Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 37(4), 2010, Pages 50–57
ISSN: 1223-6934

Multi-Use Multi-Secret Sharing Scheme for General Access Structure

Partha Sarathi Roy and Avishek Adhikari

Abstract. The main aim of this paper is to construct a multi-secret sharing scheme for general access structure in a trusted dealer model using suitable hash function and Lagrange's interpolation method. Even though, the proposed scheme is a multi-secret and multi-use one, each participant has to carry only one share. The suitable use of collision resistant one way hash function makes the scheme efficient and multi-use. Moreover, the scheme has a nice property that secrets, participants or qualified sets of participants may be added to or even may be made inactive dynamically by the dealer to get a new access structure without altering the shares of the existing participants in the old access structure. Finally, in the proposed scheme, both the combiner and the share holders can verify the correctness of the information that they are receiving from each other.

2010 Mathematics Subject Classification. 94A62.
Key words and phrases. Collision resistant one-way hash function, Lagrange's interpolation method, verifiable, pseudo share.

1. Introduction

In the open system environment, it is important to restrict access of confidential information in the system or on certain nodes in the system. Access is gained through a key, password or token and governed by a secure key management scheme. If the key or the password is shared among several participants in such a way that it can be reconstructed only by a significantly large and responsible group acting in agreement, then a high degree of security is attained. Also in many information technology applications, as well as in real world, it is desirable that actions or secrets to be protected by more than one key (jointly or separately) or that there be several keys and more than one way to recover the secret to initiate the action, using different combinations of keys.

Shamir [9] and Blakley [3] independently addressed this problem in 1979 when they introduced the concept of a threshold scheme. A (t, n) *threshold scheme* is a method where n pieces of information of the secret key K, called *shares* are distributed to n participants so that the secret key can be reconstructed from the knowledge of any t or more shares and the secret key can not be reconstructed from the knowledge of fewer than t shares.

But in reality, there are many situations in which it is desirable to have a more flexible arrangement for reconstructing the secret key. Given some n participants, one may want to designate certain authorized groups of participants who can use their shares to recover the key. This kind of scheme is called secret sharing scheme for general access structure [2], [7].

Received September 22, 2010. Revision received October 25, 2010.
The First author is supported by University of Calcutta.

Formally, a *secret sharing scheme for general access structure* is a method of sharing a secret K among a finite set of participants $\mathcal{P} = \{P_1, P_2, \ldots, P_n\}$ in such a way that

(1) if the participants in $\mathcal{A} \subseteq \mathcal{P}$ are qualified to know the secret, then by pooling together their partial information, they can reconstruct the secret K,
(2) any set $\mathcal{B} \subset \mathcal{P}$ which is not qualified to know K, cannot reconstruct the secret K.

The key is chosen by a special participant $\mathcal{D}$, called the *dealer*, and it is usually assumed that $\mathcal{D} \notin \mathcal{P}$. The dealer gives partial information, called *share*, to each participant to share the secret key K. In some schemes, there is another special participant, called the *combiner*, to whom the participants give their shares to get the corresponding secret. The collection of subsets of participants that can reconstruct the secret in this way is called *access structure* Γ.

Initially, in all the secret sharing schemes, it was assumed that the dealer, the participants and the combiner are all trusted. Under this assumption, many secret sharing schemes for threshold as well as for general access structures were proposed [9], [1]. These concepts were generalized by secret sharing schemes for general access structure with more than one secret with an aim to use the same share of a particular participant more than once. Those schemes are known as *multi-secret sharing schemes* for general access structure. But, it was pointed out that if the combiner is not trusted, the above schemes may not be safe to use more than once under the following situation. Suppose a particular participant, say P, holding only one share, is a member of two different qualified sets of participants having two different secrets. Now to reveal the first secret, the participant P has to give the share to the combiner. Having the share of P, the combiner may play the role of P, without the knowledge of P, while reconstructing the 2nd secret corresponding to the 2nd qualified set of participants. To overcome this problem, in 1994, He-Dawson [6] claimed to propose a multi-stage threshold secret sharing scheme to share multiple secrets based on collision resistant one-way hash function. But in 2007, Geng *et al.* [4] pointed out that He-Dawson scheme was actually the one-time-use scheme. In [4], they proposed a new threshold multi-use multi-secret sharing scheme. In 2006, Pang *et al.* [8] proposed a multi-secret sharing scheme, based on two variable one-way function and Lagrange's interpolation method, for general access structure in which all the secrets are revealed at a time. In 2008, Wei *et al.* [11] proposed a multi-stage secret sharing scheme, based on Lagrange's interpolation method and intractability of discrete logarithm problem, for general access structure in which secrets reveal in a predetermined order. But, in most of the cases, it is required to share different secrets with different access structures.

