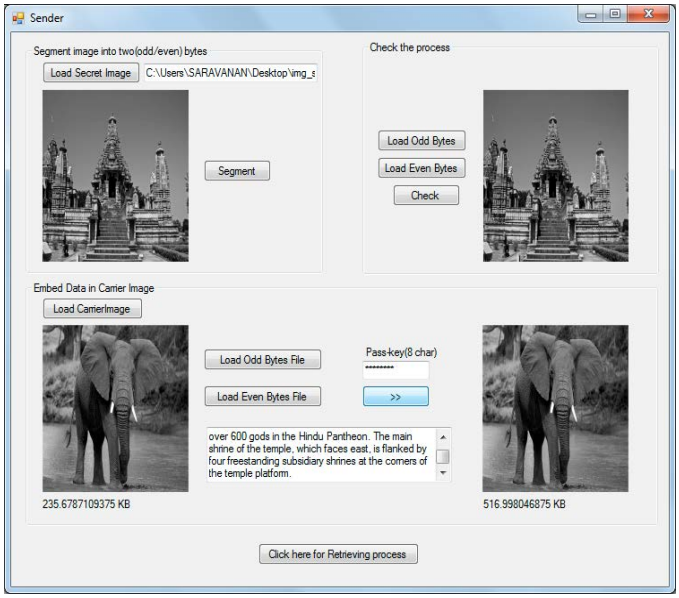

Fig. 3: Hiding secret image and Text in a carrier image

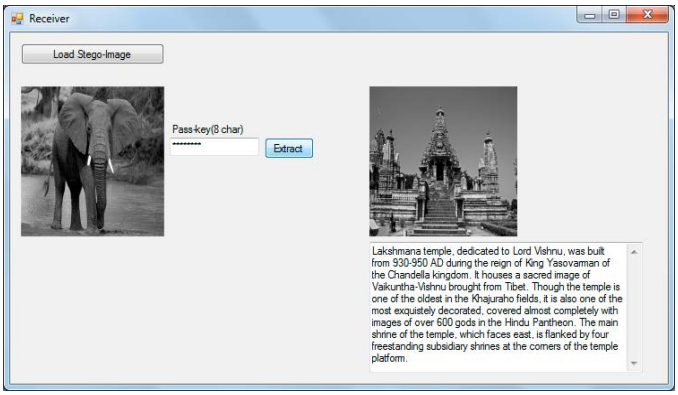

Fig. 4: Retrieving secret image and text from carrier image

Fig. 5: Retrieved secret image from the carrier image

CONCLUSION

This study introduces a new steganographic system of hiding the secret data with high security. This is ensured by dividing the secret image into two segments. Moreover, this method does not depend on the size of carrier image. Thus a secret data of any size can be embedded into the carrier image in layers without affecting the statistical properties of carrier image. This method can be further extended to hide multimedia data.

REFERENCES

Amirtharajan, R., R.R. Subrahmanyam, P.J.S. Prabhakar, R. Kavitha and J.B.B. Rayappan, 2011. MSB over hides LSB: A dark communication with integrity. Proceedings of the IEEE 5th International Conference on Internet Multimedia Systems Architecture and Application, December 12-14, 2011, Bangalore, Karnataka, India, pp: 1-6.

Amirtharajan, R. and J.B.B. Rayappan, 2012a. An intelligent chaotic embedding approach to enhance stego-image quality. Inf. Sci., 193: 115-124.

Amirtharajan, R. and J.B.B. Rayappan, 2012b. Brownian motion of binary and gray-binary and gray bits in image for stego. J. Applied Sci., 12: 428-439.

Amirtharajan, R. and J.B.B. Rayappan, 2012c. Inverted pattern in inverted time domain for icon steganography. Inf. Technol. J., 11: 587-595.

Amirtharajan, R. and J.B.B. Rayappan, 2012d. Pixel authorized by pixel to trace with SFC on image to sabotage data mugger: A comparative study on PI stego. Res. J. Inf. Technol., 4: 124-139.

Amirtharajan, R., J. Qin and J.B.B. Rayappan, 2012. Random image steganography and steganalysis: Present status and future directions. Inf. Technol. J., 11: 566-576.

Cheddad, A., J. Condell, K. Curran and P.M. Kevitt, 2010. Digital image steganography: Survey and analysis of current methods. Signal Process., 90: 727-752.

Janakiraman, S., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2012a. Firmware for data security: A review. Res. J. Inf. Technol., 4: 61-72.

Janakiraman, S., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2012b. Pixel forefinger for gray in color: A layer by layer stego. Inf. Technol. J., 11: 9-19.

Karthikeyan, B., V. Vaithiyanathan, B. Thamotharan, M. Gomathymeenakshi and S. Sruti, 2012. LSB replacement stegnography in an image using pseudorandomised key generation. Res. J. Applied Sci. Eng. Technol., 4: 491-494.

Padmaa, M., Y. Venkataramani and R. Amirtharajan, 2011. Stego on 2n: 1 Platform for users and embedding. Inf. Technol. J., 10: 1896-1907.

Rajagopalan, S., R. Amirtharajan, H.N. Upadhyay and J.B.B. Rayappan, 2012. Survey and analysis of hardware cryptographic and steganographic systems on FPGA. J. Applied Sci., 12: 201-210.

Salem, Y., M. Abomhara, O.O. Khalifa, A.A. Zaidan and B.B. Zaidan, 2011. A review on multimedia communications cryptography. Res. J. Inf. Technol., 3: 146-152.

Stefan, K. and A. Fabin, 2000. Information Hiding Techniques for Steganography and Digital Watermarking. Artech House, London, UK.

Thanikaiselvan, V., S. Kumar, N. Neelima and R. Amirtharajan, 2011. Data battle on the digital field between horse cavalry and interlopers. J. Theor. Applied Inf. Technol., 29: 85-91.

Thenmozhi, K., P. Praveenkumar, R. Amirtharajan, V. Prithiviraj, R. Varadarajan and J.B.B. Rayappan, 2012. OFDM+CDMA+Stego = Secure communication: A review. Res. J. Inf. Technol., 4: 31-46.