Information Processing Letters 115 (2015) 851–857

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Dynamic threshold secret reconstruction and its application to the threshold cryptography

Lein Harn [a], Ching-Fang Hsu [b,*]

[a] Department of Computer Science Electrical Engineering, University of Missouri-Kansas City, Kansas City, 64110, MO, USA
[b] Computer School, Central China Normal University, 430079, Wuhan, China

ARTICLE INFO

Article history:
Received 18 April 2014
Received in revised form 16 June 2015
Accepted 20 June 2015
Available online 26 June 2015
Communicated by S.M. Yiu

Keywords:
Cryptography
Secret sharing scheme
Bivariate polynomial
Secure channel
Dynamic threshold

ABSTRACT

Shamir's (t, n) secret sharing scheme (SS) is based on a univariate polynomial and is the most cited SS in the literature. The secret in a (t, n) SS can be recovered either by exactly t or more than t shareholders. Most SSs only consider when there are exactly t shareholders participated in the secret reconstruction. In this paper, we examine security issues related to the secret reconstruction if there are more than t shareholders participated in the secret reconstruction. We propose a dynamic threshold SS based on a bivariate polynomial in which shares generated by the dealer can be used to reconstruct the secret but having a larger threshold which is equivalent to the exact number of participated shareholders in the process. In addition, we extend the proposed scheme to enable shares which can also be used to establish pairwise keys to protect the reconstructed secret from non-shareholders. Shamir's SS has been used in conjunction with other public-key algorithms in most existing threshold algorithms. Our proposed SS can also be applied to the threshold cryptography to develop efficient threshold algorithms.

1. Introduction

The (t, n) secret sharing scheme (SS) was propose by Shamir [1] and Blakley [2] separately in 1979. In a (t, n) SS, the dealer divides the secret into n shares such that (a) the secret can be recovered if there are t or more than t shares, and (b) the secret cannot be recovered if there are fewer than t shares. The (t, n) SS can be implemented by different mathematical tools. For example, Shamir's scheme is based on a univariate polynomial, Blakely's scheme [1] is based on the geometry, Mignotte's scheme [3], Asmuth–Bloom's scheme [4] are based on the Chinese remainder theorem (CRT) and McEliece et al. scheme [5] is based on Reed–Solomon codes.

SS has become one of most popular cryptographic tools in many protocols of multi-party computing. The secret reconstruction of Shamir's SS is very simple and is based on the Lagrange interpolation formula. However, in the secret reconstruction, additional mechanisms are needed to protect the secret; otherwise, non-shareholders (i.e., outside attackers) or dishonest shareholders' (i.e., inside attackers) can take advantage over honest shareholders.

In 1985, Chor et al. [6] proposed the first verifiable secret sharing (VSS). Verifiability is the property of a VSS which allows shareholders to verify their shares. Invalid shares may be caused by the dealer during generation or by channel noise during transmission. VSS is executed by shareholders after receiving their shares from the dealer but before using their shares to reconstruct the secret. If VSS has detected/identified invalid shares, shareholders can request the dealer to regenerate new shares. There are vast research papers on the VSS in the literature. Based to

* Corresponding author.
E-mail addresses: harnl@umkc.edu (L. Harn), cherryjingfang@gmail.com (C.-F. Hsu).

http://dx.doi.org/10.1016/j.ipl.2015.06.014
0020-0190/

security assumptions, we can classify VSSs into two different types, schemes that are computationally secure and unconditionally secure. For example, Feldman [7] and Pedersen [8] developed non-interactive VSSs based on cryptographic commitment schemes. The security of Feldman's VSS is based on the hardness of solving discrete logarithm, while the privacy of Pedersen's VSS is unconditionally secure and the correctness of the shares depends on a computational assumption. Benaloh [9] proposed an interactive VSS scheme and it is unconditionally secure. Stinson et al. [10] proposed an unconditionally secure VSS and later, Patra et al. [11] proposed a generalized VSS scheme. In 1996, Stadler [12] proposed the first publicly verifiable secret sharing (PVSS) scheme. A PVSS scheme allows each shareholder to verify the validity of all shares, including both shares of his/her own and other shareholders. However, in most non-interactive VSSs [7,8], shareholders can only verify the validity of his/her own share; but not other shareholders' shares.

When shareholders present their shares in the secret reconstruction, dishonest shareholders (i.e., cheaters) can always exclusively derive the secret by presenting fake shares and thus the other honest shareholders get nothing but a fake secret. It is easy to see that Shamir's (t, n) secret sharing scheme does not prevent dishonest shareholders in the secret reconstruction. Cheater detection and identification are important functions in order to provide fair reconstruction of a secret. In 1989, Tompa and Woll [13] proposed the first cheater detection scheme. There are many research papers in the literature to propose algorithms for cheater detection and identification. Most of these algorithms [14–17] assume that there are exactly t shareholders participated in the secret reconstruction. The dealer needs to provide additional information to enable shareholders to detect and identify cheaters. Some algorithms [18,19] use error-correcting codes to detect and identify fake shares. In a recent paper, Harn and Lin [20] proposed a new approach to detect and identify cheaters. The algorithm uses shares to detect and identify cheaters. When there are more than t (i.e., the threshold) shares, for example j (i.e., $t < j$) shares in the secret reconstruction, the redundant shares can be used to detect and identify cheaters. In this approach, shares in a secret sharing scheme serve for two purposes; that are, (a) reconstructing the secret and (b) detecting and identifying cheaters. The detectability and identifiability of cheaters is proportional to the number of redundant shares in the secret reconstruction.