For more practical purpose, a secret sharing scheme should have the property that the access structure may be modified dynamically i.e., the secrets, participants or qualified sets of participants may be added to or may be made inactive in the existing access structure to get a new access structure without altering the shares of the existing participants in the old access structure. This type of scheme is known as *renewable* secret sharing scheme.

So, in the current scenario, a multi-use, multi-secret, renewable, verifiable secret sharing scheme for general access structure is essential. In the current paper, we deal with all the above aspects of the secret sharing scheme.

The rest of this paper is organized as follows. In Section 2, a new scheme having all the properties mentioned above is introduced. In Section 3, renew process of the proposed scheme is studied. The analysis of the proposed scheme is given in Section 4. Finally, Section 5 deals with the conclusion of the paper.

2. A New Multi-Secret Sharing Scheme for General Access Structure

In this section, a new renewable, multi-use, multi-secret sharing scheme for general access structure in a trusted dealer model is introduced using suitable hash function [10] and Lagrange's interpolation method.

Aim of the scheme

Let the dealer want to share k secret integers s_1, s_2, ..., s_k among n participants P_1, P_2, ..., P_n in such a way that corresponding to each secret s_i, there exists an access structure $\Gamma_{s_i} = \{\mathcal{A}_1^{s_i}, \mathcal{A}_2^{s_i}, \ldots, \mathcal{A}_{l_i}^{s_i}\}$, where $\mathcal{A}_q^{s_i} = \{P_1^{i_q}, P_2^{i_q}, \ldots, P_{m_{i_q}}^{i_q}\} \subseteq \{P_1, P_2, \ldots, P_n\}$, $|A_q^{s_i}| \geq 2$, $q = 1, 2, \ldots, l_i$, $i = 1, 2, \ldots, k$ and $1 \leq l_i \leq 2^n - (n+1)$. Note that we assume all the s_i's to be non negative integers as otherwise the dealer may add some suitable positive integer to all the s_i's to transfer them into non negative integers and may publish the value of the fixed positive integer in the public domain. Also, note that for some $c, d \in \{1, 2, \ldots, k\}$ with $c \neq d$ it may happen that $\Gamma_{s_c} \cap \Gamma_{s_d} \neq \phi$. To obtain such scheme, we explain the following four different phases namely, the Dealer Phase, the Participant Phase (I), the Combiner Phase and the Participant Phase (II).

2.1. Dealer phase. (I) Initialization stage

(1) The dealer chooses the following.
- p, a prime such that $s_i < p$ and $n < p$, $i = 1, 2, \ldots, k$;
- h, a secure collision resistant one-way hash function which takes as input a binary string of any length and provides as output a binary string of fixed length $[\log_2 p] + 1$.
- ID_j, the distinct identifier corresponding to each of the participant P_j, $j = 1, 2, \ldots, n$, where $ID_j \in_R \mathbb{Z}_p^* = \mathbb{Z}_p \setminus \{0\}$, where " $\in_R$ " denotes the random selection.

(2) All the above listed entities and the access structure are made public by the dealer.

(II) Pseudo share generation stage

To generate the pseudo shares and to distribute the shares to each participant the dealer performs the following:

(1) The dealer chooses distinct $x_j \in_R \mathbb{Z}_p$ and sends it secretly to each of the participants P_j, where $j = 1, 2, \ldots, n$. This step may also be performed in another way. Each participant can choose his or her share by himself or herself and can send it to the dealer through a secure channel. The dealer keeps on asking the shares from the participants, till all the shares are distinct.

(2) For the qth qualified set of Γ_{s_i}, the dealer chooses $d_1^{i_q}, d_2^{i_q}, \ldots, d_{m_{i_q}-1}^{i_q} \in_R \mathbb{Z}_p$ to construct the polynomial $f_q^{s_i}(x) = s_i + d_1^{i_q} x + d_2^{i_q} x^2 + \ldots + d_{m_{i_q}-1}^{i_q} x^{m_{i_q}-1}$, $i = 1, 2, \ldots, k$, $q = 1, 2, \ldots, l_i$.

(3) Let $l = max\{l_1, l_2, \ldots, l_k\}$, $u = [\log_2 k] + 1$, $v = [\log_2 l] + 1$. For the participant $P_b^{i_q} \in \mathcal{A}_q^{s_i}$ in Γ_{s_i}, the dealer computes the *pseudo share* $\mathcal{U}_{P_b^{i_q}} = h(x_{P_b^{i_q}} || i_u || q_v)$, where $i = 1, 2, \ldots, k$, $q = 1, 2, \ldots, l_i$, $b = 1, 2, \ldots, m_{i_q}$, i_u is the u-bit binary representation of i and q_v is the v-bit binary representation of q. Here, "||" denotes the concatenation of two binary strings. Note that, though $x_{P_b^{i_q}}$ is an element of $\mathbb{Z}_p$, to avoid the notational complexity, we use the same notation to represent the binary representation of $x_{P_b^{i_q}}$. Applying the similar argument, we consider $\mathcal{U}_{P_b^{i_q}}$ as an element of $\mathbb{Z}_p$.

