Threshold Secret Image Sharing

Teng Guo[1,2], Feng Liu[1], ChuanKun Wu[1], ChingNung Yang[3], Wen Wang[1,2], and YaWei Ren[1,2,4]

[1]State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
[2]University of Chinese Academy of Sciences, Beijing 100190, China
[3]Department of Computer Science and Information Engineering, National Dong Hwa University, Hualien 974, Taiwan.
[4]School of Information Management, Beijing Information Science and Technology University, Beijing 100192, China
{guoteng,liufeng,ckwu,wangwen}@iie.ac.cn,cnyang@mail.ndhu.edu.tw

Abstract. A (k, n) threshold *secret image sharing scheme*, abbreviated as (k, n)-TSISS, splits a secret image into n shadow images in such a way that any k shadow images can be used to reconstruct the secret image exactly. In 2002, for (k, n)-TSISS, Thien and Lin reduced the size of each shadow image to $\frac{1}{k}$ of the original secret image. Their main technique is by adopting *all coefficients* of a $(k-1)$-degree polynomial to embed the secret pixels. This benefit of small shadow size has drawn many researcher's attention and their technique has been extensively used in the following studies. In this paper, we first show that this technique is neither information theoretic secure nor computational secure. Furthermore, we point out the security defect of previous (k, n)-TSISSs for sharing textual images, and then fix up this security defect by adding an AES encryption process. At last, we prove that this new (k, n)-TSISS is computational secure.

Keywords: Secret image sharing, Security defect, Computational secure

1 Introduction

Secret image sharing has drawn considerable attention in recent years [3, 6, 14, 16, 20–23, 29–31]. A (k, n) threshold *secret image sharing scheme*, abbreviated as (k, n)-TSISS, encrypts a secret image into n shadow images (also referred to be shadows) in such a way that any k shadows can be used to reconstruct the secret image exactly, but any less than k shadows should provide no information about the secret image. The secret pixel can be hidden in the

constant term of a $(k-1)$-degree polynomial using Shamir's (k, n) secret sharing scheme, abbreviated as Shamir's (k, n)-SSS [19], and the secret image can be perfectly reconstructed from any k shadows by Lagrange's interpolation. In such a case, each shadow is the same size as the secret image. For example, to encrypt a $10GB$ satellite image by a $(5, 10)$-TSISS, we get 10 shadows, each with size $10GB$; and to reconstruct the $10GB$ satellite image, we have to collect 5 shadows, which sum up to $50GB$. The larger the amount of information grows, the severer the above problem suffers from. To solve this large shadow size problem in secret image sharing, Thien and Lin [20] embed the secret pixels in *all coefficients* of a $(k-1)$-degree polynomial and reduce the shadow size to $\frac{1}{k}$ of the secret image. This *variant* use of Shamir's (k, n)-SSS is denoted as (k, n)-VSSS in this paper. Since the smaller shadow size makes the transmission and storage more convenient, the (k, n)-VSSS has drawn many attentions in the following studies [2–4, 6, 7, 14, 16, 20–26, 28–32] of TSISS ever since. Initially, Thien and Lin [20] adopt $GF(251)$ as the coefficient field, and the pixel's gray-level degrees has to be modified to less than 251. Therefore, Thien and Lin's scheme is in fact a lossy secret image sharing scheme, in which the reconstructed secret image may be distorted slightly in gray-level. In 2007, Yang et al. [27] adopt $GF(2^8)$ as the coefficient field, avoiding the losses in gray-level. Recently, (k, n)-TSISS has been combined with steganography and authentication [3, 4, 6, 28, 31], which divides a secret image into several shadows and embeds the produced shadows in the cover images to form the stego images, which can be transmitted to authorized recipients without causing suspicion. In addition, these schemes also have some authentication mechanisms to verify the integrity of the stego images, so that the secret image can be reconstructed correctly. (k, n)-TSISS has also been combined with visual cryptography [1, 10–13, 17, 18], which provides a two-in-one (k, n)-TSISS [16, 29] with two decoding options: the first option is stacking shadows to see a vague reconstructed image like visual cryptography; and the second option is to perfectly reconstruct the original gray-level secret image by Lagrange's interpolation.

However, there is no free lunch. The (k, n)-VSSS is no longer information theoretic secure, and to the best of our knowledge, no research has conjectured that the inverting of a $k-1$ degree polynomial $f(x)$ from less than or equal to $k-1$ shadows is computational infeasible. From this viewpoint, all of the above mentioned studies of (k, n)-TSISS provide neither of the two currently well-known security guarantees: 1, *information theoretic security* (also known as perfect secrecy), which is based on Shannon's information theory, e.g. the

one-time pad, Shamir's secret sharing scheme, visual cryptography; 2, *computational security*, which is based on computational hardness assumptions, e.g. RSA, AES, DES. Motivated by the above observation, we in fact find the security defect of previous (k, n)-TSISSs for sharing textual images, in which the secret can be perceived from any single shadow. Please refer to Section 3.

