Published online：2016-10-13 09:09:58
URL：http://www.cnki.net/kcms/detail/10.1284.TN.20161013.0909.010.html

GOMSS: A Simple Group Oriented (t, m, n) Multi-secret Sharing Scheme*

MIAO Fuyou, WANG Li, JI Yangyang and XIONG Yan

(*School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027,China*)

Abstract — In most (t, n)-multi-secret sharing ((t, n)-MSS) schemes, an illegal participant, even without any valid share, may recover secrets when there are over t participants in secret reconstructions. To address this problem, the paper presents the notion of Group oriented (t, m, n)-multi-secret sharing (or (t, m, n)-GOMSS), in which recovering each secret requires all m $(n \geq m \geq t)$ participants to have valid shares and actually participate in secret reconstruction. As an example, the paper then proposes a simple (t, m, n)-GOMSS scheme. In the scheme, every shareholder has only one share; to recover a secret, m shareholders construct a Polynomial-based randomized component (PRC) each with the share to form a tightly coupled group, which forces the secret to be recovered only with all m valid PRCs. As a result, the scheme can thwart the above illegal participant attack. The scheme is simple as well as flexible and does not depend on conventional hard problems or one way functions.

Key words — Multi-secret sharing, Shamirs scheme, Tightly coupled group, Polynomial-based randomized component (PRC).

I. Introduction

As a fundamental cryptographic tool, the first (t, n) threshold secret sharing (t, n)-SS scheme was introduced independently by Shamir [1] and Blakley[2] in 1979. A (t, n)-SS scheme is used to share one secret among n shareholders such that at least t shareholders are able to reconstruct the secret, but fewer than t shareholders can't. Shareholders are usually called participants when they participate in a secret reconstruction. However, a (t, n)-SS scheme only allows participants to share a single secret.

To improve the efficiency of secret sharing, (t, n)-multi-secret sharing (or (t, n)-MSS) was proposed in the past, in which at least t shareholders are required to recover each secret and less than t shareholders cannot reconstruct any secret.

In 1994, Jakson *et al.*[3] classified multi-secret sharing schemes into two types: one is one-time-use scheme and the other is multi-use scheme. In a one-time-use scheme, the dealer must redistribute new shares to every shareholder after some particular secrets are recovered. In a multi-use scheme, every participant is allowed to use share(s) repeatedly to recover multiple secrets without share redistribution.

Schemes in Refs.[4–6] are all based on two-variable one-way function. Chien *et al.*'s scheme[4] shares all secrets by a system of linear equations, in which unknown parameters of equations serve as secrets to be shared. Yang *et al.*'s scheme[5] uses coefficients of a polynomial as secrets and all secrets are recovered at a time. Pang *et al.*'s scheme[6] employs different values of a polynomial as secrets, and the degree of the polynomial depends on the number of secrets and the number of shareholders.

However, if there are more than t participants in the above (t, n)-MSS schemes and an adversary manages to impersonate a legal participant but without any valid share, the adversary can communicate with the other participants to collect up to t valid shares during secret reconstructions, and eventually recover all secrets later. We call it Illegal participant (IP) attack. The paper focuses on how to construct a MSS scheme capable of preventing IP attack.

In order to thwart the IP attack, some verifiable (t, n) MSS schemes are proposed to prevent adversaries without valid shares from joining secret reconstructions. For example, based on Yang *et al.*'s scheme[5], Shao *et al.*'s scheme[7] employs the intractability of the discrete logarithm and RSA cryptosystem to verify the validity of shares. Similarly, schemes in Refs.[8,9] are based on elliptic curves and bilinear maps. Generally, these schemes are

*Manuscript Received Mar. 30, 2015; Accepted July. 22, 2015. This work is supported in part by the National Natural Science Foundation of China (No.61572454, No.61232018, No.61272472, No.61202404), Open Project of Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University.

DOI:10.1049/cje.2016.08.014

based on some cryptographic assumptions.

In Lin *et al.*'s scheme[10], shareholders compute new shares during each recovering phase and then works as a dealer to distribute sub-share to other participates, the threshold decreases each time a secret is recovered. While in Harn's scheme[11], the dealer selects multiple polynomials and generates the same number of shares for each shareholder; every shareholder constructs a Lagrange component, which is a linear combination of its shares, to recover a secret. In Liu *et al.*'s scheme[12], every shareholder distributes its sub-secret among n shareholders by using a polynomial of degree $t-1$ and secrets are different linear combinations of these sub-secrets.

In multi-use (t,n)-MSS scheme, a share is used more than once and thus shares should be well protected in secret reconstructions. Some schemes are available to protect shares. The most common method is to use a one-way function[13,14]. Each shareholder keeps only one share and uses the output of a one-way function of the share to reconstruct different secrets, the security of these schemes depends on one-way functions. There are also some other methods to protect shares, for example, using a multiparty zero knowledge interactive proof protocol[15], shuffling method[16] and linear combination[11], but they are all very complicated. Zhao *et al.*[17] gives an abstraction of multisecret sharing based on Lagrange interpolating polynomial, which is formalized in the applied pi-calculus by using an equational theory that characterizes the cryptographic semantics of multisecret sharing based on Lagrange interpolating polynomial.

