NETWORK SECURITY USING CRYPTOGRAPHY AND STEGANOGRAPHY

Sumesh Ranjan Sethi, Sidhartha Ku. Satapathy, Sushanta Kumar
(sumesh.rinku@gmail.com,sidhartha081@gmail.com,sushant0403@gmail.com)
UG Scholors, Bharath University

ABSTRACT:

Cryptography and steganography are well known and widely used techniques that manipulate information in order to cipher or hide their existence. These techniques have many applications in computer science and other related fields. They are used to protect e-mail messages, credit card information, corporate data, etc. In this paper we describe a method for integrating together cryptography and steganography through image processing. In particular, we present a system that able to perform steganography and cryptography at the same time using images as cover objects for steganography and as keys for cryptography. It is shown that such system is an effective steganographic one by making a comparison with the well known F5 algorithm and is also a theoretically unbreakable cryptographic one by demonstrating its equivalence to the Vernam Cipher.

INTRODUCTION

Cryptography and steganography are widely used techniques that manipulate information in order to cipher or hide their existence. These techniques have many applications in computer science and other related fields. They are used to protect e-mail messages, credit card information, corporate data, etc. Steganography is the art and science of communicating in a way which hides the existence of the communication. A steganographic system thus embeds hidden content in unremarkable cover media so as not to arouse an eavesdropper's suspicion. For e.g., it is possible to embed a text inside an image or an audio file. On the other hand, cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication. In this paper we will focus only on confidentiality. Cryptography and steganography are cousins in the spy craft family: the former scrambles a message so it cannot be understood; the latter hides the message so it cannot be seen.

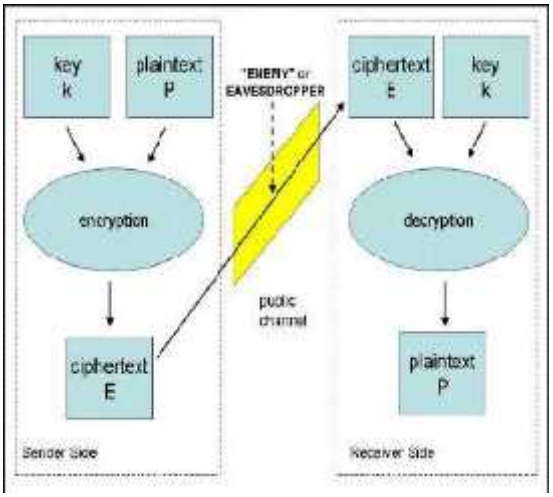

Figure 1. Symmetric Key Cryptographic Model.

A cipher message, for instance, might arouse suspicion on the part of the recipient while an invisible message created with steganographic methods will not. In fact, steganography can be useful when the use of cryptography is forbidden: where cryptography and strong encryption are banned steganography can circumvent such policies to pass message covertly. The disciplines that study techniques for deciphering cipher messages and detecting hide messages are called cryptanalysis and steganalysis. The former denotes the set of methods for obtaining the meaning of encrypted information, while the latter is the art of discovering covert messages. The aim of this paper is to describe a method for integrating together cryptography and steganography through image processing. In particular, we present a system able to perform steganography and cryptography at the same time.

IMAGE BASED STEGANOGRAPHIC SYSTEM

The majority of today's steganographic systems uses images as cover media because people often transmit digital pictures over email and other Internet communication (e.g.,

eBay). In this article, we will concentrate only on images as carrier media. The modern formulation of steganography is often given in terms of the prisoners' problem [8], [4] where Alice and Bob are two inmates who wish to communicate in order to make an escape plan. However, all communication between them is examined by the warden, Wendy, who will put them in solitary confinement at the slightest suspicion of covert communication. Specifically, in the general model for steganography (see Figure 2), we have Alice (the sender) wishing to send a secret message M to Bob (the receiver): in order to do this, Alice chooses a cover image C. The steganographic algorithm identifies C's redundant bits, then the embedding process creates a stego image S by replacing these redundant bits with data from M. S is transmitted over a public channel and is received by Bob only if Wendy has no suspicion on it. Once Bob recovers S, he can get M through the extracting process. The embedding process represents the critical task for a steganographic system since S must be as similar as possible to C for avoiding Wendy's intervention. Least significant bit (LSB) insertion overwrites the LSB of a pixel with an M's bit. If we choose a 24-bit image as cover, we can store 3 bits in each pixel. To the human eye, the resulting stego image will look identical to the cover image [2]. Westfield [9] proposed F5, an algorithm that does not overwrite LSB and preserves the stego image's statistical properties.

