A Novel (k, n) Secret key sharing scheme based on Linear Equations

Bhaskar Mondal[1], Dr. Tarni Mandal [2], Sunil Kumar Singh [1], Krishna Mohan Acharjee [1]
[1]*Department of Computer Science and Engineering*
[2]*Department of Mathematics*
National Institute of Technology Jamshedpur
India 831014

Abstract

This paper presents an approach towards the (k,n) key sharing scheme using linear equations. This approach allows the key to be shared between 'n' number of people such that: i)any 'k' shares (k<=n) are sufficient to reconstruct the secret key in the lossless manner and ii) any (k-1) or fewer image shares cannot get enough information to retrieve the actual key. It's an effective and secure method to praise the sharing scheme using linear equations. The approach's advantage completely comes in its strong protection of key.

1. Introduction

Increasing instantaneous communication technology brought forward many new issues like steganography[9] cryptography[11], secret sharing[10] etc. In time secret sharing schemes became one of the attraction to provide security of required data in number of people: (n,n) sharing scheme proposes that a data divided between 'n' number of people could be retrieved only when all the shares are well known to our computing device which is the medium to reconstruct data. With the proceeding time , requirements increased and came forward the (k,n) sharing scheme. Blakley[1] and Shamir[2] invented two (k,n) threshold based secret sharing scheme independently in 1979, which encode a secret image into n shares. The secret image can only be reconstructed from any k or more shares. Knowledge of k-1 or fewer shares provides absolutely no information about the secret. SS[9] can not only guarantee the security of information, but also greatly reduce the possibility of secret inaccessible due to misfortune or betrayal, thus it has attracted many scholars' attention. A secret sharing scheme can be evaluated by its security, contrast (reconstruction precision), computational complexity, and pixel expansion (storage requirement) in case of image.

In this paper, we proposed secret key sharing and have modified the existing schemes to provide a better and efficient technique. The previous scheme proposed by Dong and Ku [8] makes the use of matrix multiplication property for construction of shares and addition of shares to reconstruct the secret. We have improved the share construction technique by reducing the computational complexity by applying matrix addition instead of matrix multiplication. However image reconstruction still uses the matrix addition property. Our scheme has no pixel expansion and retains the contrast of the original secret image. Considering an image of size h×h pixels, the computational complexity of matrix multiplication is O(h3), whereas that of matrix addition is O(h2). The complexity of share generation improves in our scheme as compared to Dong and Ku [8]. Hence our proposed scheme adds to the merits of already known secret sharing schemes and optimizes it.

2. Preliminaries

2.1. LSB Substitution: A most widely used steganography method is the least significant bit (LSB) substitution technique. In this approach, the secret message bits are concealed into a digital cover image by replacing a number of the least significant bits (LSB) of the cover image. In digital image most of the significant information is carried out by the most significant bits (MSB) so changing the parts of MSB of the cover image will seriously degrade the quality of the stego-image. Thus, the LSB substitution scheme decides to embed sensitive data into the parts of LSB of the cover image. Figure 1 depicts the processes of the LSB substitution scheme where 8 bits secret message are embedded into the sub-image of size 2×2 by replacing first 2 LSB bits of each pixel. After embedding all secret message bits into the cover image, the cover image containing the secret message is termed as the stego-image. In general up to first 3 LSB bits of the cover image are used to conceal the secret message bits otherwise the

quality of the stego-image will be degraded excessively. The reason behind for this degradation can be explained from Figure 2 where it shows the bit plane decomposition of "Pepper" image.

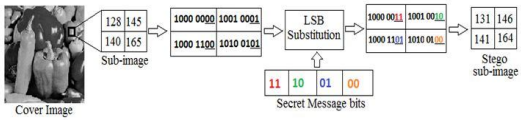

Figure 2: The LSB Substitution Scheme

From this figure, it is observed that first 3 LSB bit planes are appeared as randomly where rest of the bit planes carries most of visual information. So modifying out of first 3 LSB bits of the cover image will degrade excessively the quality of the stego-image.

The secret message extraction process from the stego-image is a straight forward process where the secret message bits are extracted from the concealed LSB bits of each pixel in the stego-image sequentially. Both the embedding and the extracting processes of the LSB substitution[10] scheme do not require complex computations. Thus, this scheme is very simple and has less computational overhead.