To avoid the above security defect, we suggest to add an AES encryption process before the sharing process to form a computational secure (k, n)-TSISS, which is denoted by (k, n)-CSTSISS. Then we prove it is computational secure by giving a construction that transforms any efficient attack of the (k, n)-CSTSISS to an efficient attack of AES. In addition to theoretic analysis, experimental results are given to show feasibility of the proposed scheme. Compared to previous (k, n)-TSISSs, the proposed (k, n)-CSTSISS needs $256n$ bits more storage space in overall, and more time for the AES encryption and decryption processes.

This paper is organized as follows. In Section 2, we give some preliminaries of TSISS. In Section 3, we point out the security defect of the previous (k, n)-TSISSs. In Section 4, we propose a computational secure (k, n)-TSISS. The paper is concluded in Section 5.

2 Preliminaries

In this section, we first give some basic knowledge of Shamir's secret sharing and its variant version that is commonly used in studies of TSISS, and then analyze their security properties sequentially.

Suppose the secret we are going to share is in some finite field, e.g. prime fields $GF(p)$ or prime power fields $GF(2^n)$. For simplicity, we will take $GF(p)$ for example to illustrate the sharing process in the following. This will not cause any limitations, for the underlying principle is the same except that the operations in $GF(2^n)$ are modular of some irreducible polynomial of degree n, while those of $GF(p)$ are modular of prime number p.

In *Shamir's* (k, n)*-SSS*, to divide the secret $S \in GF(p)$ into n shadows $S_i \in GF(p)$ $(1 \leq i \leq n)$, we first pick up $k-1$ random numbers $a_1, a_2, \ldots, a_{k-1}$ from $GF(p)$ and form a $k-1$ degree polynomial $f(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + \ldots + a_{k-1} x^{k-1}$ with $a_0 = S$. Then we evaluate each shadow by $S_i = f(i)$ $(1 \leq i \leq n)$. From any k-subset of these S_i values, we can reconstruct $f(x)$ by Lagrange's interpolation and compute the secret by $S = f(0)$. However, from any $(k-1)$-subset of these S_i values, we get no information about S. Detailed analysis can be found in [19].

In *the* (k, n)*-VSSS*, which is widely used in the studies of (k, n)-TSISS, to divide the secret $D = (D_0, D_1, \ldots, D_{k-1})$ with $D_0, D_1, \ldots, D_{k-1} \in GF(p)$ into n shadows $S_i \in GF(p)$ $(1 \leq i \leq n)$, we first form a $k-1$ degree polynomial $f(x) = D_0 + D_1 x + D_2 x^2 + D_3 x^3 + \ldots + D_{k-1} x^{k-1}$. Then we evaluate each shadow by $S_i = f(i)$ $(1 \leq i \leq n)$. From any k-subset of these S_i values, we can reconstruct $f(x)$ by Lagrange's interpolation and obtain all the coefficients $(D_0, D_1, \ldots, D_{k-1}) = D$. Detailed analysis of the information leakage can be found in the proof of Theorem 2.

To analyze the information leakage of the (k, n)-VSSS from less than k shadows, we have to assume a probability distribution on the secret. For the simplicity of analysis and consistency with our proposed scheme, we assume that the secret is uniformly distributed in its space. For some knowledge of information theory, one can refer to [5]. Here we only give some necessary backgrounds. *Entropy* is a measure of the uncertainty associated with a random variable. Suppose X is a random variable, its entropy is defined by $H(X) = \sum_{x \in X} p(x) \log \frac{1}{p(x)}$[1]. The amount of randomness in random variable X given that you know the value of random variable Y is defined by $H(X|Y) = \sum_{x \in X, y \in Y} p(x, y) \log \frac{p(y)}{p(x, y)}$, which is also known as the conditional entropy of X and Y. Briefly speaking, a (k, n)-SSS is information theoretic secure if any $k-1$ shadows provide no information about the secret. Formally, this notion is given as follows.