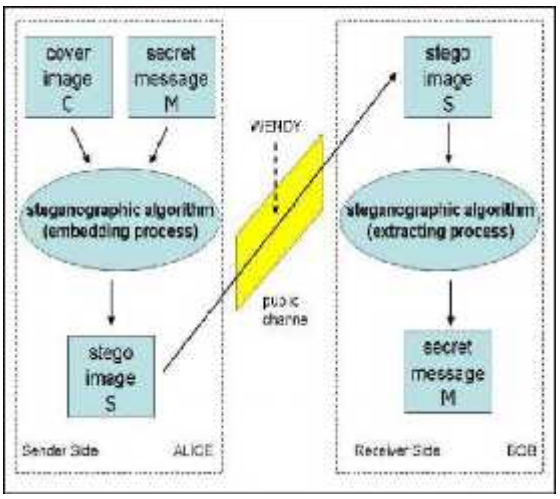

Figure 2. Steganographic Model

A STEGO- CRIPTOGRAPHIC MODEL

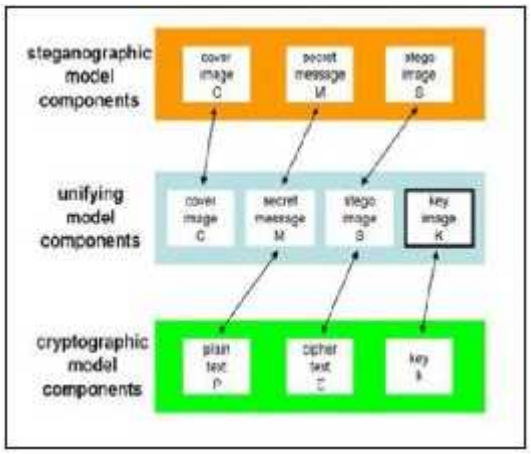

Figure 3. Mapping between Model Components

This is a new all-in one method able to perform steganography providing strong encryption at the same time. This method has been planned either to work with bit streams scattered over multiple images or to work with still images. The simplicity of this method gives the possibility of using it in real-time applications such as mobile video communication. Figures 1 and 2 depict the cryptographic and steganographic system components. The mapping between P and M, E and S, and k and K is possible because we can consider all the components in Figure 3 as bit sequences and then realize a relation between the corespective bit sets. The unifying model results as a steganographic one with the addition of a new element: the key image K. It gives the unifying model the cryptographic functionality we are searching for, reserving its steganographic nature. The unifying model embedding process yields S exploiting not only C's bits but also K's ones .In this way of proceeding gives Alice the chance to embed the secret message M (that is, the plaintext) into the cover image C encrypting M by the key image K at the same time. At the receiver side, Bob will be able to recover M through S

and K. In addition, Wendy will neither detect that M is embedded in S nor be able to access the content of the secret message (see Figure 4).

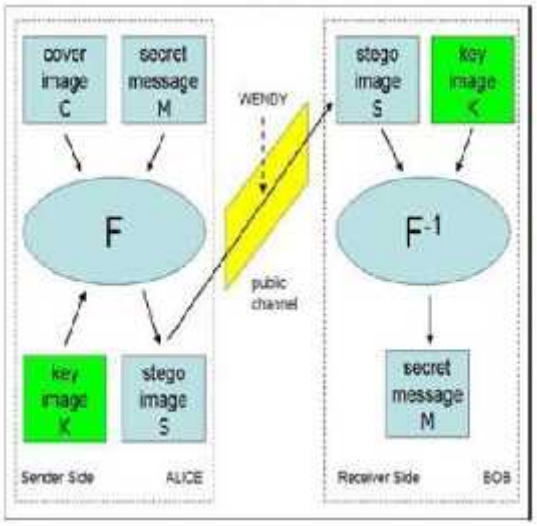

Figure 4. The Unifying Model

IMAGE BASED STEGANOGRAPHY AND CRIPTOGRAPHY (ISC)

The function denoted by F in Figure 4 represents the embedding function we are going to explain in this section. The symbol F−1 indicates the extraction function, since it is conceptually the inverse of embedding.