In this paper, we consider different security issues in the secret reconstruction. In particular, we examine problems if there are more than t shareholders participated in the secret reconstruction. We will discuss these problems in Section 3. Furthermore, we propose dynamic threshold SSs to overcome these problems. Our proposed SSs are based on a bivariate polynomial. Shares obtained from the dealer can serve for three different purposes, (a) reconstructing the secret, (b) reconstructing the secret having a dynamic threshold and (c) protecting exchange information in the secret reconstruction.

We summarize the contributions of our paper.

- A dynamic threshold SS based on a bivariate polynomial is proposed in which shares obtained from the dealer initially can be used to reconstruct the secret but having a larger threshold which is equivalent to the exact number of participants.
- An efficient (t, n) SS is proposed in which shares generated by the dealer can serve for three different purposes, (a) reconstructing the secret, (b) reconstructing the secret having a dynamic threshold and (c) protecting exchange information in the secret reconstruction.
- Our proposed SSs can be extended to the threshold cryptography to develop efficient threshold cryptographic algorithms (threshold signature/encryption).

The rest of paper is organized as follows. In Section 2, we review SSs based on polynomials. We discuss some security issues in the secret reconstruction in Section 3. A dynamic threshold SS and an efficient (t, n) SS based on a bivariate polynomial are proposed in Sections 4 and 5, respectively. The application of our proposed SS to the threshold cryptography is discussed in Section 6. The conclusion is given in Section 7.

2. Review of SSs based on polynomials

In Shamir's (t, n) SS [1], the dealer selects a univariate polynomial, $f(x)$, with degree $t-1$ and $f(0) = s$, where s is the secret. The dealer generates shares, $f(x_i)$, $i = 1, 2, \ldots, n$, for shareholders, where x_i is the public information associated with each shareholder, U_i. Each share, $f(x_i)$, is an integer in $GF(p)$. Shamir's (t, n) SS satisfies both security requirements of a (t, n) SS. That are, (a) with t or more than t shares can reconstruct the secret, and (b) with fewer than t shares cannot obtain any information of the secret. Shamir's SS is unconditionally secure.

Shamir's (t, n) SS does not provide the ability to allow shareholders to verify their shares obtained from the dealer. In 1985, Chor et al. [14] extended the notion of SS and proposed the first verifiable secret sharing (VSS). Verifiability is the property of a VSS which allows shareholders to verify their shares. Invalid shares may be caused by the dealer during generation or by channel noise during transmission. VSS is executed by shareholders after receiving their shares from the dealer but before using their shares to reconstruct the secret. If VSS has detected/identified invalid shares, shareholders can request the dealer to regenerate new shares. There are many (t, n) VSSs [21–27] based on bivariate polynomials, denoted them as BVSSs. A bivariate polynomial with degree $t-1$ is represented as $F(x, y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{2,0}x^2 + a_{1,1}xy + a_{0,2}y^2 + \ldots + a_{t-1,0}x^{t-1} + a_{t-2,1}x^{t-2}y + \ldots + a_{0,t-1}y^{t-1} \bmod p$, where $a_{i,j} \in GF(p)$, $\forall i, j \in [0, t-1]$. If the coefficients satisfy $a_{i,j} = a_{j,i}$, $\forall i, j \in [0, t-1]$, it is a symmetric bivariate polynomial. Shares generated by a bivariate polynomial enable pairwise keys to be shared between any pair of shareholders. We can classify BVSSs into two types, the asymmetric BVSSs, denoted them as ABVSSs [21,22,24,26] and the symmetric BVSSs, denoted them as SBVSSs [24–27]. In all existing (t, n) SBVSSs, the dealer selects a bivariate polynomial,

$F(x, y)$, with degree $t-1$ and $F(0,0) = s$, where s is the secret. The dealer generates shares, $F(x_i, y)$, $i = 1, 2, \ldots, n$, for shareholders, where x_i is the public information associated with each shareholder, U_i. Each share, $F(x_i, y)$, is a univariate polynomial with degree $t-1$. Note that shares generated in an SBVSS satisfy $F(x_i, x_j) = F(x_j, x_i)$, $\forall i, j \in [0, t-1]$, the pairwise key, $F(x_i, x_j) = F(x_j, x_i)$, can be shared between the pair of shareholders, U_i and U_j. In a similar way, in an ABVSS, the dealer generates a pair of shares, $F(x_i, y)$ and $F(x, x_i)$, $i = 1, 2, \ldots, n$, for each shareholder and the pairwise secret key, $F(x_i, x_j)$ or $F(x_j, x_i)$, can also be shared between the pair of shareholders, U_i and U_j.

