

THE EFFECT OF STEGANOGRAPHY AND SECRET IMAGE SHARING SCHEME TO THE SECURITY OF CRYPTOGRAPHIC KEY

Derya Arda¹, Selin Demirbilek², Sermin Kavak³

¹Trakya University, Department of Computer Engineering, 22030, Edirne, Turkey

²Trakya University, Institute of Science, Computer Engineering, 22030 Edirne, Turkey

³Trakya University, Institute of Science, Computational Science, 22030, Edirne, Turkey

Abstract

Today, with the widespread use of the internet, the importance of information security has increased even more. Cryptography and steganography are the most common security methods. In this article, the steganographic method and the secret image sharing scheme have been used together to securely store the secret key used in the AES encryption algorithm. First, the secret key is embedded into an image with the LSB technique, then the stego-image is divided into share images by Thien and Lin Secret sharing scheme. It is quite difficult to reach a meaningful knowledge with any of these share images. But hidden data can be recovered with threshold-sharing images. As a result, the efficiency of both methods has been demonstrated in terms of securing the encryption key.

Key words: Image secret sharing, Steganography, LSB, Cryptology

1. INTRODUCTION

Today, with the widespread use of the internet, the importance of information security has increased even more. Various methods and protocols are used to provide information security. Cryptography and steganography are the most common security methods. In Cryptology, while the data content is changed with a key, in steganography, the contents are not changed, but the data is hidden in a cover object. So that, it is aim that the hidden message is not notice by third parties. According to the method used in cryptography, it is very important that the encryption key is kept secret. This key is either stored on a computer or on a person or in any magnetic location. With the development of encryption methods, there is a problem that the key is protected or transmitted securely. [2]In the literature, secret sharing schemes have been proposed as solutions to these problems. For the first time, the concept of secret sharing scheme has been proposed by Shamir and Blakley in 1979 (Shamir, 1979, Blakely, 1979). At the same time, Secret sharing schemes (SSS) are known as (k, n) threshold scheme. A (k, n) threshold scheme allows a secret to be shared among n participants in such a way that any k of them can recover the secret, but any $k-1$, or fewer, have absolutely no information on the secret (Shamir, 1979, Blakely, 1979).

Shamir's secret sharing method is a threshold scheme based on Lagrange's polynomial interpolation. Blakley used secret sharing as a finite geometry based on hyperplanes in k -dimensional space. Later, the remaining theorem-based threshold secret sharing schemes using the properties of number theory were found in 1983 by Mignotte (Mignotte. M., 1983) and Asmuth-Bloom (Asmuth C. A. and Bloom J., 1983). These schemes use specially selected integer arrays during the Chinese Remainder Theorem.

There has been a problem of secure transmission of data that must be kept secret on the military, commercial and medical areas, such as electronic information, images, texts and sounds shared with the use of the internet. A lot of work has been done to ensure the safety of secret images.

Visual Secret Sharing Schemes were originally proposed by Naor and Shamir in 1994 (Naor M. and Shamir A., 1995). In this scheme (k, n) , as in the threshold scheme, hidden images are applied to visual cryptography techniques to generate n meaningless shares and distributed to participants who share the secret. In order to recover the hidden image, at least k participant's share images must be overlaid. If the number of participants is less than k , no information about the hidden image can be obtained. In this technique, the contrast is lost when the original image is compared with the hidden

image during the reconstruction phase, because the size of the shares is twice the hidden image of the size.

In 2002, proposed by Thien and Lin (Thien C. C. and Lin J. C., 2006), in the secret image sharing scheme (k, n) , the hidden image is divided into n image shares among the participants. In order to reconstruct the secret image, at least k participant shares must be brought together, and if k is less participant, no secret image is obtained.

In this article, the steganographic method and the secret image sharing scheme have been used together to securely store the secret key used in the AES encryption algorithm. First, the secret key is embedded in the RGB color image with the LSB technique only on the blue channel. After that, the permutation process is done. Then the permuted stego-image is divided into share images by Thien and Lin Secret sharing scheme. It is quite difficult to reach a meaningful knowledge with any of these share images. But hidden key can be recovered with threshold-sharing images. In this work, both of these methods, which are solutions to the problem of storing the cryptographic key securely, have been used together and have been shown to be stored more securely.