ISC Embedding Process

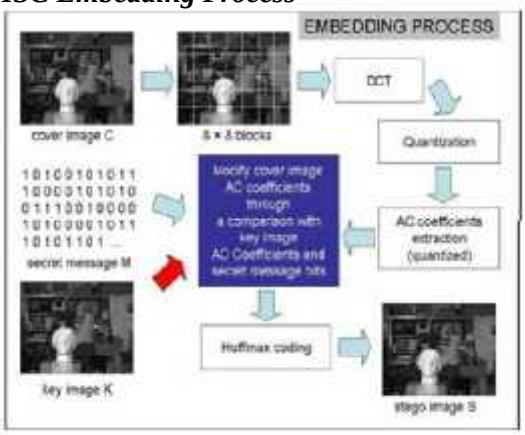

Figure 5. ISC Embedding Process

Figure 5 shows the embedding process. The choice of the stego image format makes a very big impact on the design of a secure steganographic system. Raw, uncompressed formats, such as BMP, provide the biggest space for secure steganography, but their obvious redundancy would arise Wendy's suspicion. Thus, ISC embedding algorithm must yield a compressed stego image, in particular we choose to produce a JPEG file, because it is the most widespread image format. While the output of the embedding process is a JPEG image, the inputs are: the secret message bit sequence, an image C, and an image K. C and K can be either uncompressed images (e.g., BMP) or compressed ones (e.g., JPEG), in addition they can be either distinct images or the same image.

The embedding process will be a modification of the JPEG encoding scheme. First of all, we subdivide C in a set of 8 x 8 pixel blocks and compute the Discrete Cosine Transform (DCT) on each block obtaining a set of DCT coefficients; then they are quantized. After quantization, DC coefficients and AC zero coefficients are discarded. We have to repeat the previous list of operations for the key image K obtaining keyAC[i], a signed integer array as coverAC[i]. Now, in order to yield the stego image S, we are able to modify coverAC[i] according to the following Em1 embedding algorithm. We will call stegoAC[i] the modified coverAC[i] array.

1) Embedding Algorithm Em1

Input: coverAC[i], keyAC[i], message bit array
M Output: stegoAC[i]
for every bit M[i] of the message array
M if (M[i] == 1) // we want to modify 1
if (coverAC[i] and keyAC[i] are both even or both odd numbers)
if(coverAC[i] == 1) stegoAC[i] = 2
else if(coverAC[i] == -1) stegoAC[i] = -2
else
if(random() < 0.5)
stegoAC[i] = coverAC[i] - 1; else
stegoAC[i] = coverAC[i] + 1; end if
else // M[i] = 0, we want to codify a 0
if (coverAC[i] and keyAC[i] are one equal and one uneven)
if(coverAC[i] == 1) stegoAC[i] = 2
else if(coverAC[i] == -1) stegoAC[i] = -2
else
if(random() < 0.5)
stegoAC[i] = coverAC[i] - 1; else
stegoAC[i] = coverAC[i] + 1; end if
end if
end for

In the algorithm, random () returns a real value in [0, 1) that is chosen pseudo randomly with uniform distribution from that range. Once the embedding algorithm terminates, we can proceed with stegoAC[i] Huffman coding and eventually we obtain a JPEG image S as similar as possible to C. We can embed into S a number of bits equal to min (length (coverAC[i]), length (keyAC[i])).

ISC Extracting Process

The ISC extracting process is very simple and consists in a comparison between S nonzero AC coefficients. In order to obtain these two sets of coefficients we perform a Huffman decoding step followed by the quantized AC coefficients extraction (see Figure 6).

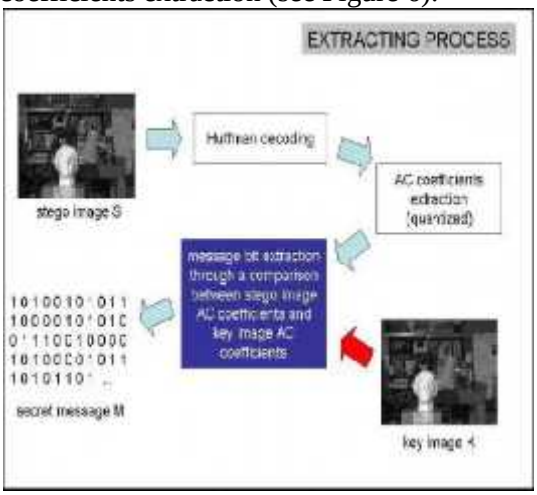

Figure 6. ISC Extracting Process

Once the extraction is finished we compute the following Ex1 extracting algorithm:

1) Extracting Algorithm Ex1

Input: stegoAC[i], keyAC[i] Output: message bit array M
for every coefficient stegoAC[i]
if (stegoAC[i] and keyAC[i] are both even or both odd) M[i] = 0;
else
M[i] = 1; end if end for
Images C and K depicted in Fig. 5 are two well known stereo images.