3. Security issues in the secret reconstruction

In a (t, n) SS, the secret can be recovered if there are exactly t or more than t shares. Most existing SSs only consider when there are exactly t shareholders participated in the secret reconstruction. In this paper, we examine security issues related to the secret reconstruction when there are more than t shareholders participated in the secret reconstruction.

If there are more than t shareholders participated in the secret reconstruction, one straightforward solution is to select exactly t shareholders to actively act in the process. Since non-shareholders may impersonate to be shareholders participated in the secret reconstruction to gain the access of the secret, employing a user authentication or a VSS scheme can prevent this security threat. However, all user authentication schemes or VSSs verify shareholders/shares one at a time. It is a time-consuming process to verify all shareholders/shares. There is one more serious problem following this approach. That is, the reconstructed secret is only limited to t shareholders but not to all participated shareholders. In some applications, it is necessary to let all participated shareholders know the secret. This limitation makes this solution undesirable.

One alternative solution is to require that all participants actively act in the secret reconstruction and the recovered secret is shared among all participated shareholders. In this solution, every shareholder needs to compute a value using his/her share and send this value to all other participants. The reconstructed secret is computed using all values of participants. Shamir's secret reconstruction scheme can be generalized to take more than t shares. For example, when there are j (i.e., $t < j \leq n$) shareholders with their shares, $\{f(x_1), f(x_2), \ldots, f(x_j)\}$, participated in the secret reconstruction, the secret can be computed as $s = f(0) = \sum_{r=1}^{j} f(x_r) \prod_{v=1, v\neq r}^{j} \frac{-x_v}{x_r - x_v} \bmod p$. In this generalization, each participant needs to contribute his/her share in the secret reconstruction. However, this generalization cannot prevent an attacker from obtaining the secret. This is because the threshold of shares is t. The attacker needs only t shares to recover the secret. Again, employing a user authentication or a VSS scheme to ensure that all participants are shareholders can prevent this security threat. But, this will add additional communication and computational complexity to the secret reconstruction process.

Employing a *threshold changeable SS* (TCSS) in which shares generated by the dealer initially can be used to reconstruct the secret but having a larger threshold j (i.e., $t < j$) which is equivalent to the number of participants is an alternative solution. In 1999, Martin et al. [28] proposed the first TCSS. TCSSs can be classified into three types, schemes based on a linear polynomial [29,30], schemes based on the geometry [31], and schemes based on the CRT [32,33]. Since standard Shamir's SS is very simple and is unconditionally secure, most efforts have been devoted to propose TCSSs [29,34] to support standard Shamir's SS. Most TCSSs are interactive and need secure channels to refresh new shares. In 2004, Steinfeld et al. [29] proposed a Lattice-based TCSS to support standard Shamir's secret generation algorithm. Their scheme does not need any secure channels and is called a ***TCSS without dealer***. However, their scheme cannot use standard Shamir's secret reconstruction to recover the secret. Recently, a dealer-free TCSS is proposed by Nojoumian et al. [35] and a collusion attack resistance TCSS is proposed by Zhang et al. [36]; but both TCSSs need interactions among shareholders in the secret reconstruction. In the next section, we propose a dynamic threshold SS based on a bivariate polynomial. In our proposed scheme, shares generated by the dealer initially can be used to reconstruct the secret but having a larger threshold which is equivalent to the number of participants. Our proposed scheme is dealer-free and non-interactive. Furthermore, our proposed scheme can support standard Shamir's SS and secret reconstruction.

The shares released by participated shareholders in the secret reconstruction need to be protected by communication keys; otherwise, non-shareholders can also obtain the secret. The key establishment protocol is used to establish secret keys for shareholders. However, adding a key establishment protocol in the secret reconstruction can slow down the secret reconstruction process. In Section 5, we extend our proposed dynamic threshold SS to enable shares generated by the dealer initially can serve for three different purposes, (a) reconstructing the secret, (b) reconstructing the secret having a dynamic threshold and (c) protecting exchange information in the secret reconstruction.

4. Proposed dynamic threshold SS based on a bivariate polynomial

There is one major difference between shares generated by using a univariate polynomial and using a bivariate polynomial. The shares generated by a univariate polynomial are integers in $GF(p)$; but shares generated by a bivariate polynomial are univariate polynomials with degree $t-1$. In this section, we introduce a dynamic threshold SS, in which shares generated initially by the dealer having the threshold t can be used to reconstruct a secret having a larger threshold j, with $t < j \leq 1 + \frac{t(t+1)}{2}$ (we will provide detail discussion on the upper bound of this dynamic range in Section 4.2.1). Our proposed scheme is dealer-free and non-interactive. In other words, depending on the number of participated shareholders, j, each participated shareholder uses his/her share obtained from the

dealer initially which has the threshold t to compute and released a value to all other shareholders. It needs exactly j released values in order to recover the secret; otherwise, no information of the secret is revealed.