Definition 1 (Information theoretic secure). *In a* (k, n)*-SSS, suppose the secret is distributed according to random variable* S *and the* n *shadows are distributed according to random variables* $S_1, S_2, \ldots, S_n$*. The* (k, n)*-SSS is information theoretic secure if* $H(S) = H(S|S_{i_1}, \ldots, S_{i_{k-1}})$ *holds for any* $k-1$ *shadows* $S_{i_1}, \ldots, S_{i_{k-1}}$*.*

Theorem 1 ([19]). *Shamir's* (k, n)*-SSS is information theoretic secure.*

Theorem 2. *In the* (k, n)*-VSSS, there is only* $\frac{1}{k}$ *fraction of the uncertainty of the secret left, given the knowledge of any* $k-1$ *shadows.*

Proof: Having no shadows, the uncertainty of the secret $D = (D_0, D_1, \ldots, D_{k-1})$ is $H(D) = \sum_{i=0}^{p^k - 1} \frac{1}{p^k} \log p^k = \log p^k = k \log p$. Now we calculate the uncertainty of the secret D, provided any $k-1$ shadows $S_{i_1}, \ldots, S_{i_{k-1}}$. For each candidate

[1] In this paper, the base of logarithm is 2.

$D_0 \in GF(p)$, we have exactly one $k-1$ degree polynomial $f'_j(x)$ with $f'_j(0) = D_0$ and $f'_j(i_t) = S_{i_t}$ for $1 \le t \le k-1$. As D_0 is randomly drawn from $GF(p)$, the p polynomials $f'_j(x)$ are equally likely to be the real one. Hence the uncertainty of the secret D with the knowledge of any $k-1$ shadows is $H(D|S_{i_1}, \ldots, S_{i_{k-1}}) = \sum_{i=0}^{p-1} \frac{1}{p} \log p = \log p$. The amount of information we have gotten from any $k-1$ shadows is $H(D) - H(D|S_{i_1}, \ldots, S_{i_{k-1}}) = k \log p - \log p = (k-1) \log p$ and only $\frac{1}{k}$ fraction of the uncertainty of the secret D is still left. □

As Theorem 2 shows, the (k, n)-VSSS is far from being information theoretic secure. On the other hand, from the computational viewpoint, no research has conjectured that the inverting of a $k-1$ degree polynomial $f(x)$ from less than or equal to $k-1$ shadows is computational infeasible. Hence the (k, n)-VSSS that is widely used in secret image sharing has neither of the two well-known security guarantees, which may cause security risks to the previous studies of TSISS [2–4, 6, 7, 14, 16, 20–26, 28–32]. In the following, we present the common subprogram that all the above studies of (k, n)-TSISS share formally as Construction 1.

Construction 1

Input: *A secret image S.*

Output: *n shadows $S_1, S_2, \ldots, S_n$.*

Step 1. *Adopt all coefficients of a $k-1$ degree polynomial $f(x)$ to embed the secret pixels of S. Each time we share k successive pixels, say $p_1, p_2, \ldots, p_k$. Then fix $f(x) = p_1 + p_2 x + p_3 x^2 + \ldots + p_k x^{k-1}$. The pixel value for each shadow is calculated as $q_i = f(i)$ for $i = 1, 2, \ldots, n$. Repeat the above process until all pixels of S have been shared. The shadows are denoted as $S_1, S_2, \cdots, S_n$.*

Step 2. *Participant i is distributed S_i for $i = 1, 2, \ldots, n$.*

Remark: In Step 1., we use the (k, n)-VSSS, whose security is not guaranteed. This may cause hidden security risk to Construction 1. Indeed we have found its security defect for sharing textual images, please refer to Section 3.

3 The security defect of Construction 1

In this section, we present some experimental results to illustrate the security defect of Construction 1, while concrete theoretical analysis of the experiment is given in the Appendix.

Here we only give a general idea of the cause of the security defect. Since the sharing process (Step 1.) of Construction 1 is deterministic, the same combination of k secret pixels will always contribute to the same combination of n share values. In such a case, if the secret image is of little variation in gray-level, e.g. textual images, its content might be leaked from a single shadow.

For Construction 1 of (2, 3) threshold access structure, the experimental results on coefficient fields $GF(251)$ and $GF(2^8)$ can be found in Figures 1 and 2 respectively, in which any single shadow reveals the content of the secret image.

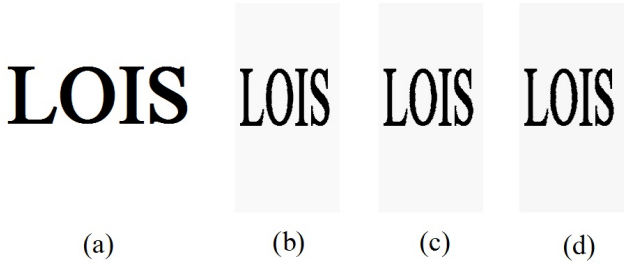

Fig. 1. Experimental results of Construction 1 on $GF(251)$, (a) the original secret image with image size 300×300, (b) shadow 1 with image size 150×300, (c) shadow 2 with image size 150×300, (d) shadow 3 with image size 150×300