2.2. Secret Sharing: Secret Sharing refers to a method for distributing a secret amongst a group of participants, each of whom is allocated a share of the secret. The secret can be reconstructed only when a sufficient number of shares are combined together; individual shares are of no use on their own.

Now, the definition of secret sharing scheme is as follows:

Definition [2]: A (k, n) secret sharing scheme divides a secret s into n shares s1,…,sn such that the following terms and conditions are satisfied.

(T1): The secret s is recoverable from any k shares, i.e., for any set of k indices, H(s|(si1,...,sik)) = 0.

(T2): Knowledge of k-1 or fewer shares provides absolutely no information about s, i.e., for any set of k-1 indices, H(s|(si1,...,sik-1)) = H(s) ,where H(s) denotes the uncertainty of s, H(a|b) denotes the uncertainty of a when event b happened.

The first condition is called precision and the second condition is called security. When k = n, it is the definition of (n, n) secret sharing scheme.

2.3. Secret Key Sharing: As we already told that we are going to utilize linear equation for sharing scheme, first question comes in mind is how?? As we know that an equation can be solved surely if we know the exact number of equation same as that of number of variables. The proposed approach is solely dependent upon this concept.

2.4. Linear Equation: As we already told that we are going to utilize linear equation for sharing scheme, first question comes in mind is how? As we know that an equation can be solved surely if we know the exact number of equation same as that of number of variables. The proposed approach is solely dependent upon this concept.

3. Proposed Scheme:

The proposed key sharing scheme for key in standard Hexadecimal format, which consists of shares construction phase and revealing phase, is given as follows:

Input: Key in Hexadecimal format which generates 'x' number of binary bits after converting to secret binary string.

3.1. Share Construction:

Step 1: Generating a random matrix A of size x=a*b with each element having decimal number ranging from 0 to 'c' where: (c>k<n). If x=64 bits then, it may be a=8 & b=8.

Step 2: Embed the key into the random matrix A using LSB Substitution techniques. Say the resulted matrix is A'.

Step 3: As minimum k number of shares required for retrieving the original key; we have to share our above resulting matrix A' in between 'k' coefficients.

i. Generate k-1 number of random matrices A_i (where i={1,2….,k-1}) with decimal number element in it, ranging [0 to 'c'].

ii. Generation of k^{th} random matrices A_k

$A_k=\{(2*c)J - \sum_{i=1}^{k-1} a_i\} \bmod (2*c)$; where J is a unit matrix

Step 4: Generation of k number of intermediate shares S_i where i={1,2…..,k}:

i. S_1 to S_{k-1} can be computed using following computational formulae:

$S_i= \{a_i + h\} \bmod (2*c)$

and S_k can be computed as :

$S_i= \{a_k + (2*c)J – (k-2)h\} \bmod (2*c)$

Step 5: Generating k number of prime numbers $P_i=\{1,2,….,n\}$ for each n share; with a precaution that no one shares has exactly same set of prime numbers with exactly same sequence.

Example:

P = {211, 157, 59}

P_2= {163, 37, 5}
P_3= {233, 31, 103}
P_4= {131, 239, 101}
P_5= {109, 151, 13}

Step 6: Computing n numbers of matrices D_i where i={1,2….,n} using following computational formula:

$D_i = \sum_{j=0}^{k} (S_j * P_{ij})$

Where P_{ij} is the j^{th} element of set P_i

Thus D_i will be a × b matrix which is the n^{th} shares.

Similarly, P_i also going to be shared in such manner:

First share: [D_1 and P_1]

Second share: [D_2 and P_2] etc.

3.2. Key Reconstruction:

Step 1: Taking D_i and P_i as input in our computing device from various share to reconstruct the key. The computing device will go for solving linear equation with k number of variables where it will take P_i as coefficient and D_i as constant in equations. Like:

$P_{11} * x + P_{12} * y + P_{13} * z = D_1$
$P_{21} * x + P_{22} * y + P_{23} * z = D_2$
$P_{31} * x + P_{32} * y + P_{33} * z = D_3$
$P_{41} * x + P_{42} * y + P_{43} * z = D_4$
$P_{51} * x + P_{52} * y + P_{53} * z = D_5$

Solving this Linear equation to find S_i

As here x is representing S_1, y is representing S and z is representing S_3

Step 2: Computing;

$A' = (\sum_{i=1}^{k} S_i) \bmod (2*c)$

Step 3: Extracting LSB from each element of matrix A' to find the secret bits and convert the bits to hexadecimal.