4.1. Algorithm

Share generation:

The dealer selects a $t-1$ degree symmetric polynomial,

$$F(x,y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{2,0}x^2 + a_{1,1}xy + a_{0,2}y^2 + \ldots + a_{t-1,0}x^{t-1} + a_{t-2,1}x^{t-2}y + \ldots + a_{0,t-1}y^{t-1} \bmod p,$$

where $a_{i,j} \in Z_p$, $a_{i,j} = a_{j,i}$, $\forall i, j \in [0, t-1]$, and the secret $s \in Z_p$ satisfies $s = F(0,0) + bF(1,1)$, where $b \in Z_p$. The dealer computes shares, $s_i(y) = F(x_i, y) \bmod p$, for shareholders, U_i, $i = 1, 2, \ldots, n$, where $x_i \notin \{0, 1\}$ is the public information associated with each shareholder, U_i. The dealer sends each share, $s_i(y)$, to shareholder U_i secretly.

Secret reconstruction:

When j (i.e., $t < j \le 1 + \frac{t(t+1)}{2}$) shareholders, for example $\{U_{v_1}, U_{v_2}, \ldots, U_{v_j}\}$, want to recover the secret, each shareholder, U_{v_i}, accesses the public information, b, v and uses his/her shares, $s_{v_i}(y)$, to execute the following steps.

Step 1.
Each shareholder U_{v_i} uses his share, $s_{v_i}(y)$, to compute

$$w_{v_i} = s_{v_i}(0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p.$$

w_{v_i} is sent to other shareholders.

Step 2.
After receiving w_{v_i}, $i = 1, 2, \ldots, j$, each shareholder computes $s = \sum_{i=1}^{j} w_{v_i} \bmod p$.

Remark 1. We want to point out that the upper bound of the threshold range should be $1 + \frac{t(t+1)}{2} \le n$, where n is the total number of shares generated by the dealer; otherwise, if $1 + \frac{t(t+1)}{2} > n$, the secret can never be reconstructed.

4.2. Correctness of the secret reconstruction

Theorem 1. *The secret can be reconstructed as* $s = \sum_{i=1}^{j} w_{v_i} \times \bmod p$.

Proof. If all j shareholders act honestly to compute and release values, $w_{v_i} = s_{v_i}(0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \times \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p$, $i = 1, 2, \ldots, j$, then we have

$$\begin{aligned} s &= \sum_{i=1}^{j} w_{v_i} \bmod p \\ &= \sum_{i=1}^{j} s_{v_i}(0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + b \sum_{i=1}^{j} s_{v_i}(1) \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p \\ &= \sum_{i=1}^{j} F(x_{v_i}, 0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + b \sum_{i=1}^{j} F(x_{v_i}, 1) \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p \\ &= F(0,0) + bF(1,1) \\ &= s. \qquad \square \end{aligned}$$

4.2.1. Secrecy of shares

In the following discussion, we analyze the secrecy of shares and the secrecy of the secret in different subsections. In our proposed scheme, each shareholder has one share which is a univariate polynomial having $t-1$ degree. Since each released value, w_{v_i}, is a linear combination of two values, $s_{v_i}(0)$ and $bs_{v_i}(1)$, of each share, we want to show that the secrecy of the shares cannot be obtained from either a single released value or multiple released values; otherwise, fewer than j shares can recover the secret which violates the security requirement.

It is obvious that it is impossible to directly solve the two values, $s_{v_i}(0)$ and $bs_{v_i}(1)$, from each released value, $w_{v_i} = s_{v_i}(0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p$. In the following discussion, we consider the secrecy of shares from multiple released values. Since the new threshold, j, is limited to be $t < j \le 1 + \frac{t(t+1)}{2}$, in the following discussion, we analyze the secrecy of shares if the new threshold is $k = 1 + \frac{t(t+1)}{2}$. We want to show that the shares cannot be recovered from $k-1$ released values; otherwise, fewer than the threshold can reconstruct the secret. If the secrecy of shares is satisfied for this threshold, the secrecy of shares for other thresholds can also be satisfied. Since each released value, $w_{v_i} = s_{v_i}(0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p = F(x_{v_i}, 0) \prod_{l=1, l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bF(x_{v_i}, 1) \prod_{l=1, l\neq i}^{j} \frac{1 - x_{v_l}}{x_{v_i} - x_{v_l}} \bmod p$, is a linear function of $\frac{t(t+1)}{2}$ coefficients of the symmetric bivariate polynomial $F(x,y) = a_{0,0} + a_{1,0}x + a_{0,1}y + a_{2,0}x^2 + a_{1,1}xy + a_{0,2}y^2 + \ldots + a_{t-1,0}x^{t-1} + a_{t-2,1}x^{t-2}y + \ldots + a_{0,t-1}y^{t-1} \bmod p$, $k-1$ equations can be established from $k-1$ released values. The condition, $t < j \le 1 + \frac{t(t+1)}{2}$, implies that $\frac{t(t+1)}{2} > k-1$. Therefore, it is impossible to solve the polynomial, $f_l(x)$, from these $k-1$ equations. The secrecy of shares cannot be recovered from released values in the secret reconstruction.

4.2.2. Secrecy of the secret

Since $s = \sum_{i=1}^{j} w_{v_i} mod\, p$, it is obvious that it needs w_{v_i}, $i = 1, 2, \ldots, j$, to obtain the secret and fewer than j values of w_{v_i}, cannot recover the secret.