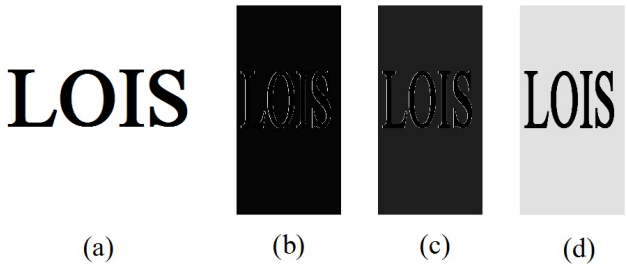

Fig. 2. Experimental results of Construction 1 on $GF(2^8)$, (a) the original secret image with image size 300×300, (b) shadow 1 with image size 150×300, (c) shadow 2 with image size 150×300, (d) shadow 3 with image size 150×300

Remark: The above security defect seems to be obvious, so why it is not discovered in previous studies? One of the reasons may be that in previous experiments,

they only use Construction 1 to encode natural dithered images and never use Construction 1 to encode textual images. But we think a good secret image sharing scheme should be able to deal with all kinds of images, and shouldn't make any restriction on the content of the image.

4 The proposed computational secure (k, n)-TSISS

In this section, we first propose a new (k, n)-TSISS. Then we will prove that this (k, n)-TSISS is computational secure.

Definition 2 (Computational secure). *Let the secret s be drawn from $GF(2^m)$. A (k, n)-SSS is computational secure if for any probability polynomial-time (PPT) algorithm A, $Pr[A(s_{i_1}, \ldots, s_{i_{k-1}}) = s]$ is negligible in m, which is the success probability of getting the secret s from any $k-1$ shadows $s_{i_1}, \ldots, s_{i_{k-1}}$.*

Remark: In other words, it is computational infeasible to invert the (k, n) secret sharing scheme from any $k-1$ shadows $s_{i_1}, \ldots, s_{i_{k-1}}$.

To achieve computational security, we need to have a computational hardness assumption. For some knowledge of computational security and AES in CBC mode, one can refer to [9, 15]. In this paper, we will use the following assumption:

Assumption 1 *It is computational infeasible to invert AES in CBC mode without the key.*

The proposed computational secure (k, n)-TSISS, also abbreviated as (k, n)-CSTSISS, contains two parts: *Construction 2*, which is the sharing program run by the dealer in the encoding phase; *Construction 3*, which is the revealing program run by the shadow holders in the decoding phase.

Construction 2

Input: *A secret image S.*
Output: *n shadows $(S_1, K_1, IV), (S_2, K_2, IV), \ldots, (S_n, K_n, IV)$.*
Step 1. *Pick up a random 128 bit key K and a random 128 bit initialization vector IV.*
Step 2. *Encrypt the secret image S by AES with key K and initialization vector IV in CBC mode. Each time we encrypt a 128 bit block, which contains 16 pixels for gray level image. The encrypted secret image is denoted as D.*

Step 3. *Fix* $f(x) = p_1 + p_2x + p_3x^2 + \ldots + p_kx^{k-1}$*, where* $p_1, p_2, \ldots, p_k$ *are* k *successive pixels. Then the pixel value for each shadow is calculated as* $q_i = f(i)$ *for* $i = 1, \ldots, n$*. Repeat the above process until all pixels of* D *have been shared. The shadows are denoted as* $S_1, S_2, \cdots, S_n$*.*

Step 4. *Pick up a random* $k-1$ *degree polynomial* $g(x) = a_1 + a_2x + a_3x^2 + a_4x^3 + \ldots + a_kx^{k-1}$ *with* $a_1 = K$ *and* $a_2, \ldots, a_k \overset{r}{\in} GF(2^{128})$*. Then the shadows are calculated by* $K_i = g(i)$ *for* $i = 1, \ldots, n$*.*

Step 5. *Participant* i *is distributed* (S_i, K_i, IV) *as his shadow, where* $i = 1, 2, \ldots, n$*.*

Remark: In Step 3., we use the (k, n)-VSSS, whose security is not guaranteed, and in Step 4., we use Shamir's (k, n)-SSS, which is information theoretic secure. The coefficient field that we use in Step 3. is $GF(2^8)$.