ISC PERFORMANCE

In this section we will present ISC performance ,we first demonstrate that ISC has optimum cryptographic performance, by proving that it is equivalent to Vernam cipher [5], and then compare ISC steganographic performance with respect to the well known F5 algorithm [9].

ISC Cryptographic Performance

The Vernam cipher is a symmetric-key cipher defined on the alphabet A = {0,1}. A binary message m1,m2, ...,mt is operated on by a binary key string k1,k2,...,kt of the same length to produce a cipher text string c1,c2,...,ct, where ci = mi XOR ki, for 1< i< t. The cipher text is turned back into plaintext simply inverting the previous procedure, i.e., mi = ci XOR ki, for 1< i< t.
If the key string is randomly chosen and never used again, the Vernam cipher is called a one-time pad. Onetime pad is theoretically unbreakable: if a cryptanalyst has a cipher text string c1, c2... ct encrypted using a random key string which as been used only once, the cryptanalyst can do no better then guess at the plaintext being any binary string of length t. To realize an unbreakable system requires a random key of the same length as the message [7].

Equivalence between Vernam Cipher and ISC

Let keyAC[i] and coverAC[i] be two arrays containing the AC nonzero coefficients extracted from the key image K and the cover image C respectively. Let stegoAC[ ] be an array initialized identical to coverAC[i]. Let M[i] be a binary array containing all the bits from the secret message M and let us suppose, for the sake of simplicity, that length(keyAC[i]) =length(coverAC[i])=length(M[i]). Now we transform Em1 in order to work with bit sequences, obtaining the algorithm Em2:

1) Embedding Algorithm Em2

Input: coverEO[i], keyEO[i], M[i] Output: stegoEO[i]
for every bit M[i] of the binary array M[i] if (M[i] == 1)
if (coverEO[i] XOR key EO[i] == 0) (1) stegoEO[i] = coverEO[i] XOR 1 (2) end if
end if
else //M[i] = 0

if (coverEO[i] XOR keyEO[i] == 1) (3) stegoEO[i] = coverEO[i] XOR 1 (4) end if
end else
end for

Lines 1,2,3, and 4 perform the same operations made by algorithm Em1. Table 1 shows the truth table for every input feasible by algorithm Em2.

Table I. Truth Table For Algorithm Em2.

M[i]	keyEO[i]	coverEO[i]	stegoEO[i]
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

You can notice that bold values correspond to the truth table for ci = mi XOR ki. Since M[i] corresponds to the Vernam plaintext m1,m2, ...,mt, keyAC[ ] corresponds to the Vernam key k1,k2, ...,kt, and stegoAC[ ] corresponds to the Vernam ciphertext c1,c2, ...,ct, we can conclude asserting: I S C embedding process and Vernam cipher encrypting step are equal.The proof of equivalence between ISC extracting process and Vernam cipher decrypting step is trivial. Let us transform the algorithm Ex1 in order to work with M[i], keyEO[i], and stegoEO[i].

1) Algorithm Ex2
Input: stegoEO[i], keyEO[i] Output: keyEO[i]
for every bit stegoEO[i] of stegoEO[i]
M[i] = stegoEO[i] XOR keyEO[i]
end for

Since Ex2 is identical to the Vernam cipher decrypting step (mi = ci XOR ki, for 1< i<t), we have that ISC extracting process and Vernam cipher decrypting step are equal.

Eventually, ISC and Vernam cipher are equivalent.

ISC Steganographic Performance

The ISC steganographic performance will be measured by comparing it with the well known F5 algorithm [9].In order to do this, we will compare the statistical behavior of these two algorithms on the same input set. This will demonstrate that ISC withstands both visual and statistical attacks [10],visual attacks mean that one can see steganographic messages on the low bit planes of an image because they overwrite visual structures; statistical attacks consist in measure distortions in the DCT coefficients' frequency histogram produced by embedding.