The secret, $s = F(0,0) + bF(1,1)$, is a linear combination of $\frac{t(t+1)}{2}$ coefficients of the symmetric bivariate polynomial, $F(x,y)$. For any threshold, j (i.e., $t < j \leq 1 + \frac{t(t+1)}{2}$), if there are $j-1$ released values, $w_{v_i} = s_{v_i}(0) \prod_{l=1,l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1,l\neq i}^{j} \frac{1-x_{v_l}}{x_{v_i} - x_{v_l}} mod\, p$, from Section 4.2.1, we show that it is impossible to solve the polynomial, $F(x,y)$. The security of the proposed scheme is unconditionally secure.

Remark 2. In Shamir's (t,n) SS, each shareholder has only one share; but, in our proposed TCSS, the number of shares of each shareholder is expanded by a factor of t. However, shares in our proposed scheme can be used to reconstruct the secret but having a different threshold; but shares in Shamir's (t,n) SS can only be used to reconstruct the secret having the original threshold. One straightforward approach to implement a TCSS is that during share generation, the dealer generates multiple shares for each shareholder and each share can be used to reconstruct the secret having a distinct threshold varying from t to $1 + \frac{t(t+1)}{2}$. The number of shares for each shareholder is $2 + \frac{t(t-1)}{2}$ using this approach; but the number of shares for each shareholder is t using our proposed dynamic threshold SS.

4.3. Performance

Each share, $s_i(y)$, is a univariate polynomial with degree $t-1$ and the shareholder needs to store t coefficients of the polynomial. Thus, the memory storage of each shareholder is $t \log_2 p$ bits, where p is the modulus. Horner's rule [37] can be used to evaluate polynomials. In the following discussion, we show the cost for computing $w_{v_i} = s_{v_i}(0) \prod_{l=1,l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1,l\neq i}^{j} \frac{1-x_{v_l}}{x_{v_i} - x_{v_l}} mod\, p$, in the secret reconstruction. From Horner's rule, evaluating a polynomial of degree $t-1$ needs $t-1$ multiplications and t additions. Since each multiplication takes more time than each addition, the performance is only addressed to the number of multiplication needed. The computational cost in Step 1 to compute w_{v_i} consists of only the cost of evaluating one polynomial. In addition, the computational cost in Step 2 to compute, $s = \sum_{i=1}^{j} w_{v_i} mod\, p$, needs only additions. Overall, the computational cost to reconstruct the secret, each shareholder needs to evaluate $t + 1 + 2(j-2)$ multiplications.

The proposed scheme needs to employ additional key establishment algorithm to establish secret communication keys to protect w_{v_i} in order to prevent non-shareholders obtain the secret. In the next section, we extend this scheme to use same shares to establish pairwise keys between every two users. Shareholder can use these pairwise keys to protect w_{v_i}.

5. An efficient (t,n) SS based on a bivariate polynomial

In this section, we propose a (t,n) SS in which shares of shareholders cannot only be used to reconstruct a secret but also to protect the secrecy of the recovered secret. The proposed scheme keeps the recovered secret to be known only to shareholders but not to non-shareholders. The proposed scheme is unconditionally secure.

5.1. Algorithm

Share generation:

The dealer follows the same procedures as described in Section 4.1 to generate shares of shareholders.

Secret reconstruction:

Assume j (i.e., $t < j \leq 1 + \frac{t(t+1)}{2}$) shareholders, for example $\{U_{v_1}, U_{v_2}, \ldots, U_{v_j}\}$, want to recover the secret, s.

Step 1.
Each shareholder U_{v_i} uses his share, $s_{v_i}(y)$, to compute

$$w_{v_i} = s_{v_i}(0) \prod_{l=1,l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1,l\neq i}^{j} \frac{1-x_{v_l}}{x_{v_i} - x_{v_l}} mod\, p.w_{v_i}$$

is revealed to other shareholders.

Step 2.
Each shareholder U_{v_i} uses his share, $s_{v_i}(y)$, to compute pairwise shared keys, $k_{i,j} = s_{v_i}(x_{v_j}) = F(x_{v_i}, x_{v_j})$, $j = 1, 2, \ldots, u$, $j \neq i$, where $k_{i,j}$ is the secret key shared between shareholders, U_{v_i} and U_{v_j}.

Step 3.
Each shareholder U_{v_i} sends w_{v_i} secretly to other shareholders as $c_{i,j} = E_{k_{i,j}}(w_{v_i})$, $j = 1, 2, \ldots, u$, $j \neq i$, where $E_{k_{i,j}}(w_{v_i})$ is the encryption of w_{v_i} using the key $k_{i,j}$.

Step 4.
After receiving ciphertext, $c_{j,i}$, $j = 1, 2, \ldots, u$, $j \neq i$, from other shareholders, shareholder U_{v_i} computes $w_{v_j} = D_{k_{i,j}}(c_{j,i})$, $j = 1, 2, \ldots, u$, $j \neq i$, where $D_{k_{i,j}}(c_{j,i})$ is the decryption of $c_{j,i}$ using the key $k_{i,j}$.