4. Experimental Results

We had applied the method to share a 16 character hexadecimal key. After converting to binary its become 64 bits.

We hide these 64 bits in a matrix of 64 elements. And then we have generated 5 shares out of the 5 shares it was possible to retrieve the secret using any three or more than three shares. Knowledge of less than three shares will not able to reveal any information about the secret.

5. Security Analysis

As we know that there are many methods which can be used to solve equation in less than the required number of variables; to overcome it we have intelligently used prime numbers as the coefficient cause all the methods to eliminate requirement of equation is based on the mistake of taking parallel equations as we are taking prime numbers thus there is no chance that the equation could get parallel except that each of the coefficient are equal simultaneously.

ii) There may be a possibility that people with 'k-1' share can use brute force method to extract the required key but our method is already as manipulated that still people with the 'k-1' share have to do a brute force attempt of $(512)^{256}$ for a key of 256 binary bits; which is clearly useless to go cause it would be better for the hacker to directly do brute force of $(2)^{256}$ for the same key.

iii)There is another possibility that hacker can use the idea of variables being integral to solve out the equation for it we can simply add some decimal value to S_i and as per the requirement can get the floor value to get our result right.

This simple manipulation will surely strike there last hope too.

5. Conclusion

This paper proposed a new (k,n) secret key sharing scheme which uses addition for the construction and reconstruction operation. Compared with the other sharing schemes, the proposed (*k, n*) scheme can construct random shares and reconstruct the secret key precisely with low computational complexity. Common software tools, such as Matlab can be used to implement the matrix operations and reconstruct the secret images.

It can be easily extended to image or text sharing. Moreover, the proposed schemes provide enormous security. The obvious advantages of our schemes in terms of low computation complexity, no pixel expansion and high reconstruction contrast/accuracy are encouraging. Secret sharing schemes have a vast scope of improvement. Researchers are looking for new fields of applications. We are currently investigating the approaches of extending this scheme to a more general *(k, n)* scheme and other schemes like multi-image sharing and video streaming.

10. References

[1] G. R. Blakley, "Safeguarding cryptographic keys," *Proc. AFIPS NCC*, vol.48, 1979, pp.313-317.
[2] A.Shamir, "How to share a secret," *Commun. ACM*, vol.22 (11), 1979, pp.612-613.
[3] C. C. Thien, J. C. Lin, "Secret image sharing," *Computers and Graphics*, vol.26(5) , 2002, pp.765-770.
[4] M. Naor, A. Shamir, "Visual cryptography," *Advances in Cryptology-EUROCRYPTO'94, Springer-Verlag*, vol.950, 1995, pp.1-12.
[5] M. Iwamoto, H. Yamamoto, "The optimal n-out of-n visual secret sharing scheme for gray-scale images," *IEICE Trans. Fundam.* E85-A(10) , 2002, pp.2238–2247.

[6] F. Yi, D.S. Wang, P. Luo, Y.q. Dai, "Two new color (n, n)-secret sharing schemes," *Journal on Communications (Chinese)*, vol.28(5), 2007,pp.30-35.

[7] D.S.Wang, L. Zhang, N. Ma, X.B. Li, "Two secret sharing schemes based on Boolean operations," *Pattern Recognition*, vol.40, 2007, pp.2776-2785.

[8] Lin Dong, Min Ku, "Novel (n,n) secret image sharing scheme based on addition," *Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (2010)*, iih-msp, pp.583-586.

[9] Bhaskar Mondal et. al. "An Optimal (n, n) secret image sharing schem" *International Journal of Computer Science and its Applications* – Volume 2(3), 2012; Page(s): 61 – 66;

[10] Bhaskar Mondal, S. K. Singh "A Highly Secure Steganography Scheme For Secure Communication", *Proc International Conference of Computation and Communication Advancement (IC3A)-2013*, JIS College of Engineering, January, 2013.

[11] Bhaskar Mondal and et. al. "An Improved Cryptography Scheme for Secure Image Communication", *International Journal of Computer Applications (0975 – 8887)* Number 18 (ISBN: 973-93-80874-18-3) April 2013 Issue. Volume 67(18) pages 23-27.