Construction 3

Input: *Any* k *shadows:* $(S_{i_1}, K_{i_1}, IV), \ldots, (S_{i_k}, K_{i_k}, IV)$*.*

Output: *A image* S*.*

Step 1. *Use* $K_{i_1}, \ldots, K_{i_k}$ *and Lagrange's interpolation to recover the constant term* K *of* $g(x)$*.*

Step 2. *Use* $S_{i_1}, \ldots, S_{i_k}$ *and Lagrange's interpolation to recover all coefficients* $p_1, p_2, \ldots, p_k$ *of* $f(x)$*. Repeat the above process until all pixels of* D *have been recovered.*

Step 3. *Decrypt* D *by AES in CBC mode with key* K *and initialization vector* IV*. The output is* S*.*

Theorem 3. *If Assumption 1 holds, then the proposed* (k, n)*-CSTSISS is computational secure.*

Proof: Assume we have an attack algorithm Adv that can recover the secret from some $k-1$ shadows generated by Construction 2. In other words, we have $Adv((S_{i_1}, K_{i_1}, IV), \ldots, (S_{i_{k-1}}, K_{i_{k-1}}, IV)) = S$. In the following, we give a new construction that can recover the plaintext from the ciphertext generated by AES in CBC mode by calling Adv as an oracle.

Construction 4

Input: *The ciphertext* (C, IV')*.*

Output: *The plaintext* S*.*

Step 1. *Fix* $f(x) = p_1 + p_2x + p_3x^2 + \ldots + p_kx^{k-1}$*, where* $p_1, p_2, \ldots, p_k$ *are* k *Bytes of* C*. The pixel value for each shadow is calculated as* $q_i = f(i)$ *for* $i = 1, \ldots, n$*. Repeat the above process until all Bytes of* C *have been shared. The shadows are denoted as* $S'_1, S'_2, \cdots, S'_n$*.*

Step 2. *Generate $k-1$ random key shadows of 128 bits:* $K'_1, K'_2, \cdots, K'_{k-1} \overset{r}{\in} GF(2^{128})$.

Step 3. *Feed* $((S'_1, K'_1, IV'), \ldots, (S'_{k-1}, K'_{k-1}, IV'))$ *into Adv, and output whatever Adv outputs.*

What we have to show is that *Adv* cannot distinguish $((S_1, K_1, IV), \ldots, (S_{k-1}, K_{k-1}, IV))$ from $((S'_1, K'_1, IV'), \ldots, (S'_{k-1}, K'_{k-1}, IV'))$, so that if *Adv* can decrypt the first ciphertext, it can decrypt the second ciphertext too. We first give the following Lemma.

Lemma 1. $(K_1, \ldots, K_{k-1})$ *and* $(K'_1, \ldots, K'_{k-1})$ *are equal in probability distribution.*

Proof of Lemma 1: Since $(K'_1, \ldots, K'_{k-1})$ is in uniform distribution, we only need to prove that $(K_1, \ldots, K_{k-1})$ is also in uniform distribution.

$(K_1, \ldots, K_k)$ are generated by the following equations:

$$\begin{aligned} a_1 + a_2 + a_3 + \ldots + a_k &= g(1) = K_1 \\ a_1 + a_3 \times 2 + a_3 \times 2^2 + \ldots + a_k \times 2^{k-1} &= g(2) = K_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots & \\ a_1 + a_3 \times k + a_3 \times k^2 + \ldots + a_k \times k^{k-1} &= g(k) = K_k \end{aligned}$$

Write the above equations in matrix form:

$$\begin{pmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & 2 & 2^2 & \ldots & 2^{k-1} \\ 1 & 3 & 3^2 & \ldots & 3^{k-1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & k & k^2 & \ldots & k^{k-1} \end{pmatrix} \begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} K_1 \\ K_2 \\ K_3 \\ \vdots \\ K_k \end{pmatrix}$$

Observe that the coefficient matrix is of *Van der Monde* (of full rank). Therefore, it can be taken as a 1-1 mapping from $(a_1, a_2, a_3, \ldots, a_k)$ to $(K_1, K_2, \ldots, K_k)$. Since $(a_1, a_2, a_3, \ldots, a_k)$ is in uniform distribution, $(K_1, K_2, \ldots, K_k)$ is also in uniform distribution and so is $(K_1, K_2, \ldots, K_{k-1})$. This is the end of the proof of Lemma 1.

Since Shamir's (k, n)-SSS is information theoretic secure, the $k-1$ shares $(K_1, K_2, \ldots, K_{k-1})$ are independent of the *AES* encryption key K of cipertext $(S_1, S_2, \ldots, S_{k-1})$. The $k-1$ randomly generated key shadows $(K'_1, K'_2, \ldots, K'_{k-1})$ are also independent of the *AES* encryption key of $(S'_1, S'_2, \ldots, S'_{k-1})$, say K',

which is unknown. In addition to this, IV and IV' are both in uniform distribution and independent from K and K' respectively. Hence Adv cannot distinguish $((S_1, K_1, IV), \ldots, (S_{k-1}, K_{k-1}, IV))$ from $((S'_1, K'_1, IV'), \ldots, (S'_{k-1}, K'_{k-1}, IV'))$.