In this paper, we will present a new type of MSS scheme in which shareholders, with one share each, are allowed to share multiple secrets in a simpler and more efficient way; the scheme can prevent IP attack without any verifiable secret sharing and does not depend on any conventional hard problem or one way function.

The rest of this paper is organized as follows. Section II gives the definition of the group oriented (t,m,n) multi-secret sharing (or (t,m,n)-GOMSS); Section III proposes a simple (t,m,n)-GOMSS based a single polynomial, which is followed by the correctness and security analysis in Section IV. Section V compares our scheme with related work and Section VI concludes the paper.

II. Definition of (t,m,n)-GOMSS

In this section, we will present the notion of (t,m,n)-GOMSS.

1. Informal definition of (t,m,n)-GOMSS

A shareholder is called participant when it takes part in a secret reconstruction. A (t,m,n)-GOMSS scheme shares multiple secrets among n shareholders such that

1) any group of at least t shareholders is qualified to reconstruct all secrets,

2) a group of less than t shareholders cannot recover any secret and

3) one secret can be recovered only if all $m (m \geq t)$ participants in the same group have valid shares and actually participate in the secret reconstruction.

2. Formal description of (t,m,n)-GOMSS

Definition 1 ((t,m,n)-GOMSS): Suppose $\mathcal{SH}$ and $\mathcal{SE}$ denote the share space and secret space respectively; $\{s_i | s_i \in \mathcal{SE}\}$ are secrets to be shared among totally n shareholders, $\mathcal{U} = \{U_i | i \in \mathcal{I}\}$, $\mathcal{I} = \{1,2,\ldots,n\}$; each shareholder, *e.g.* U_i, has the public information $x_i (x_i \in \mathcal{SH})$; $\mathcal{U}_m = \{U_i | i \in \mathcal{I}_m \subseteq \mathcal{I}, |\mathcal{I}_m| = m\} \subseteq \mathcal{U}$, denotes m $(m \geq t)$ out of n shareholders. A (t,m,n)-GOMSS comprises 3 algorithms $\{SD, RCC, SR\}$.

1) Share distribution (SD):

The algorithm generates a couple of shares for each shareholder.

$SD : \mathcal{SH} \rightarrow \mathcal{SH}$ is a polynomial time algorithm which takes a shareholder's public information as input and produces a share as output. *i.e.* $h_i = SD(x_i)$, where h_i is $U_i{}'s$ share in $\mathcal{SH}$.

2) Randomized component construction (RCC):

$RCC : \mathcal{SH} \times \mathcal{SH}^{(m)} \times \mathcal{SE} \rightarrow \mathcal{SH}$ is a probabilistic polynomial time algorithm which binds the share of a participant with the public information of all m participants; *i.e.* $c_{vi} = RCC(h_i, \mathcal{X}_m, r_{vi})$, where c_{vi} $(c_{vi} \in \mathcal{SH})$ is the Randomized component (RC) of U_i for the secret s_v, $\mathcal{X}_m = \{x_i | x_i \in \mathcal{SH}, i \in \mathcal{I}_m\}$ is the public information set of $\mathcal{U}_m$, and r_{vi} $(r_{vi} \in \mathcal{SE})$ is a random integer chosen by U_i privately.

3) Secret reconstruction (SR):

The algorithm recovers secrets with RCs.

$SR : \mathcal{SH}^{(m)} \rightarrow \mathcal{SE}$ is a polynomial time algorithm which takes m RCs as input and produces a secret as output; that is, $s_v = SR(\mathcal{RC}^v_m)$, where $\mathcal{RC}^v_m = \{c_{vi} | c_{vi} = RCC(h_i, \mathcal{X}_m, r_{vi}) \in \mathcal{SH}, i \in \mathcal{I}_m\}$ is the RC set of $\mathcal{U}_m$ for the secret $s_v \in \mathcal{SE}$. Besides, if $\mathcal{RC}^v_m$, $(n \geq m \geq t)$, is the right RC set generated for $s_v (v = 1, 2...k)$, and $\mathcal{RC}'$ is a RC set actually used in recovering s_v, a (t,m,n)-GOMSS satisfies

$$P(s_v = SR(\mathcal{RC}') | \mathcal{RC}' \cap \mathcal{RC}^v_m \neq \Phi) = 1 \rightarrow \mathcal{RC}' = \mathcal{RC}^v_m \tag{1}$$

where $P(.)$ is the probability distribution function.