1) F5 Algorithm

The F5 steganographic algorithm was introduced by Andreas Westfeld in 2001 [9]. Instead of replacing the least-significant bit of a DCT coefficient with message data, F5 decrements its absolute value in a process called matrix encoding. Moreover, F5 (as ISC) embeds data in JPEG images thus resulting immune against visual attacks because it operates in a transform space and not in a spatial domain.

2) Comparison Between F5 and ISC

In order to realize a meaningful comparison between ISC and F52, we must embed the same message m into the same cover image c using both ISC and F5. After embedding, we have two stego images: SF5 produced by F5 and SISC generated by ISC. Both SF5 and SISC present a DCT coefficients histogram different from the c's original one.

What we are interested in is to compare the amount of modifications introduced by F5 and ISC.

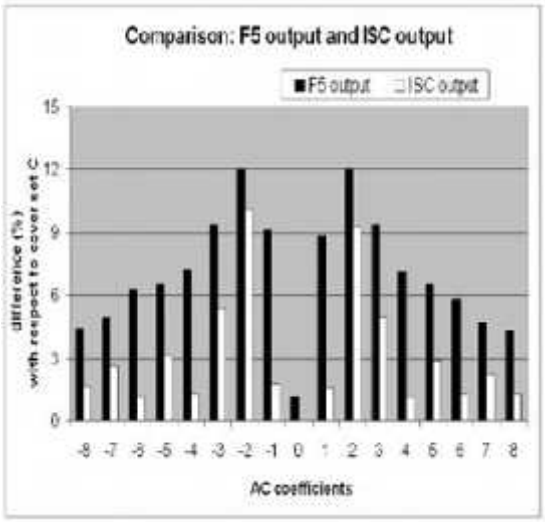

Figure 7. F5 and ISC Comparison

Figure 7 shows the result of such comparison obtained using a JPEG cover set Cset of 20 images. Only for ISC, we also used the images of Cset as key images. In Figure 7, in particular the black columns represent the differences introduced by F5 embedding step while the white ones correspond to the number of modifications yielded by ISC embedding process.

CONCLUSION

In this paper we have presented a novel method for integrating in an uniform model cryptography and steganography. The presented ISC algorithm is both an effective steganographic method (we made a comparison with F5) as well as a theoretically unbreakable cryptographic one (ISC is an image based one-time pad). T he strength of this system resides in the new concept of key image. Involving two images (the cover and the key) in place of only one (the cover) we are able to change the cover coefficients randomly. This opportunity does not give a steganalytic tool the chance of searching for a predictable set of modifications. The proposed approach has many applications in hiding and coding messages within standard medias, such as images or videos. As future work, we intend to study steganalytic techniques for ISC and to extend ISC to mobile video communication.

REFERENCES

[1] Johnson, Neil F. and Sushil Jajodia. "Exploring Steganography: Seeing the Unseen." ,IEEE Computer,32:2. 26-34. 1998.

[2] Kerckhoffs, A"La cryptographie militaire. " Journal des Sciences Militaries, 9th series (IX):5–38, (1883).

[3] Kharrazi, M., Sencar, H. T., and Memon, N. "Image steganography: Concepts and practice." , In WSPC Lecture Notes Series, (2004).

[4] Menezes, A., van Oorschot, P., and Vanstone, S. "Handbook of Applied Cryptography.", CRC Press, (1996).

[5] Provos, N. and Honeyman, P. "Hide and seek: An introduction to steganography.", IEEE SECURITY & PRIVACY, (2003).

[7] Simmons, G. J. "The prisoners' problem and the subliminal channel. ",In Advances in Cryptology:
Proceedings of Crypto 83, pages 51–67. Plenum Press, (1984).

[8] Westfeld, A., "F5-a steganographic algorithm: High capacity despite better steganalysis.", In Proc. 4th Int'l
Workshop Information Hiding, pages 289–302, (2001).

Dr.T.Saravanan has obtained his bachelor degree from Madras University in 2002. He obtained his M.E from in Annamalai University; Chidambaram 2005. He did his Ph.D from Sathyabama University in 2012. He is a life member of IETE. His research area is Power Electronic converter.

Sumesh Ranjan Sethi, Sidhartha Ku. Satapathy, Sushanta Kumar - UG Scholors, Doing Final Year Electronics and Telecommunication Engineering at Bharath University.