In addition, Construction 4 is polynomial computable, which guarantees that if Adv is an efficient attack of the proposed (k, n)-CSTSISS, Construction 4 is also an efficient attack of AES in CBC mode by calling Adv as an oracle. The reason why Theorem 3 holds is that inverting AES in CBC mode without the key is computational infeasible, thus the assumed attack algorithm Adv cannot exist, otherwise we can attack AES in CBC mode efficiently. □

Remark: For some knowledge of probabilistic analysis, one can refer to [8].

Compared to previous (k, n)-TSISSs, the price we have payed out is $2\times128\times n$ bits more storage space in overall, and more time for the AES encryption and decryption processes.

The experimental results of the proposed $(2, 3)$-CSTSISS can be found in Figure 3, in which shadow images (b,c,d) are noise-like and provide no information about the content of the secret image (a). Compared to previous (k, n)-TSISSs, the proposed (k, n)-CSTSISS indeed does not have security defect for sharing textual images.

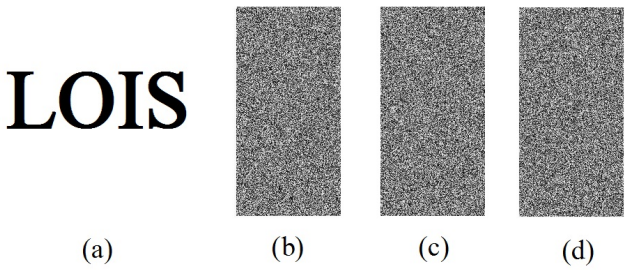

Fig. 3. Experimental results of the proposed $(2, 3)$-CSTSISS: (a) the original secret image with image size 300×300, (b) shadow 1 with image size 150×300, (c) shadow 2 with image size 150×300, (d) shadow 3 with image size 150×300

5 Conclusions

In this paper, we have shown that the (k, n)-VSSS, being extensively used in the studies of (k, n)-TSISS, does not have security guarantees. Furthermore, we have found those studies' security defect for sharing textual images and then patched up this security defect by adding an AES encryption process before the sharing process, which combines the beauty of small shadow size with computational security guarantee.

6 Acknowledgments

This work was supported by NSFC grant No. 60903210, the "Strategic Priority Research Program" of the Chinese Academy of Sciences No. XDA06010701 and the IIE's Projects No. Y1Z0011102, No. Y3Z001B102, No. Y2Z0011102 and No. Y3Z0071C02.

Bibliography

[1] G. Ateniese, C. Blundo, A. D. Santis, and D. Stinson. Visual cryptography for general access structures. In *Information and Computation*, volume 129, pages 86–106, 1996.

[2] L. Bai. A reliable (k,n) image secret sharing scheme. In *DASC*, pages 31–36, 2006.

[3] C. Chan and P. Sung. Secret image sharing with steganography and authentication using dynamic programming strategy. In *PCSPA*, pages 382–395, 2010.

[4] C. Chang, Y. Hsieh, and C. Lin. Sharing secrets in stego images with authentication. In *Pattern Recognition*, volume 41, pages 3130–3137, 2008.

[5] T. Cover and J. Thomas. Elements of information theory, 2nd edition. In *New York: Wiley-Interscience*, 2006.

[6] E. Elsheh and A. Hamza. Robust approaches to 3d object secret sharing. In *ICIAR*, volume LNCS 6111, pages 326–335, 2010.

[7] J. Feng, H. Wu, C. Tsai, and Y. Chu. A new multi-secret images sharing scheme using lagrange's interpolation. In *The Journal of Systems and Software*, volume 76, pages 327–339, 2005.

[8] O. Goldreich. Randomized methods in computation. In *Available at: http://www.wisdom.weizmann.ac.il/ oded/rnd.html*, 2001.

[9] S. Goldwasser and M. Bellare. Lecture notes on cryptography. In *Available at: http://cseweb.ucsd.edu/ mihir/papers/gb.pdf*, 2008.

[10] T. Guo, F. Liu, and C. Wu. Multi-pixel encryption visual cryptography. In *Inscrypt 2011*, volume LNCS 7537, pages 86–92, 2012.

[11] T. Guo, F. Liu, and C. Wu. On the equivalence of two definitions of visual cryptography scheme. In *ISPEC 2012*, volume LNCS 7232, pages 217–227, 2012.

[12] T. Guo, F. Liu, and C. Wu. Threshold visual secret sharing by random grids with improved contrast. In *The Journal of Systems and Software*, volume 86, pages 2094–2109, 2013.