Remark 1 In Eq.(1), $\mathcal{RC}^v_m$ denotes the correct RC set of $\mathcal{U}_m$ for a secret s_v, *i.e.* each RC in $\mathcal{RC}^v_m$ is generated with the corresponding valid shares according to RCC algorithm and $s_v = SR(\mathcal{RC}^v_m)$. $\mathcal{U}_m$ produces the only RC set,$\mathcal{RC}^v_m$, for the secret s_v. Eq.(1) implies that, if a RC set $\mathcal{RC}'$, having at least one RC in common with $\mathcal{RC}^v_m$, could correctly recover s_v, it must be identical with $\mathcal{RC}^v_m$. Under this circumstances, an adversary in (t,m,n)-GOMSS, even having up to $(m-1)$ valid RCs, (note that $(m-1)$ could be much larger than the threshold t), is still unable to recover the secret. It also means that recovering each secret requires all m participants in $\mathcal{U}_m$ to have valid shares.

III. Our Proposed Scheme

In the proposed (t, m, n)-GOMSS scheme, there are 3 types of entities, a dealer, n shareholders and some adversaries. We assume that the dealer is trusted by all shareholders, there is a secure channel between the dealer and each shareholder, a private channel exists between each pair of shareholders.

1. Adversary model

The proposed scheme aims to prevent an adversary, without any valid share, from obtaining secrets rather than to ensure recovering correct secrets. Thus, we assume that all legal participants would rather give up the secret reconstruction than leak the secret to adversaries, because safeguarding secrets is of the first importance in secret sharing schemes.

We assume that outsiders are main adversaries in our scheme.

An outsider is an adversary without any valid share, it exists in either of the 2 cases, 1) Outsiders may crack some participants' private channels and intercept the information to derive secrets. 2) An outsider may impersonate some legal shareholder to join secret reconstructions but without the corresponding valid share. They can communicate with the other participants to collect their legal shares and derive secrets.

Of course, less than t legal shareholders may try to reconstruct secrets by conspiring. However, this case is the same as that in basic (t, n)-SS schemes, therefore, we won't consider this attack for simplicity.

2. Design of (t, m, n)-GOMSS

On one hand, the proposed (t, m, n)-GOMSS scheme remains the basic properties of a (t, n)-SS scheme (*e.g.* Shamir's SS[1]), i) any group of at least t legal shareholders are able to recover each secret while less than t shareholders can't; ii) Each shareholder has a single share. On the other hand, it also has the new property, *i.e.* recovering each secret requires all m participants to have valid shares and substantially take part in the secret reconstructions.

To achieve these goals, each participant has to protect the share from exposure during each secret reconstruction because the share needs to be used repeatedly; at the same time, to force all participants to actually join the secret reconstruction, the scheme requires all participants to form a tightly coupled group. In our scheme, all m participants protect shares and form a tightly coupled group merely by constructing a PRC each.

According to the formal definition, the proposed (t, m, n)-GOMSS scheme contains the following 3 phases:

1) Share generation:

Suppose there are n shareholders, $\mathcal{U}_n = \{U_i | i \in \mathcal{I}_n\}, \mathcal{I}_n = \{1, 2...n\}$, and each shareholder, *e.g.* U_i, has the public information $x_i \in Z_p$, where p and q are large primes with $p > (n+1)q^2 + q$. $\mathcal{S} = \{s_i | s_i \in Z_q, i = 1, 2, ..., k\}$ are $k, (k << q)$, secrets to be shared among $\mathcal{U}_n$. The dealer privately selects a polynomial $f(x) = a_0 + a_1 x + ... + a_{t-1}x^{t-1} \bmod p$, where $a_i, (i = 0, 1, 2, ..., t-1)$, are uniformly selected in Z_p. For each secret $s_i (s_i \in \mathcal{S})$, the dealer first selects an integer d_i, $(d_i \in Z_p,\ d_i \notin \mathcal{X} = \{x_1, x_2...x_n\})$, and a random integer k_i $(k_i \in Z_q)$, then it evaluates e_i $(e_i \in Z_p)$ such that $s_i + k_i q = e_i f(d_i) \bmod p$ and hold. Next, the dealer makes d_i and e_i public. Finally, the dealer computes $f(x_j)$ and sends it as the share to shareholder U_j secretly.

2) Construction of PRC:

Suppose m $(n \geq m \geq t)$ shareholders $\mathcal{U}_m = \{U_i | i \in \mathcal{I}_m\}$, $(\mathcal{I}_m \subseteq \mathcal{I}_n, |\mathrm{I}_m| = m)$, with the public information set $\mathcal{X}_m = \{x_i | x_i \in \mathcal{X}, i \in \mathcal{I}_m\}$, collaborate to reconstruct each secret, s_i $(s_i \in \mathcal{S})$, each participant U_j in $\mathcal{U}_m$ uniformly picks an integer r_{ij} over Z_q privately and computes the PRC for s_i as $c_{ij} = (e_i f(x_j) \prod_{\substack{x_v \in \mathcal{X}_m \\ x_v \neq x_j}} \dfrac{d_i - x_v}{x_j - x_v} + r_{ij} q) \bmod p$.

Remark 2 A PRC serves as 2 functions in the scheme simultaneously, one is to protect a shareholder's share (e.g. $U_j{}'s$ share $f(x_j)$ in the c_{ij}) from exposure; the other is to help all participants form a tightly coupled group because each PRC bind the share with the other participants public information by a random number (*e.g.* r_{ij} in c_{ij}) .