2. OVERVIEW OF STEGANOGRAPHY

Steganography is the art and science of invisible communication. This is accomplished through hiding information in other information, thus hiding the existence of the communicated information. In image steganography the information is hidden exclusively in images. The intent is to transmit hidden information. Steganography system consists of three elements: cover-image (which hides the secret message), the secret message and the stegano-image (which is the cover object with message embedded inside it).

The secret message is hidden in the color bits of the cover image pixel. A colored image consists of pixels that hold three digits corresponding to the levels of red, green, and blue. Red, green, and blue colors can be added together to create all the colors. Each color is in the range of 0-255, and each color pixel consists of 24 bits (Johnson, N.F. & Jajodia, S., 1998, Owens, M., 2002).

The Steganography system which uses an image as the cover, there are several techniques to conceal information inside cover-image. The spatial domain techniques manipulate the cover-image pixel bit values to embed the secret information. The secret bits are written directly to the cover image pixel bytes. Consequently, the spatial domain techniques are simple and easy to implement. The Least Significant Bit (LSB) is one of the main techniques in spatial domain image Steganography (Johnson, N. F., Katzenbeisser, S., 2000).

2.1. Least Significant Bit (LSB) Method

Least significant bit (LSB) insertion is a common, simple approach to embedding information in a cover image. The last bit of each pixel swaps with secret message's bit. When using a 24-bit image, a bit of each of the red, green and blue colour components can be used, since they are each represented by a byte. In other words, one can store 3 bits in each pixel (Johnson, N. F., Katzenbeisser, S., 2000).

A 256×256 pixel image, can thus store a total amount of 196.608 bits of embedded data. For example, 3 pixels grid for of a 24-bit color image can be as follows

Color image pixels: 10101111 00011000 11000010 (175, 24, 194)
 10110000 00010110 11001000 (176, 22, 200)
 10110100 00011000 11000100 (180, 24, 196)

Secret data is 'a': ASCII value of 'a' is 97 = 01100001

After making changes on LSB of pixels, new pixel values are;

10101110 00011001 11000011 (174, 25, 195)

10110000 00010110 11001000 (176, 22, 200)

10110100 00011001 11000100 (180, 25, 196)

By changing the LSB of a pixel results in small changes in the intensity of the colors. These changes cannot be identify by the human eye, thus the message is successfully hidden in image (Dumitrescu, S., W. Xiaolin and Z. Wang, 2003).

3. THIEN-LIN SECRET IMAGE SHARING SCHEME

Thien and Lin proposed a (k, n) threshold-based image SSS by cleverly using Shamir's SSS to generate image shares. The essential idea is to use a polynomial function of order $(k-1)$ to construct n image shares from an $l \times l$ pixels secret image (denoted as I) as,

$$S_x(i, j) = I(ik + 1, j) + I(ik + 2, j)x + \dots + I(ik + k, j)x^{k-1} \pmod{p} \quad (1)$$

Where $0 \leq i \leq \left(\frac{l}{k}\right)$ and $1 \leq j \leq l$.

For gray level images in secret image sharing schemes, the pixel values are in the range [0-255]. The polynomial module value used in the sharing scheme is 251, which is the largest prime selected in this range. The prime importance of choosing a prime number is achieving a single solution to the original image. However, the pixel values obtained here are between [0-250]. Therefore, the values between [251-255] are lost and are shifted to 250. This causes a loss of brightness in the image. For color images, the color range is separately [0-255] for the Red-Green-Blue colors found in the RGB palette (Cimato, S., De Prisco, R., De Santis, A., 2007).

This method reduces the size of image shares to become $\frac{l}{k}$ of the size of the secret image (Thien C. C. and Lin J. C., 2006). The original image is reconstructed by the Lagrange Interpolation method by combining at least k units from the n pieces of image obtained. The Lagrange Interpolation formula is also described below (Kurosawa K., Obana S., Ogata W., 1995, Schoenmaker B., 2011)

$$h(x) = \sum_{i=1}^k y_i \prod_{\substack{j=1 \\ j \neq i}}^k \frac{x - x_j}{x_i - x_j} \pmod{p} \quad (2)$$

Thien and Lin also recommend that the image be permuted before it is split into shares. The permutation process can be performed with any key value or with various algorithms. In this way, security and resistance to attack are increased (Thien C. C. and Lin J. C., 2006).