[13] T. Guo, F. Liu, and C. Wu. k out of k extended visual cryptography scheme by random grids. In *Signal Processing*, volume 94, pages 90–101, 2014.

[14] C. Huang, C. Hsieh, and P. Huang. Progressive sharing for a secret image. In *The Journal of Systems and Software*, volume 83, pages 517–527, 2010.

[15] J. Katz and Y. Lindell. Introduction to modern cryptography. In *CRC PRESS*, 2007.

[16] P. Li, P. Ma, X. Su, and C. Yang. Improvements of a two-in-one image secret sharing scheme based on gray mixing model. In *Journal of Visual Communication and Image Representation*, volume 23, pages 441–453, 2012.

[17] F. Liu, T. Guo, C. Wu, and L. Qian. Improving the visual quality of size invariant visual cryptography scheme. In *Journal of Visual Communication and Image Representation*, volume 23, pages 331–342, 2012.

[18] M. Naor and A. Shamir. Visual cryptography. In *EUROCRYPT '94, Springer-Verlag Berlin*, volume LNCS 950, pages 1–12, 1995.

[19] A. Shamir. How to share a secret. In *Communications of the ACM*, volume 22 (11), pages 612–613, 1979.

[20] C. Thien and J. Lin. Secret image sharing. In *Computers and Graphics*, volume 26, pages 765–770, 2002.

[21] C. Thien and J. Lin. An image-sharing method with user-friendly shadow images. In *IEEE Transactions on Circuits and Systems for Video Technology*, volume 13, pages 1161–1169, 2003.

[22] M. Ulutas, G. Ulutas, and V. Nabiyev. Medical image security and epr hiding using shamir's secret sharing scheme. In *The Journal of Systems and Software*, volume 84, pages 341–353, 2011.

[23] R. Wang, Y. Chien, and Y. Lin. Scalable user-friendly image sharing. In *Journal of Visual Communication and Image Representation*, volume 21, pages 751–761, 2010.

[24] R. Wang and S. Shyu. Scalable secret image sharing. In *Signal Processing: Image Communication*, volume 22, pages 363–373, 2007.

[25] R. Wang and C. Su. Secret image sharing with smaller shadow images. In *Pattern Recognition Letters*, volume 27, pages 551–555, 2006.

[26] Y. Wu, C. Thien, and J. Lin. Sharing and hiding secret images with size constraint. In *Pattern Recognition*, volume 37, pages 1377–1385, 2004.

[27] C. Yang, T. Chen, K. Yu, and C. Wang. Improvements of image sharing with steganography and authentication. In *The Journal of Systems and Software*, volume 80, pages 1070–1076, 2007.

[28] C. Yang and C. Ciou. A comment on "sharing secrets in stegoimages with authentication". In *Pattern Recognition*, volume 42, pages 1615–1619, 2009.

[29] C. Yang and C. Ciou. Image secret sharing method with two-decoding-options: Lossless recovery and previewing capability. In *Image and Vision Computing*, volume 28, pages 1600–1610, 2010.

[30] C. Yang, Y. Huang, and J. Syue. Reversible secret image sharing based on shamir's scheme with discrete haar wavelet transform. In *ICECE*, volume 16-18, pages 1250–1253, 2011.

[31] C. Yang, J. Ouyang, and L. Harn. Steganography and authentication in image sharing without parity bits. In *Optics Communications*, volume 285, pages 1725–1735, 2012.

[32] R. Zhao, J. Zhao, F. Dai, and F. Zhao. A new image secret sharing scheme to identify cheaters. In *Computer Standards and Interfaces*, volume 31, pages 252–257, 2009.

Appendix

In the following, we will analyze the shadows generated by Construction 1 on two commonly used coefficient fields: $GF(251)$ and $GF(2^8)$. In computers, pure black pixel is represented by gray-level 0 and pure white pixel is represented by gray-level 255. If the t pixels being encoded are all pure black, then we get a zero polynomial $f(x) = 0$. In such a case, all share values are $f(i) = 0$ for $i = 1, 2, \ldots, n$. In other words, black area on the secret image will promise black on the corresponding area on shadows.