3) Secret reconstruction:

Each participant, *e.g.* U_j in $\mathcal{U}_m$ sends c_{ij} to the other participants. On obtaining all the m PRCs, $\mathcal{RC}_m = \{c_{ij} | c_{ij} = (e_i f(x_j) \prod_{\substack{x_v \in \mathcal{X}_m \\ x_v \neq x_j}} \dfrac{d_i - x_v}{x_j - x_v} + r_{ij} q) \bmod p, j \in \mathcal{I}_m\}$, U_j can evaluates the secret $s_i = \sum\limits_{j \in \mathcal{I}_m} c_{ij} \bmod p \bmod q$.

IV. Correctness and Security Analysis

1. Correctness

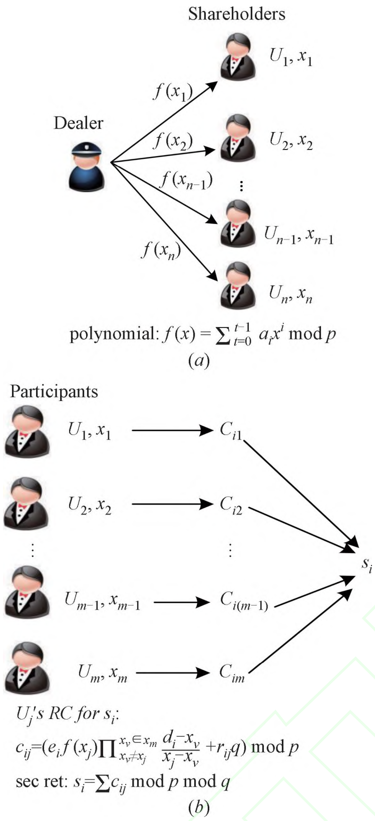
The following fact ensures that a group of m $(m \geq t)$ participants can recover each secret, *e.g.*, s_i.

$$
\begin{aligned}
&\sum_{j \in \mathcal{I}_m} c_{ij} \bmod p \bmod q \\
&= \sum_{j \in \mathcal{I}_m} (e_i f(x_j) \prod_{\substack{x_v \in \mathcal{X}_m \\ x_v \neq x_j}} \frac{d_i - x_v}{x_j - x_v} + r_{ij} q) \bmod p \bmod q \\
&= (e_i \sum_{j \in \mathcal{I}_m} (f(x_j) \prod_{\substack{x_v \in \mathcal{X}_m \\ x_v \neq x_j}} \frac{d_i - x_v}{x_j - x_v}) \bmod p \\
&\qquad + \sum_{j \in \mathcal{I}_m} r_{ij} q) \bmod p \bmod q
\end{aligned}
$$

$$= (s_i + k_i q + \sum_{j\in\mathcal{I}_m} r_{ij} q) \bmod p \bmod q \qquad (2)$$

$$= (s_i + k_i q + \sum_{j\in\mathcal{I}_m} r_{ij} q) \bmod q \qquad (3)$$

$$= s_i$$



(*a*)

(*b*)

Fig. 1. (*a*) Share Generation; (*b*) Construction of PRC and Secret Reconstruction; Phases of the proposed (t, m, n)-GOMSS

Eq.(2) is equivalent to Eq.(3) because of $s_i + k_i q < q + q^2$, $\sum_{j\in\mathcal{I}_m} r_{ij} q < mq^2 \le nq^2$, $q + (n+1)q^2 < p$ and thus $(s_i + k_i q + \sum_{j\in\mathcal{I}_m} r_{ij} q) < p$. However, it is known from Lagrange interpolation that the secret cannot be reconstructed in the case of $m < t$.[1]

2. Security analysis

In this subsection, Theorem 1 proves that a share can be well protected by PRCs; Theorem 2 demonstrates the security of our scheme against outsiders and Theorem 3 assures us that our scheme is still secure even if adversaries obtain recovered secrets.

Theorem 1 In the proposed $(t, m, n)-$GOMSS, given a PRC, an adversary only has the probablity $1/q$ to obtain the share from the PRC.

Proof From $c_{ij} = (e_i f(x_j) \prod_{\substack{x_v\in\mathcal{X}_m \\ x_v\neq x_j}} \frac{d_i - x_v}{x_j - x_v} + r_{ij} q) \bmod p$, we have $f(x_j) = (e_i \prod_{\substack{x_v\in\mathcal{X}_m \\ x_v\neq x_j}} \frac{d_i-x_v}{x_j-x_v})^{-1}(c_{ij} - r_{ij}q) \bmod p$.