4. IMPLEMENTED APPLICATION

In this work, 256x256 pixel size Lena.BMP format RGB image is used. The secret key to be used in AES encryption is embedded in this image by LSB technique only using the least significant bit of the blue channel. Thus, only 65.536 bits of data can be embedded in this image. In this work, the encryption secret key is taken as "trakya üniversitesi". This secret key is 152 bits of data. The pixel to be buried is selected sequentially, not by a specific algorithm. After embedding the image with the secret key LSB, the addition or multiplication permutation is performed with the selected hidden values. As shown in **Figure 2** and in algorithm 1, the image is permuted by multiplying by the key value (150,155,160). Then, this image is divided into shares with (k,n)=(2,4) threshold secret image sharing scheme. Thus, both the steganographic method and the secret image sharing method are used to securely store the secret key. At the same time, an AES encryption and decryption was performed with this secret key. The application was developed with the C # programming language with Visual Studio 2015. The implementation scheme is shown in **Figure 1**.

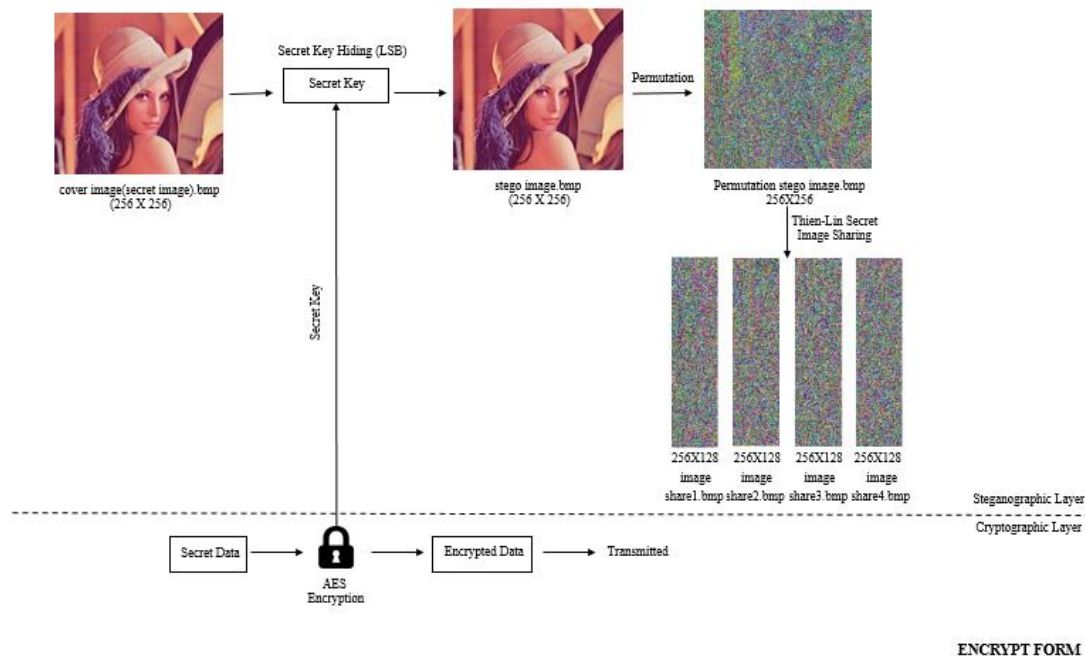


Figure 1. Secret Image Sharing, Steganographic Method and AES encryption process

The algorithm of **Figure 1** is as follows.

Algorithm 1. Stego-Image Sharing Algorithm (Steganographic Layer)

Input: A secret image I and a secret key

Output: n shadows $I_1, I_2, \dots, I_n$ and

Step 1. Convert secret key to binary

Step 2. Convert all pixel (RGB) of I image to binary

Step 3. Embed secret key in blue color with LSB

Step 4. Multiply or add permutation by a key value

Step 5. $S_x(i, j) = I(ik + 1, j) + I(ik + 2, j)x + \dots + I(ik + k, j)x^{k-1} \pmod{p}$ ($p=251$)

Adopt all coefficients of a (k-1) degree polynomial $S_x(i, j)$ to embed the secret pixels of I.

The pixel value for each shadow is calculated as $q_i = S(i)$ for $i = 1, 2, \dots, n$.