(4) The dealer computes $\mathcal{B}_{P_b^{i_q}} = f_q^{s_i}(ID_b^{i_q})$, where $i = 1, 2, \ldots, k$, $q = 1, 2, \ldots, l_i$, $b = 1, 2, \ldots, m_{i_q}$.
(5) Finally, the dealer computes and publishes $\mathcal{M}_{P_b^{i_q}} = (\mathcal{B}_{P_b^{i_q}} - \mathcal{U}_{P_b^{i_q}})$, where $i = 1, 2, \ldots, k$, $q = 1, 2, \ldots, l_i$, $b = 1, 2, \ldots, m_{i_q}$.

(III) Prerequisites for verification stage

In this stage, the dealer prepares all the prerequisites for the verification of the participants by the combiner and the verification of the combiner by the participants.

(1) For the verification of the participants by the combiner, the dealer computes and publishes $\mathcal{N}_{P_b^{i_q}} = h(\mathcal{U}_{P_b^{i_q}})$, where $i = 1, 2, \ldots, k$, $q = 1, 2, \ldots, l_i$, $b = 1, 2, \ldots, m_{i_q}$.
(2) For the verification of the combiner by the participants, the dealer computes and publishes $S_i = h(s_i)$, where $i = 1, 2, \ldots, k$.

2.2. Participant Phase (I). Let all the members of $\mathcal{A}_q^{s_i} = \{P_1^{i_q}, P_2^{i_q}, \ldots, P_{m_{i_q}}^{i_q}\}$ in Γ_{s_i} accumulate to reveal s_i. Each participant $P_b^{i_q}$ of $\mathcal{A}_q^{s_i}$ in Γ_{s_i}, $b = 1, 2, \ldots, m_{i_q}$, executes the following steps:

(1) Each participant $P_b^{i_q}$ of $\mathcal{A}_q^{s_i}$ in Γ_{s_i} computes his or her pseudo share $\mathcal{U}_{P_b^{i_q}} = h(x_{P_b^{i_q}} || i_u || q_v)$ with the help of his or her share $x_{P_b^{i_q}}$ and the publicly available entities h, q_v, i_u.
(2) Each member $P_b^{i_q}$ sends his or her pseudo share to the combiner.

2.3. Combiner Phase. In this phase the combiner verifies the participants and computes their secret as follows:

(I) Participants Verification Stage

(1) The combiner receives each of the pseudo share $\mathcal{U}_{P_b^{i_q}}$ from each of the participants of $\mathcal{A}_q^{s_i} \in \Gamma_{s_i}$.
(2) The combiner verifies each participant $P_b^{i_q}$ by computing $\mathcal{N}_{P_b^{i_q}} = h(\mathcal{U}_{P_b^{i_q}})$, $b = 1, 2, \ldots, m_{i_q}$. At this stage, if there be any dishonest participant, he or she will be identified by the combiner.

(II) Secret Reconstruction Stage

(1) Using the fact that $\mathcal{U}_{P_b^{i_q}} + \mathcal{M}_{P_b^{i_q}} = \mathcal{B}_{P_b^{i_q}} = f_q^{s_i}(ID_b^{i_q})$, the combiner computes the secret s_i as follows [5]:

$$s_i = \sum_{b \in \{1,2,\ldots,m_{i_q}\}} (\mathcal{U}_{P_b^{i_q}} + \mathcal{M}_{P_b^{i_q}} \prod_{r \in \{1,2,\ldots,m_{i_q}\} r \neq b} \frac{-ID_{P_r^{i_q}}}{ID_{P_b^{i_q}} - ID_{P_r^{i_q}}} \mod p.$$

(2) The combiner sends the secret s_i securely to each of the participants of $\mathcal{A}_q^{s_i} \in \Gamma_{s_i}$.

2.4. Participant Phase (II). After getting s_i from the combiner, each participant $P_b^{i_q}$ verifies whether the revealed s_i is correct or not with the help of publicly available information h and S_i, $b = 1, 2, \ldots, m_{i_q}$.