If there exist $r_{ij}^{'}$ and r_{ij} $(r_{ij}^{'}, r_{ij} \in Z_q)$ such that $(e_i \prod_{\substack{x_v\in\mathcal{X}_m \\ x_v\neq x_j}} \frac{d_i-x_v}{x_j-x_v})^{-1}(c_{ij} - r_{ij}q) \bmod p = (e_i \prod_{\substack{x_v\in\mathcal{X}_m \\ x_v\neq x_j}} \frac{d_i-x_v}{x_j-x_v})^{-1}(c_{ij} - r_{ij}^{'}q) \bmod p = f(x_j)$ holds, then we have $(c_{ij} - r_{ij}q) \bmod p = (c_{ij} - r_{ij}^{'}q) \bmod p$, *i.e.* $p|(r_{ij} - r_{ij}^{'})$, because of $\gcd((e_i \prod_{\substack{x_v\in\mathcal{X}_m \\ x_v\neq x_j}} \frac{d_i-x_v}{x_j-x_v})^{-1}, p) = 1$ and $\gcd(p, q) = 1$. As a result, $r_{ij} = r_{ij}^{'}$ follows due to $r_{ij}, r_{ij}^{'} \in Z_q$ and $q < p$, which means different r_{ij} produces distinct $f(x_j)$ for the given c_{ij}. Since r_{ij} is uniformly selected in Z_q by participant, there are q distinct possible values of $f(x_j)$ for the given c_{ij}, each with the same possibility. Therefore, the probability for an adversary to guess the share $f(x_j)$ is $1/q$.

The above theorem implies that c_{ij} is able to protect the share $f(x_j)$ even if it is exposed to the outside, because guessing $f(x_j)$ from c_{ij} is as difficult as directly guessing the secret s_i in Z_q .

Lemma 1 Suppose that random variable x is uniformly distributed in Z_p, for any value $t \in Z_p^*$, xt has a uniform distribution over Z_p .

Proof $t \in Z_p^*$ means $t \in Z_p$ and $\gcd(t, p) = 1$. Suppose x_1 and x_2 are 2 distinct values of x in Z_p, $tx_1, tx_2 \in Z_p$ follows; if $tx_1 = tx_2 \bmod p$ holds, then we have $p|(x_1 - x_2)$ due to $\gcd(t, p) = 1$; it is followed by $x_1 = x_2$ because of x_1,$x_2 \in Z_p$, which is contradictory to $x_1 \neq x_2$. Therefore,$tx_1 \neq tx_2 \bmod p$ holds if $x_1 \neq x_2$. That is, xt is a permutation of $\{0, 1, 2, \ldots, p-1\}$ when x varies from 0 to $p-1$, *i.e.* xt is uniformly distributed over Z_p .

Lemma 2 Suppose that p, q $(p > q)$ are prime numbers, if random variables x and y are mutually independent uniformly distributed in Z_p and Z_q respectively, then $(ax+by) \bmod p$ has a uniform distribution in Z_p for fixed values $a, b \in Z_p^*$.

Proof We know from Lemma 1 that ax is uniformly distributed over Z_p. Let y_1 and y_2 $(y_1, y_2 \in Z_q)$ be 2 distinct values of random variable y. Obviously, $by_1 \bmod p$ and $by_2 \bmod p$ are 2 different values in Z_p. Furthermore,$(ax+by_1) \bmod p$ and $(ax+by_2) \bmod p$ are 2 distinct permutations of $\{0, 1, 2, \ldots, p-1\}$ when x changes from 0 to $p-1$. As a result, $(ax + by) \bmod p$ generates q distinct permutations of $\{0, 1, 2, \ldots, p-1\}$ when x changes from 0 to $p-1$ and y varies from 0 to $p-1$. As we all know, each value in Z_p appears once and only once in each permutation of $\{0, 1, 2, \ldots, p-1\}$, therefore, each value of $(ax + by) \bmod p$ appears totally q and only q times in these q distinct permutations of $\{0, 1, 2, \ldots, p-1\}$. Consequently, the probability of each value of $(ax+by) \bmod p$

is $q/pq = 1/p$.

Theorem 2 If m $(n \geq m \geq t)$ participants collaborate to recover each secret, *e.g.* s_j, in the proposed (t, m, n)-GOMSS, outsiders, even with $m-1$ valid PRCs for s_j, cannot get the secret. Specifically, suppose the participants' PRC set for s_j is $\mathcal{RC}_m^j = \{c_{ji} | i \in \mathcal{I}_m\}, (\mathcal{I}_m \subseteq \mathcal{I}_n = \{1, 2, ...n\}, |\mathcal{I}_m| = m)$, if outsiders have already known $\mathcal{RC}_{m-1}^j = \{c_{ji} | i \in \mathcal{I}_{m-1}\}, (\mathcal{I}_{m-1} \subset \mathcal{I}_m, |\mathcal{I}_{m-1}| = m-1)$, then $P(s_j | \mathcal{RC}_{m-1}^j)$, the probability for outsiders to get the secret is no more than $(\lfloor p/ \rfloor + 1)/p$.

Proof From the theorem 1, we learn that it is impractical for Outsiders to obtain the secret by deriving shares from known PRCs. In this case, Outsiders have to get the secret by the following 2 ways.