The shadows are denoted as $I_1, I_2, \dots, I_n$

Step 6. Participant i is distributed I_i for $i = 1, 2, \dots, n$

For example, let's calculate (0,0) pixel values of the parts to be sent to the receiver. From equation 1

$$S_{x=1}(0,0) = 161 + 61x \pmod{251} = 222$$

$$S_{x=2}(0,0) = 161 + 61x \pmod{251} = 32$$

$$S_{x=3}(0,0) = 161 + 61x \pmod{251} = 93$$

$$S_{x=4}(0,0) = 161 + 61x \pmod{251} = 154$$

Similarly, other colors and pixel values are calculated.

Table 1. Numerical values of obtained image

	(0,0).pixel value	(0,1).pixel value	(0,2).pixel value
Secret image	R:225 G:138 B:128	R:224 G:137 B:127	R:224 G:134 B:123
Stego-secret image	R:225 G:137 B:128	R:224 G:137 B:126	R:224 G:134 B:122
Permuteted stego-image	R:161 G:241 B:248	R:61 G:41 B:150	R:61 G:194 B:205
image share1	R:222 G:31 B:147	R:120 G:235 B:104	R:69 G:82 B:159
image share2	R:32 G:72 B:46	R:179 G:25 B:3	R:79 G:123 B:15
image share3	R:93 G:113 B:196	R:238 G:66 B:153	R:89 G:164 B:122
image share4	R:154 G:154 B:95	R:46 G:107 B:52	R:99 G:205 B:229

The steps performed in **Algorithm 1** are given in **Table 1**. The first three pixel values of the secret image, the permuted stego-secret image, the stego-secret image and the share images are given.

The screenshot of the program is given in **Figure 2**. Furthermore, the obtained values are given in **Table 1** in detail.

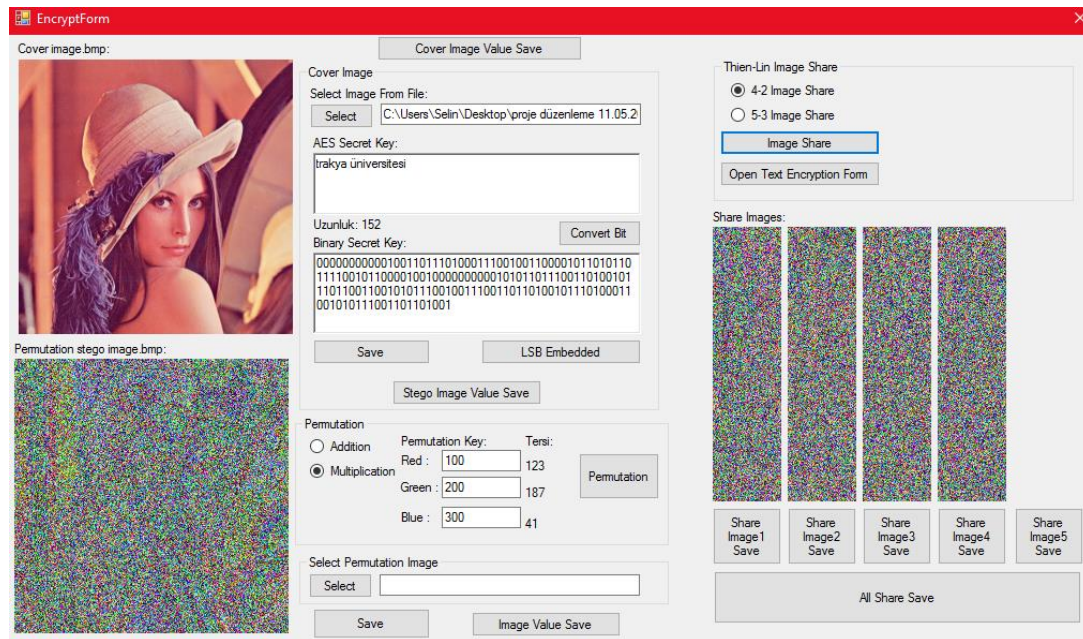


Figure 2. Display of Stego-Image Sharing Scheme

AES encryption has been implemented in **Figure 3**. The steps in **Figure 3** are described in the algorithm 2.