Step 5.
After recovering, w_{v_i}, $i = 1, 2, \ldots, j$, each shareholder computes $s = \sum_{i=1}^{j} w_{v_i} mod\, p$.

5.2. Security

In the secret reconstruction, in Step 1, each released value,

$$w_{v_i} = s_{v_i}(0) \prod_{l=1,l\neq i}^{j} \frac{-x_{v_l}}{x_{v_i} - x_{v_l}} + bs_{v_i}(1) \prod_{l=1,l\neq i}^{j} \frac{1-x_{v_l}}{x_{v_i} - x_{v_l}} mod\, p,$$

is an output of a valid share, $s_i(y)$, of shareholder U_{v_i}. In addition, in Step 2, each pairwise key, $k_{i,j} = s_{v_i}(x_{v_j}) = F(x_{v_i}, x_{v_j})$, is also an output of a valid share, $s_i(y)$, of shareholder U_{v_i}. Since non-shareholders do not own any valid share, so non-shareholders cannot recover the secret from the exchange information, $c_{i,j} = E_{k_{i,j}}(w_{v_i})$, in Step 3. In other words, this proposed scheme is able to protect the secret in the secret reconstruction. The security of the proposed scheme is unconditionally secure.

6. Application to the threshold cryptography

In Shamir's SS, the shares can be used to reconstruct only one secret. This is because the secret and shares are known to all participated shareholders in the secret reconstruction. Thus, the efficiency of the SS is very low. The threshold cryptography was first introduced by Desmedt in 1987 [38]. Threshold cryptography is the study of efficient multiparty computation protocols for cryptographic functions (e.g. signing or decrypting) in which each group member has a share of the private key of the group and multiple members jointly compute an output of the cryptographic function. Shamir's (t, n) SS has been used in conjunction with public-key algorithms, such as RSA scheme [39] or ElGamal scheme [40], in most existing threshold algorithms. For example, [41,42] are based on the ElGamal scheme, [43–46] are based on the RSA, [47,48] are based on the Elliptic Curve public-key scheme and [49,50] are based on Pairing. Since shares are protected by public-key algorithms in threshold cryptography, shares can be reused to compute multiple functions.

In the processing to compute any threshold function, values computed by group members need to be protected by communication keys; otherwise, non-members can also obtain the output. For example, in a threshold decryption, if computed values of group members are not protected, non-members can also recover the plaintext. Most existing threshold algorithms use Shamir's (t, n) SS as the building block to generate shares of group members. Shares generated by Shamir's SS can only be used to recover the secret. Thus, additional key establishment protocol is needed to generate communication keys for group members. Adding a key establishment protocol in the threshold application can slow down the process significantly. As we have shown in previous sections, shares generated by a bivariate polynomial can be used to establish pairwise keys between any pair of group members. Furthermore, shares can also be used to reconstruct the secret but having a dynamic threshold. By extending our proposed SSs to threshold algorithms can improve their efficiency significantly.

In a threshold algorithm, the group manager (GM) is responsible to select a pair of public and private keys of the group and to register group members initially. The GM follows our proposed SS in Section 5 to select the private key of the group as the secret and generate shares of group members. The share of each group member is a univariate polynomial with degree $t - 1$. In the process to compute the threshold function, each participated group member uses his/her share to compute an individual output of the function and pairwise keys shared with other group members. Then, the individual output is encrypted using the pairwise key shared with every other group member separately. The ciphertext is sent to every participated member. Similarly, each received ciphertext needs to be decrypted using the pairwise key shared with every other group member to recover each individual output. By combining all individual outputs of participated members can obtain an output of the threshold function. Since non-members do not have any share generated by the GM, the output of threshold function is prevented from non-members.

7. Conclusion

Shamir's SS is one of the most popular cryptographic tools. However, there are many security issues related to the secret reconstruction process. In this paper, we pay special attention to address how to reconstruct the secret when there are more than t participants in the secret reconstruction. We propose a dynamic threshold SS in which shares generated by the dealer initially can be used to reconstruct a secret but having a larger threshold (i.e., the threshold is determined by the number of participated shareholders in the secret reconstruction). Our proposed schemes are dealer-free and non-interactive. The dynamic threshold SS can be extended to enable shares of shareholders to establish pairwise keys used to protect the recovered secret. These proposed SSs can also be applied to the threshold cryptography to develop efficient threshold algorithms.

Acknowledgements

This work was supported by the self-determined research funds of CCNU from the colleges' basic research and operation of MOE, under grant CCNU15ZD003 and CCNU15A02018, and the major project of National Social Science Fund, under grant 12&2D223.