1) Outsiders evaluate $s_j^{'} = \sum_{i \in \mathcal{I}_{m-1}} c_{ji} \bmod p \bmod q$ to get the secret s_j if $s_j^{'}$ happens to be equal to $s_j = \sum_{i \in \mathcal{I}_m} c_{ji} \bmod p \bmod q$. In this case, $s_j^{'} = s_j$ means $(\sum_{i \in \mathcal{I}_m} c_{ji} \bmod p - \sum_{i \in \mathcal{I}_{m-1}} c_{ji} \bmod p) \bmod q = 0$, that is,

$$c_{jh} = \lambda q \tag{4}$$

where λ is an integer, $h \in \mathcal{I}_m$ and $h \notin \mathcal{I}_{m-1}$.

Note that, in $c_{jh} = (e_j f(x_h) \prod_{\substack{v \in \mathcal{I}_m \\ v \neq h}} \frac{d_j - x_v}{x_h - x_v} + r_{jh} q) \bmod p$, $f(x_h)$ includes $a_i (i = 0, 1, 2, ..., t-1)$, which, for outsiders, are t random variables uniformly distributed over Z_p. Therefore, by repeatedly using lemma 1 and lemma 2, we know that c_{jh} is uniformly distributed in Z_p for outsiders. As a result, there are at most $\lfloor p/q \rfloor + 1$ values of c_{jh} satisfying Eq.(4), *i.e.*, c_{jh} is a multiple of q, and thus the probability of $s_j^{'} = s_j$ is at most $(\lfloor p/q \rfloor + 1)/p$.

2) outsiders guess $c_{jh}^{'}$ such that $s_j = (\sum_{i \in \mathcal{I}_{m-1}} c_{ji} + c_{jh}^{'}) \bmod p \bmod q$ holds, which means $s_j = (\sum_{i \in \mathcal{I}_{m-1}} c_{ji} + c_{jh}) \bmod p \bmod q = \sum_{i \in \mathcal{I}_{m-1}} c_{ji} + c_{jh}^{'}) \bmod p \bmod q$, *i.e.*,

$$(c_{jh} - c_{jh}^{'}) \bmod p = \lambda q \tag{5}$$

where λ is an integer, $h \in \mathcal{I}_m$ and $h \notin \mathcal{I}_{m-1}$. As mentioned above, c_{jh} has a uniform distribution over Z_p, and thus, given the guessed $c_{jh}^{'}$, the left hand side of Eq.(5) is uniformly distributed over Z_p for Outsiders. Consequently, there are at most $\lfloor p/q \rfloor + 1$ out of p values of satisfying Eq.(5). Therefore, the probability of the secret is also at most $(\lfloor p/q \rfloor + 1)/p$ while guessing $c_{jh}^{'}$.

It concludes from 1) and 2) that outsiders cannot obtain the secret with a probability more than $(\lfloor p/q \rfloor + 1)/p$ even if they have $m-1$ PRCs available.

Theorem 3 It is computationally infeasible for adversaries to obtain a secret from the recovered ones.

Proof Without losing generality, suppose adversaries have the recovered secrets $\{s_1, s_2 ... s_j\}, (j < k)$, available, let examine the feasibility of obtaining the new secret

$$\begin{aligned} s_{j+1} &= e_{j+1} f(c_{j+1}) \bmod p \bmod q \\ &= e_{j+1}(a_0 + a_1 c_{j+1} + ... \ + a_{t-1} {c_{j+1}}^{t-1}) \bmod p \bmod q \end{aligned} \tag{6}$$

From the recovered secrets, adversaries have the following system of equations,

$$\begin{cases} s_1 + k_1 q = e_1 f(c_1) \bmod p \\ = e_1(a_0 + a_1 c_1 + ... \ + a_{t-1} {c_1}^{t-1}) \bmod p \\ \quad \vdots \\ s_j + k_j q = e_j f(c_j) \bmod p \\ = e_j(a_0 + a_1 c_j + ... \ + a_{t-1} {c_j}^{t-1}) \bmod p \end{cases} \tag{7}$$

There are $2j$ unknown parameters (*i.e.* $k_1 ... k_j ; f(c_1) ... f(c_j)$) or $j + t$ unknown parameters (*i.e.*$k_1, k_2, ..., k_j ; a_0, a_1, ..., a_{t-1}$) in j available equations. To get s_{j+1} in Eq.(6), adversaries need to 1) guess $f(c_{j+1})$ or 2) solve $\{a_0, a_1, ..., a_{t-1}\}$ in Eq.(7).

1) To guess $f(c_{j+1})$ and then evaluate $e_{j+1} f(c_{j+1}) \bmod p \bmod q$. The probability to get the right s_{j+1} is at most $(\lfloor p/q \rfloor + 1)/p$, which converges to $1/q$ when q converges to infinity. The probability is obviously negligible since q is a large integer.