Algorithm 2. AES Encryption (Cryptographic Layer)

Input: Secret Data

Output: Ecryption Data

Step 1. AES encryption with secret key

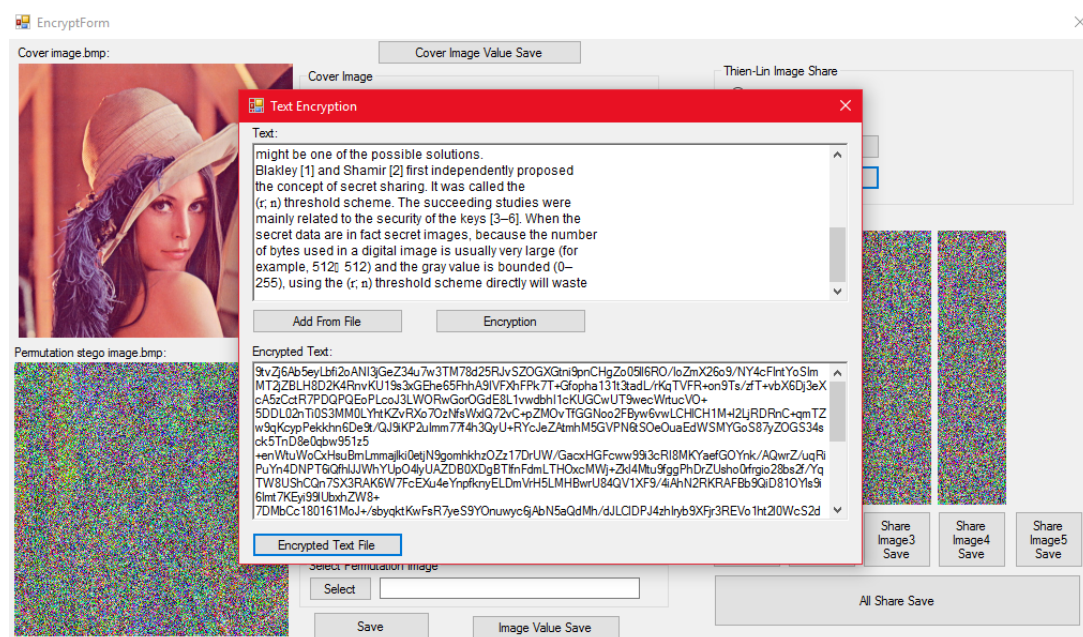


Figure 3. AES Encryption

The reconstruction of the secret image and secret key using $k = 2$ threshold image shares is shown in the **Figure 4**.

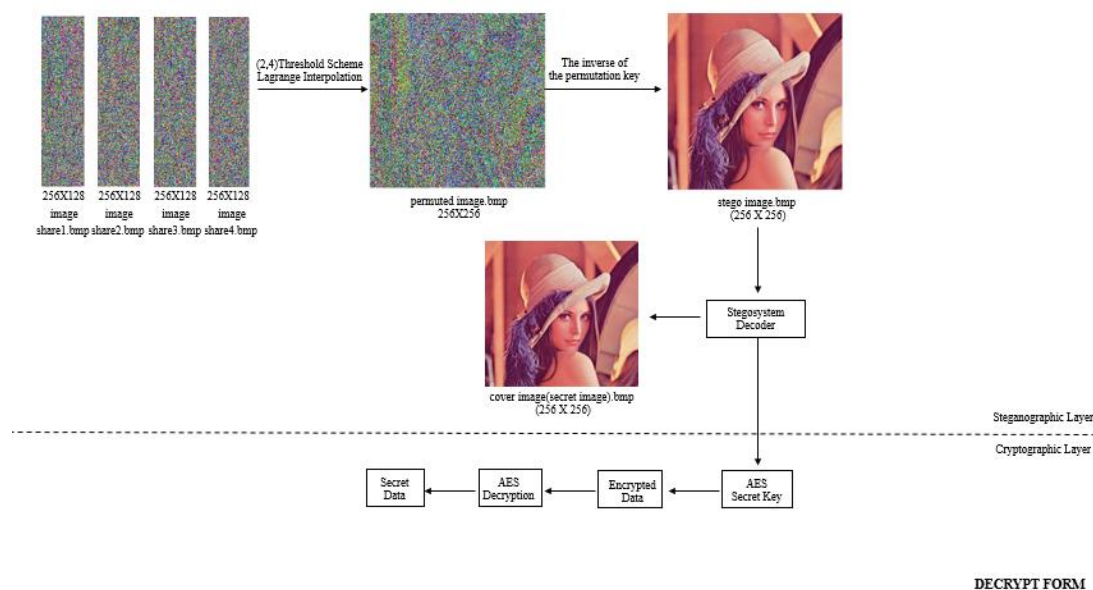


Figure 4. Reconstruction of secret image, secret key and AES decryption process

The steps in Figure 4 are explained with algorithm 3 and algorithm 4. The screenshot of the program is given in **Figure 5**