References

[1] A. Shamir, How to share a secret, Commun. ACM 22 (11) (1979) 612–613.
[2] G.R. Blakley, Safeguarding cryptographic keys, in: Proceedings of AFIPS'79 Nat. Computer Conf., vol. 48, AFIPS Press, 1979, pp. 313–317.
[3] M. Mignotte, How to share a secret, in: Cryptography-Proceedings of the Workshop on Cryptography, in: Lecture Notes in Computer Science, vol. 149, Springer-Verlag, 1983, pp. 371–375.
[4] C.A. Azimuth, J. Bloom, A modular approach to key safeguarding, IEEE Trans. Inf. Theory IT-29 (2) (1983) 208–210.
[5] R.J. McEliece, D.V. Sarwate, On sharing secrets and Reed–Solomon codes, Commun. ACM 24 (9) (1981) 583–584.
[6] B. Chor, S. Goldwasser, S. Micali, B. Awerbuch, Verifiable secret sharing and achieving simultaneity in the presence of faults, in: Proceedings of the 26th IEEE Symposium on the Foundations of Computer Science, Oregon, Portland, 1985, pp. 383–395.
[7] P. Feldman, A practical scheme for non-interactive verifiable secret sharing, in: Proceedings of the 28th IEEE Symposium on Foundations of Computer Science, 27–29 October, IEEE Computer Society, Los Angeles, California, 1987, pp. 427–437.
[8] T.P. Pedersen, Non-interactive and information-theoretic secure verifiable secret sharing, in: Advances in Cryptology, CRYPTO'91, in: Lecture Notes in Computer Science, vol. 576, Springer-Verlag, 1992, pp. 129–140.
[9] J.C. Benaloh, Secret sharing homomorphisms: keeping shares of a secret, in: Advances in Cryptology, CRYPTO'86, in: Lecture Notes in Computer Science, vol. 263, Springer-Verlag, 1987, pp. 251–260.