In order to make the encoding process feasible for coefficient field $GF(251)$, the pixels of gray-level larger than 250 are all set to gray-level 250. Although this method has some losses in gray-level degree, it is still the most widely used technique in secret image sharing. The reasons may be that this method is very easy to understand and implement, and the losses in visual quality are hardly noticeable. For (2, 3) threshold access structure, the sharing polynomial can be denoted by $f(x) = ax + b$. Each time, we encode two successive pixels by the coefficients "a, b". If the pixels are "0,0", the sharing polynomial is $f(x) = 0 \mod 251$, which results shares $f(1) = 0, f(2) = 0, f(3) = 0$. If the pixels are "250,250", the sharing polynomial is $f(x) = 250x + 250 \mod 251$, which results shares $f(1) = 249, f(2) = 248, f(3) = 247$. If the pixels are "250,0", the sharing polynomial is $f(x) = 250x \mod 251$, which results shares $f(1) = 250, f(2) = 249, f(3) = 248$. If the pixels are "0,250", the sharing polynomial is $f(x) = 250 \mod 251$, which results shares $f(1) = 250, f(2) = 250, f(3) = 250$. The above calculation shows that white areas ("250,250") and boundary areas ("250,0", "0,250") are encoded into almost white on each shadow while black areas ("0,0") are encoded into black on each shadow. Hence we conclude that the secret may be perceived from a single shadow for binary textual secret images. Please refer to images in Figure 1.

The operations in $GF(2^8)$ are a little more complicated than those in $GF(251)$. In the following, we will take some related examples to briefly introduce the operations in $GF(2^8)$, which are based on irreducible polynomial $\alpha^8+\alpha^4+\alpha^3+\alpha+1$.

$$\begin{aligned} 255 + 255 &= \{(\alpha^7 + \alpha^6 + \alpha^5 + \alpha^4 + \alpha^3 + \alpha^2 + \alpha + 1) + \\ &\qquad (\alpha^7 + \alpha^6 + \alpha^5 + \alpha^4 + \alpha^3 + \alpha^2 + \alpha + 1)\} \mod (\alpha^8 + \alpha^4 + \alpha^3 + \alpha + 1) \\ &= (00000000)_2 = 0 \end{aligned}$$

$$\begin{aligned}
255 \times 2 &= (\alpha^7+\alpha^6+\alpha^5+\alpha^4+\alpha^3+\alpha^2+\alpha+1)\times\alpha \mod (\alpha^8+\alpha^4+\alpha^3+\alpha+1)\\
&= (\alpha^8+\alpha^7+\alpha^6+\alpha^5+\alpha^4+\alpha^3+\alpha^2+\alpha) \mod (\alpha^8+\alpha^4+\alpha^3+\alpha+1)\\
&= (\alpha^8+\alpha^7+\alpha^6+\alpha^5+\alpha^4+\alpha^3+\alpha^2+\alpha)+(\alpha^8+\alpha^4+\alpha^3+\alpha+1)\\
&= \alpha^7+\alpha^6+\alpha^5+\alpha^2+1\\
&= (11100101)_2\\
&= 229
\end{aligned}$$

$$\begin{aligned}
255 \times 3 &= (\alpha^7+\alpha^6+\alpha^5+\alpha^4+\alpha^3+\alpha^2+\alpha+1)\times(\alpha+1) \mod (\alpha^8+\alpha^4+\alpha^3+\alpha+1)\\
&= (\alpha^7+\alpha^6+\alpha^5+\alpha^2+1)+(\alpha^7+\alpha^6+\alpha^5+\alpha^4+\alpha^3+\alpha^2+\alpha+1)\\
&= \alpha^4+\alpha^3+\alpha\\
&= (00011010)_2\\
&= 26
\end{aligned}$$

In [29], Yang et al. pointed out that it is better to use $GF(2^8)$ than to use $GF(251)$ as the coefficient field, since it has no loss in gray-level. For (2, 3) threshold access structure, the sharing polynomial can be denoted by $f(x) = ax + b$. Each time, we encode two successive pixels by the coefficients "a, b". If the pixels are "0,0", the sharing polynomial is $f(x) = 0$, which results shares $f(1) = 0, f(2) = 0, f(3) = 0$. If the pixels are "255,255", the sharing polynomial is $f(x) = 255x+255$, which results shares $f(1) = 0, f(2) = 26, f(3) = 229$. If the pixels are "255,0", the sharing polynomial is $f(x) = 255x$, which results shares $f(1) = 255, f(2) = 229, f(3) = 26$. If the pixels are "0,255", the sharing polynomial is $f(x) = 255$, which results shares $f(1) = 255, f(2) = 255, f(3) = 255$. The above calculation shows that for shadows 1 and 2, white areas ("255,255") and black areas ("0,0") are encoded into almost black and boundary areas ("250,0", "0,250") are encoded into almost white, while for shadow 3, white areas ("255,255") and boundary areas ("250,0", "0,250") are encoded into almost white and black areas ("0,0") are encoded into black. Hence we conclude that the secret may still be perceived from a single shadow for binary textual secret images, although some shadows may have inverted gray-levels. Please refer to images in Figure 2.