Algorithm 3. Reconstruction of secret image and secret key (Steganographic Layer)

Input: Any k shadows: $I_1, I_2, \dots, I_n$

Output: A secret image .and secret key

Step 1. Use $I_1, \dots, I_k$ and Lagrange's interpolation to recover the permuteted stego-secret image

Step 2. Obtain stego-image with reverse permutation

Step 3. Obtain secret image and secret key with stegosystem decoder.

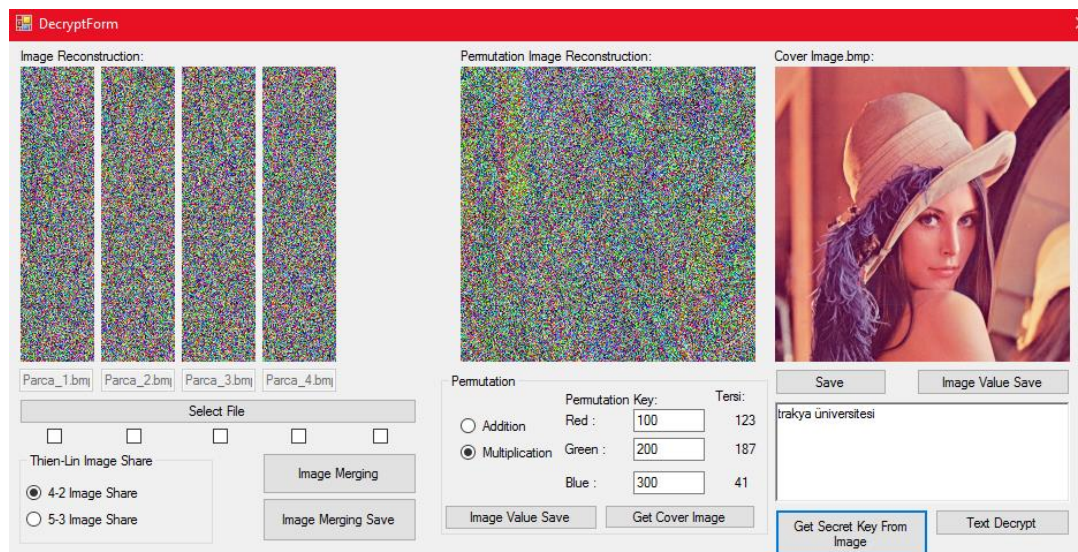


Figure 5. Reconstruction of secret image and secret key

As shown in **Figure 5** and in algorithm 3, Lagrange interpolation is used to obtain the original image from the shared images at the threshold number. We have shown that the first pixelin of the permuted stego image from the first and second share image values below.

$$Ip(0,0-1) = \left(222 \frac{x-2}{1-2} + 32 \frac{x-1}{2-1} \right) \mod 251$$

$$= 81x + 225$$

The same process was done for all pixel. Then the original image is obtained by inverse permutation and the hidden key is obtained by applying the stego-decoder to this image. The encrypted text is decrypted with the recovered secret key.

AES decryption has been implemented in **Figure 6**. The steps in **Figure 6** are described in the algorithm 4.

Algorithm 4. AES Decryption (Cryptographic Layer)