2) To solve $\{a_0, a_1, \ldots, a_{t-1}\}$ in Eq.(7) to get the polynomial $f(x)$ and compute s_{j+1} by Eq.(6). Specifically, we can obtain from Eq.(7) an indeterminate equation over Z_p with t unknowns, $\{k_1, k_2, \ldots, k_t\}$, *e.g.*

$$a_{t-1} = g(k_1, k_2, \ldots, k_t) \bmod p \tag{8}$$

where $g(k_1, k_2, \ldots, k_t)$ is a linear combination of $k_1, k_2, \ldots, k_t$; given a_{t-1}, we can get $a_{t-2}, a_{t-3}, \ldots, a_0$. However, there exist p possible values of a_{t-1} for totally q^t tuples$\{k_1, k_2, \ldots, k_t\}$. It is computationally infeasible to identify the correct a_{t-1} in $f(x)$ from Eq.(8) since p is a large prime.

V. Properties and Comparisons

Compared with related schemes, our scheme has the following features.

1. Properties

1) Simplicity

On one hand, there is only one polynomial in our scheme and each shareholder has a single share; no matter how many secrets need to be recovered, a shareholder just uses the same share to construct different PRCs. On the other hand, in constructing different PRCs, *e.g.* $c_{ij} = (e_i f(x_j) \prod_{\substack{x_v \in \mathcal{X}_m \\ x_v \neq x_j}} \frac{d_i - x_v}{x_j - x_v} + r_{ij} q) \bmod p$, $(i = 1, 2, \ldots)$, the shareholder U_j in the same tightly coupled group just reuses the same part $f(x_j)\prod_{\substack{x_v\in\mathcal{X}_m\\ x_v\neq x_j}}\dfrac{d_i-x_v}{x_j-x_v}$. That is, U_j just needs to evaluate $f(x_j)\prod_{\substack{x_v\in\mathcal{X}_m\\ x_v\neq x_j}}\dfrac{d_i-x_v}{x_j-x_v}$ only once.

The reconstruction of each secret is almost as simple as Shamir's basic (t,n)-SS.

2) Flexibility and efficiency

The proposed scheme allows the dealer to decide the number of secrets to be shared freely even after share distribution, *i.e.* the number of secrets is independent of other parameters such as the threshold or the number of shareholders. Thus it is flexible.

As a multi-secret scheme, it is of great importance that a group of shareholders could share as many secrets as possible while each shareholder just keeps fixed number of shares. Our scheme allows shareholders, with a single share each, to share more secrets than shares. Thus it is efficient in term of the number of secrets.

3) Tightly coupled

In each secret reconstruction of our scheme, all participants are required to form a tightly coupled group by constructing a PRC each, which is used to prevent the IP attack. In this case, recovering each secret requires each participant in the group to possess a valid share and actually join the secret reconstruction. Otherwise, no correct secret can be recovered.

4) Free of conventional hard problems and one-way functions

Unlike most MSS schemes, the proposed scheme does not depend on any conventional hard problem or one-way function. It is actually based on the fact that an indeterminate equation has multiple solutions over a finite field.

2. Comparisons

We will compare our scheme with related ones[4–6][8–11][18] in the following aspects.

1) Shares/shareholder (S/S)

For a shareholder, S/S denotes the efficiency to share a secret with others. Obviously, the less S/S is, the more efficient a MSS scheme is.

Our scheme requires a shareholder to have just one share, which is optimal and the same as schemes in[4–6][8–10]. S/Ss in Liu's scheme[18] and Harn's scheme[11] are the same as the number of shareholders and polynomials respectively, both of which are much larger than 1.

2) Flexibility of secret number

A flexible MSS scheme should allow users to choose the number of secrets freely after share distribution. Among all above schemes, Ours and schemes in Refs.[4–6,8] allow users to pick the number of secrets freely, but schemes in Refs.[9–11,18] don't.

3) Computational assumption independency

Generally speaking, a scheme without computational assumptions is obviously better.

Ours and schemes in Refs.[10–11,18] are independent of conventional hard problems and one-way functions while schemes[4–6] depend on two-variable one-way functions and schemes in[8–9] depend on hard problems related to elliptic curves and bilinear maps.

4) Tightly coupled group

This property means the capability to prevent IP attack without user authentication or share verification. All these schemes except for ours and Harn's[11] don't possess the property.

5) Multi-use

As mentioned in Section I, multi-use is a significant metric to measure the efficiency of secret sharing. A desirable MSS scheme is supposed to allow users to share multiple secrets with the same share(s). Among all above schemes, Lin's[10], Harn's[11], Liu's[17] and ours are multi-use schemes while schemes in Refs.[4–6,8–9] are not because they recover all secrets at once.

Table 1. Comparisons with other MSS schemes

Schemes	Shares/ Share-holder	Flexibility of secret number	Computational assumption independency	Tightly coupled group	Multi-use
Chien[4]	1	√	×	×	×
Yang[5]	1	√	×	×	×
Pang[6]	1	√	×	×	×
Eslami[8]	1	√	×	×	×
Wang[9]	1	×	×	×	×
Lin[10]	1	×	√	×	√
Liu[18]	n	×	√	×	√
Harn[11]	k	×	√	√	√
Our Scheme	1	√	√	√	√

Note:n:number of shareholders;k:number of polynomials used

It concludes from the comparisons that, among of all above MSS schemes, only our scheme shows positive in all the 5 aspects simultaneously, which are main metrics of a MSS scheme.