[10] D.R. Stinson, R. Wei, Unconditionally secure proactive SS with combinatorial structures, in: Proceedings of SAC'99, in: Lecture Notes in Computer Science, vol. 1758, Springer-Verlag, 2000, pp. 200–214.
[11] A. Patra, A. Choudhary, C.P. Rangan, Efficient statistical asynchronous verifiable secret sharing with optimal resilience, in: Proceedings of ICITS'09, in: Lecture Notes in Computer Science, vol. 5973, Springer-Verlag, 2010, pp. 74–92.
[12] M. Stadler, Publicly verifiable secret sharing, in: Advances in Cryptology, EUROCRYPT'96, in: Lecture Notes in Computer Science, vol. 1070, Springer-Verlag, 1996, pp. 190–199.
[13] M. Tompa, H. Wol, How to share a secret with cheaters, J. Cryptol. 1 (3) (1989) 133–138.
[14] T. Rabin, M. Ben-Or, Verifiable secret sharing and multiparty protocols with honest majority, in: Proceedings of the 21st Annual ACM Symposium on the Theory of Computing, 1989, pp. 73–85.
[15] M. Carpentieri, A. De Santis, U. Vaccaro, Size of shares and probability of cheating in threshold schemes, in: Advances in Cryptology, EUROCRYPT'93, in: LNCS, vol. 765, Springer-Verlag, 1994, pp. 118–125.
[16] K. Kurosawa, S. Obana, W. Ogata, t-Cheater identifiable (k, n) secret sharing schemes, in: Advances in Cryptology, CRYPTO'95, in: LNCS, vol. 963, Springer-Verlag, 1995, pp. 410–423.
[17] J. He, E. Dawson, Shared secret reconstruction, Des. Codes Cryptogr. 14 (3) (1998) 221–237.
[18] R.J. McEliece, D.V. Sarwate, On sharing secrets and Reed–Solomon codes, Commun. ACM 24 (9) (1981) 583–584.
[19] C. Blundo, A. De Santis, L. Gargano, U. Vaccaro, Secret sharing schemes with veto capabilities, in: Proceedings of the First French–Israeli Workshop on Algebraic Coding, in: LNCS, vol. 781, Springer-Verlag, 1993, pp. 82–89.
[20] L. Harn, C. Lin, Detection and identification of cheaters in (t, n) secret sharing scheme, Des. Codes Cryptogr. 52 (1) (2009) 15–24.
[21] R. Cramer, I. Damgard, S. Dziembowski, M. Hirt, T. Rabin, Efficient multiparty computations secure against an adaptive adversary, in: Proceedings of 18th Annual IACR EUROCRYPT, Prague, Czech Republic, in: LNCS, vol. 1592, Springer, 1999, pp. 311–326.
[22] M. Fitzi, J. Garay, S. Gollakota, C. Pandu Rangan, K. Srinathan, Round-optimal and efficient verifiable secret sharing, in: S. Halevi, T. Rabin (Eds.), Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, 4–7 March, 2006, in: LNCS, vol. 3876, Springer, 2006, pp. 329–342.
[23] R. Gennaro, Y. Ishai, E. Kushilevitz, T. Rabin, The round complexity of verifiable secret sharing and secure multicast, in: STOC, 2001, pp. 580–589.
[24] J. Katz, C. Koo, R. Kumaresan, Improved the round complexity of VSS in point-to point networks, in: Proceedings of ICALP'08, Part II, in: LNCS, vol. 5126, Springer, 2008, pp. 499–510.
[25] R. Kumaresan, A. Patra, C.P. Rangan, The round complexity of verifiable secret sharing: the statistical case, in: Advances in Cryptology, ASIACRYPT 2010, in: LNCS, vol. 6477, Springer, 2010, pp. 431–447.
[26] V. Nikov, S. Nikova, On Proactive Secret Sharing Schemes, LNCS, vol. 3357, Springer, 2005, pp. 308–325.
[27] A. Patra, A. Choudhary, T. Rabin, C.P. Rangan, The round complexity of verifiable secret sharing revisited, in: Advances in Cryptology, Proceedings of the Crypto'09, 16–20 August, Santa Barbara, CA, USA, in: LNCS, vol. 5677, Springer, 2009, pp. 487–504.
[28] K. Martin, J. Pieprzyk, R. Safavi-Naini, H. Wang, Changing thresholds in the absence of secure channels, J. Aust. Comput. 31 (1999) 34–43.
[29] R. Steinfeld, H. Wang, J. Pieprzyk, Lattice-based threshold changeability for standard Shamir secret-sharing schemes, in: Advances in Cryptology, ASIACRYPT'04, in: Lecture Notes in Computer Science, vol. 3329, 2004, pp. 170–186.
[30] Z. Zhang, Y.M. Chee, S. Ling, M. Liu, H. Wang, Threshold changeable secret sharing schemes revisited, Theor. Comput. Sci. 418 (2012) 106–115.
[31] K.M. Martin, J. Pieprzyk, R. Safavi-Nain, H. Wang, Changing thresholds in the absence of secure channels, in: Proceedings of ACISP'99, in: Lecture Notes in Computer Science, vol. 1587, 1999, pp. 177–191.
[32] T. Lou, C. Tartary, Analysis and design of multiple threshold changeable secret sharing, in: Proceedings of CANS'08, in: Lecture Notes in Computer Science, vol. 5339, Springer-Verlag, 2008, pp. 196–213.
[33] R. Steinfeld, J. Pieprzyk, H. Wang, Lattice-based threshold changeability for standard CRT secret-sharing schemes, Finite Fields Appl. 2 (2006) 653–680.
[34] C. Tartary, H. Wang, Dynamic threshold and cheater resistance for Shamir SS, in: Proceedings of INSCRYPT'06, in: Lecture Notes in Computer Science, vol. 4318, 2006, pp. 103–117.
[35] M. Nojoumian, D.R. Stinson, Dealer-free threshold changeability in secret sharing schemes, Adv. Math. Commun. 7 (1) (2013) 39–56.
[36] X. Zhang, M. He, Collusion attack resistance and practice-oriented threshold changeable secret sharing schemes, in: Proc. 2010 24th IEEE International Conference on Advanced Information Networking and Applications, 2010, pp. 745–752.
[37] D.E. Knuth, The Art of Computer Programming, Semi-Numerical Algorithms, vol. II, Addison–Wesley, Reading, Massuchussetts, 1981.
[38] Y. Desmedt, Society and group oriented cryptography: an new concept, in: Advances in Cryptography, CRYPTO'87, in: LNCS, vol. 293, Springer-Verlag, 1987, pp. 120–127.
[39] R. Rivest, A. Shamir, L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Commun. ACM 21 (2) (1978) 120–126.
[40] T.A. ElGamal, A public-key cryptosystem and a signature scheme based on discrete logarithms, IEEE Trans. Inf. Theory 31 (4) (1985) 469–472.
[41] Y. Desmedt, Y. Frankel, Threshold cryptosystems, in: Advances in Cryptology, Crypto'89, 1989, pp. 307–315.
[42] L. Harn, Group-oriented (t, n) threshold digital signature scheme and digital multisignature, IEE Proc., Comput. Digit. Tech. 141 (5) (Sep. 1994) 307–313.
[43] Y. Desmedt, Y. Frankel, Shared generation of authenticators and signatures, in: Advances in Cryptology, Crypto'91, 1991, pp. 457–569.
[44] A. De Santis, Y. Desmedt, Y. Frankel, M. Yung, How to share a function securely, in: 26th Annual ACM Symposium on Theory of Computing, 1994, pp. 522–533.
[45] R. Gennaro, S. Jarecki, H. Krawczyk, T. Rabin, Robust and efficient sharing of RSA functions, in: Advances in Cryptology, Crypto'96, 1996, pp. 157–172.
[46] V. Shoup, Practical threshold signatures, in: Advances in Cryptology, Eurocrypt 2000, 2000, pp. 207–220.
[47] L. Ertaul, W. Lu, ECC based threshold cryptography for secure data forwarding and secure key exchange in MANET (I), in: Networking 2005, in: LNCS, vol. 3462, 2005, pp. 102–113.
[48] Y. Shang, X. Wang, Y. Li, Y. Zhang, A general threshold signature scheme based on Elliptic Curve, in: Proceedings of the 2012 2nd International Conference on Computer and Information Application, ICCIA, 2012.
[49] W. Gao, G. Wang, X. Wang, Z. Yang, One-round ID-based threshold signature scheme from bilinear pairings, Informatica 20 (4) (2009) 461–476.
[50] W. Gao, G. Wang, X. Wang, Z. Yang, One-round ID-based threshold signature scheme from bilinear pairings, Informatica 20 (4) (2009) 461–476.