Input: Ecrption Data

Output: Secret Data

Step 1. AES decryption with secret key

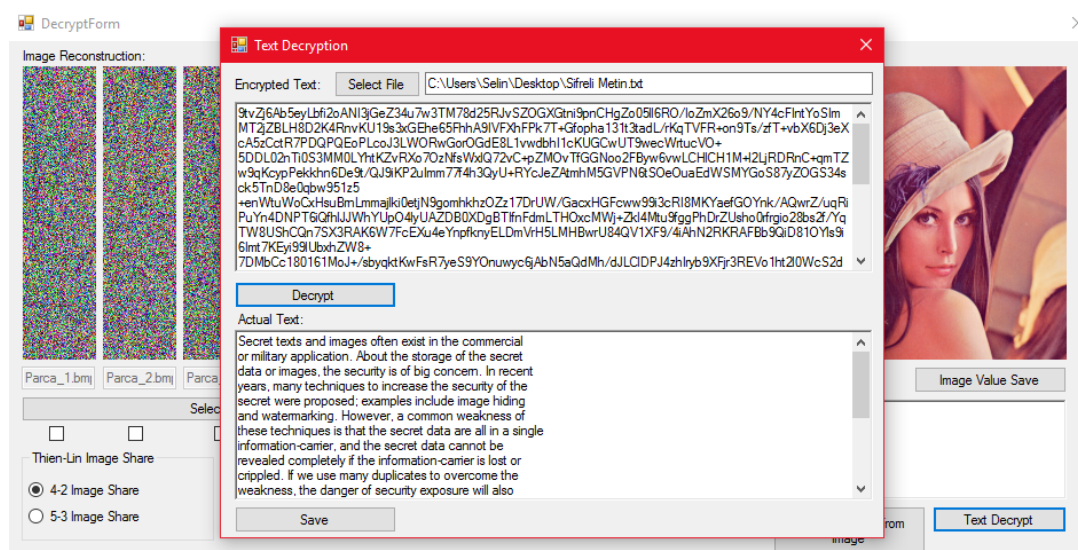


Figure 6. AES Decryption

5. CONCLUSIONS

In this article, the steganographic method and the secret image sharing scheme have been used together to securely store the secret key used in the AES encryption algorithm. First, the secret key is embedded in the RGB color image with the LSB technique only on the blue channel. By changing the LSB of a pixel results in small changes in the intensity of the colors. The changes cannot be identified by the human eye, thus the message is successfully hidden in the image. After that, the permutation process is done. Then the permuted stego-image is divided into share images by Thien and Lin Secret sharing scheme. It is quite difficult to reach a meaningful knowledge with any of these share images. But hidden key can be recovered with threshold-sharing images. Both of these methods have been applied together as a solution to the problem of keeping the cryptographic key secure. As a result, it has been shown that the cryptographic key is stored more securely. Similarly, it can be applied with different steganographic methods and different secret sharing schemes.

REFERENCES

- Shamir, A., 1979. "How to Share a Secret", Communications of the ACM, 22:612-61
- Blakely, G. R., 1979, "Safeguarding cryptography keys", Proc. AFIPS 1979 National Computer Conference, 48, (1979), 313-317.
- Asmuth C. A. and Bloom J., 1983, "A modular approach to key safeguardin", IEEE Transactions on Information Theory, IT-29(2): 208-210
- Mignotte. M., 1983, " How to share a secret", In T. Beth, editor, Cryptography- Proceedings of the Workshop on Cryptography, Burg Feuertein, 1982, volume 149 of Lecture Notes in Computer Science, pages 371-375. Springer- Verlag
- Thien C. C. and Lin J. C., 2006, "Secret image sharing," Computers & Graphics, vol. 26, no. 5, pp. 765–770, 2002 Indianapolis, Indiana. USA: IEEE Press, 31-36, ISBN: 0-7695-25393
- Moni Naor and Adi Shamir, 1995, "Visual cryptography". In Proceedings of Advances in Cryptology, EUROCRYPT 94, Lecture Notes in Computer Science, (950):pp. 1-12.
- Johnson, N. F., Katzenbeisser, S. 2000, "A Survey of Steganographic Techniques", Information Hiding Techniques for Steganography and Digital Watermarking, ISBN-13: 978-1580530354, 43-78.

Dumitrescu, S., W. Xiaolin and Z. Wang, 2003. "Detection of LSB steganography via sample pair analysis". In: LNCS, Vol. 2578, Springer-Verlag, New York, pp: 355 -372.

Johnson, N.F. & Jajodia, S., 1998, "Exploring Steganography: Seeing the Unseen", Computer Journal, February.

Owens, M., 2002, "A discussion of covert channels and steganography", SANS Institute

Kurosawa K., Obana S., Ogata W., 1995, "t-cheater identifiable (k, n) secret sharing schemes" In: Proceedings of Crypto'95, LNCS, vol. 963, pp. 410–423. Springer-Verlag

Schoenmaker, B., 2011, "Cryptography 2/Cryptographic Protocols", Lecture Notes Part 1 Cryptographic Protocols, version 0.999, The Netherlands.

Cimato, S., De Prisco, R., De Santis, A., 2007, "Colored visual cryptography without color darkening", Theoretical Computer Science, vol. 374, no. 1-3, pp. 261-276.