VI. Conclusion

In order to prevent the IP attack in most (t,n)-MSS schemes, the paper presents the notion of (t,m,n)-GOMSS and proposes a (t,m,n)-GOMSS scheme based on polynomial. In the scheme, recovering a secret requires all participants to have valid shares and actually take part in the secret reconstruction. Otherwise, no correct secret can be recovered. Compared with related schemes, our scheme has the features of one share for each shareholder, flexible secret number, multi-use, tightly coupled group and independency of hard problems/one-way functions.

References

[1] A. Shamir, "How to share a secret", *Communications of the ACM*, Vol.22, No.11, pp.612–613, 1979.

[2] G. Blakley, "Safeguarding cryptographic keys", *Proc.AFIPS 1979 Natl.Conf*, pp.313–317, 1979.

[3] W.A. Jackson, K.M. Martin and C.M. O.Keefe, "On sharing many secrets", *Asiacrypt94*, pp.42–54, 1994.

[4] H.-Y. Chien, J.-K. Jan and Y-M. Tseng, "A practical (t, n) multi-secret sharing scheme", *IEICE Transactions on Fundamentals*, Vol.E83-A, No.12, pp.2792–2765, 2000.

[5] C.C. Yang, T.Y. Chang and M.S. Hwang, "A (t, n) multi-secret sharing scheme", *Applied Mathematics and Computation*, Vol.151, No.2, pp.483–490, 2004.

[6] L.J. Pang and Y.M. Wang, "A new (t, n) multi-secret sharing scheme based on Shamirs secret sharing", *Applied Mathematics and Computation*, Vol.167, No.2, pp.840–848, 2005.

[7] J. Shao and Z. Cao, "A new efficient (t, n) Verifiable multi-secret sharing (VMSS) based on YCH scheme", *Applied Mathematics and Computation*, Vol.168, No.1, pp.135–140, 2005.

[8] Z. Eslami and S.K. Rad, "A new verifiable multi-secret sharing scheme based on bilinear maps " *Wireless Personal Communications*, Vol.63, No.2, pp.459–467, 2012.

[9] S.J. Wang, Y.R. Tsai and C.C. Shen, "Verifiable threshold scheme in multi-secret sharing distributions upon extensions of ECC", *Wireless Personal Communications*, Vol.56, No.1, pp.173–182, 2011.

[10] C. Lin and L. Harn, "Unconditionally secure multi-secret sharing scheme "IEEE International Conference on Computer Science and Automation Engineering (CSAE)"", Vol.1, pp.169–172, 2012.

[11] L. Harn, "Secure secret reconstruction and multi-secret sharing schemes with unconditional security", *Security and Communication Networks*, Vol.7, No.3, pp.567–573, 2014.

[12] J. He and E. Dawson, "Multistage secret sharing based on one way function", *Electronic letters*, Vol.30, No.19, pp.1591–1592, 1994.

[13] L. Harn, "Efficient sharing (broadcasting) of multiple secrets", *IEE Computers and Digital Techniques*, Vol.142, No.3, pp.237–240, 1995.

[14] C. Tang and Z.-A. Yao, "A new (t, n)-threshold secret sharing scheme", *Proc. of 2008 International Conference on Advanced Computer Theory and Engineering - ICACTE08*, pp.920–924, 2008.

[15] X. Zhang, L. Zhang, Q. Zhang, and C. Tang, "A secret sharing shuffling scheme based on polynomial", *Proc. of 2008 IEEE International Conference on Information and Automation*, pp.1746–1750, 2008.

[16] Ch-Q. Hu, X-F. Liao and X-ZH. Cheng, "Verifiable multi-secret sharing based on LFSR sequences", *Theoretical computer science*, Vol.445, No.11, pp.52–62, 2012.

[17] H. Zhao, J.Z. Sun, F. Wang, *et al.*, "A finite equivalence of multisecret sharing based on Lagrange interpolating polynomial", *Security and Communication Networks*, Vol.6, No.9, pp.1169–1175, 2013.

[18] Y.X. Liu, L. Harn, C.N. Yang, et al "Efficient (n, t, n) secret sharing schemes", *Journal of Systems and Software*, Vol.85, No.6, pp.1325–1332, 2012.

Miao Fuyou received his Master degree in Computer Science and Technology from the China University of Mining Technology (Beijing) in 1999. In 2005, he received his Ph.D. degree in Computer Science from the University of Science and Technology of China. Currently, he is an associate professor with the School of Computer Science and Technology, University of Science and Technology of China. His research interests include cryptography, network security and mobile computing. (E-mail: mfy@ustc.edu.cn, personal homepage:cs.ustc.edu.cn/szdw/fjs/201012/t20101218_84381.html)