Remark 2.1. *The verification steps may also be done using the concept of intractability of the discrete logarithm problem over the filed $\mathbb{Z}_p$, instead of using secure collision resistant one way hash function. In that case, for the verification of participants, the dealer may publish $g^{\mathcal{U}_{P_b^{i_q}}}$ instead of $h(\mathcal{U}_{P_b^{i_q}})$, where g is a primitive element of the field $\mathbb{Z}_p$. Also for the verification of the combiner, the dealer may publish g^{s_i} instead of $h(s_i)$, $i = 1, 2, \ldots, k$, $q = 1, 2, \ldots, l_i$, $b = 1, 2, \ldots, m_{i_q}$.*

3. Renew Process

For more practical use, it is desirable that in an existing secret sharing scheme, secret or secrets, participant or participants and qualified set or sets may be added or may be made inactive by the dealer dynamically without updating the shares of participants of old access structure. In the proposed scheme, that nice property can be incorporated as follows:

(1) **Modification of secret**
Suppose to the existing scheme, a new secret, say s_t, is to be added by the dealer. Let the set of qualified sets of participants corresponding to the secret s_t be $\Gamma_{s_t} = \{\mathcal{A}_1^{s_t}, \mathcal{A}_2^{s_t}, \ldots, \mathcal{A}_{l_t}^{s_t}\}$, where $\mathcal{A}_q^{s_t} = \{P_1^{t_q}, P_2^{t_q}, \ldots, P_{m_{t_q}}^{t_q}\} \subseteq \{P_1, P_2, \ldots, P_n\}$, $q = 1, 2, \ldots, l_t$. The dealer may achieve this as follows:
- The dealer constructs a new polynomial $f_q^{s_t}$ as described earlier, $q = 1, 2, \ldots, l_t$.
- The dealer publishes the values $\mathcal{M}_{P_b^{t_q}}$, $\mathcal{N}_{P_b{}^{t_q}}$, and $\mathcal{S}_t$, where $q = 1, 2, \ldots, l_t, b = 1, 2, \ldots, m_{t_q}$.

To make a secret, say s_t, inactive corresponding to the access structure Γ_{s_t}, the dealer may replace that secret by s_t' and may update all the public values related to the secret.

(2) **Modification of participant**
Suppose, a new participant, say P_{n+1}, has to be added by the dealer to the set of participants $\{P_1, P_2, \ldots, P_n\}$. For that, the dealer does the following:
- The dealer chooses $x_{n+1} \in_R \mathbb{Z}_p$ and $ID_{n+1} \in_R \mathbb{Z}_p^*$, distinct from the previously used x_i's and ID_i's for the existing participants.
- The dealer sends the value of x_{n+1} securely to the participant P_{n+1}.
- The dealer publishes the value ID_{n+1} to the public domain.
- The dealer has to look at the newly formed qualified sets of participants where the new participant P_{n+1} is being added.
- Modify the pseudo shares as well as the public values of the participants related to the modified qualified sets.

If a participant has to be made inactivate by the dealer from the set of participants $\{P_1, P_2, \ldots, P_n\}$, the dealer first has to look at all the access structures, where that participant was present. Then the dealer may change all the secrets of those access structures and update all the public values accordingly.

(3) **Modification of qualified set**
Suppose a new qualified set of participants, say $\mathcal{A}_{l_i+1}^{s_i}$, for the secret s_i is to be added by the dealer to the existing qualified set of participants. To incorporate that, the dealer constructs $f_{l_i+1}^{s_i}$ and publishes the corresponding values $\mathcal{M}_{P_b^{i_q}}$, $\mathcal{N}_{P_b^{i_q}}$, where $q = l_i + 1,\ b = 1, 2, \ldots, m_{i_q}$. To make some qualified set of participants inactivate, the dealer has to incorporate the similar work as in the case of modification of participants.

4. Analysis Of The Proposed Scheme

4.1. Security. Let us explain the security of the pseudo shares, the shares and the secrets as follows. Here hash function plays an important role.

(I) Security of pseudo Share

An adversary can try to derive participant's pseudo share by using publicly available information $\mathcal{M}_{P_b^{i_q}}$. As pseudo share $\mathcal{U}_{P_b^{i_q}}$ is protected by $\mathcal{B}_{P_b^{i_q}}$, it is not possible

for others to derive participant's pseudo share without the knowledge of $\mathcal{B}_{P_b^{i_q}}$ which is obtained from the secret polynomial chosen by the dealer. And it is also impossible to adversary to compute $\mathcal{U}_{P_b^{i_q}}$ from the previously used pseudo shares (it may happen that set of previously used pseudo shares is empty) of $P_b^{i_q}$ under the protection of collision resistant one way hash function. Moreover, it is computationally infeasible to get back $\mathcal{U}_{P_b^{i_q}}$ from publicly available entity $\mathcal{N}_{P_b^{i_q}}$.

(II) Security of Share

Share is chosen by the dealer randomly and delivered to each user P_j secretly. Even though the pseudo share $\mathcal{U}_{P_b^{i_q}}$ is compromised, any malicious adversary can not successfully derive x_j from the equation $\mathcal{M}_{P_b^{i_q}} = \mathcal{B}_{P_b^{i_q}} - \mathcal{U}_{P_b^{i_q}}$ under the protection of the collision resistant one-way hash function.

(III) Security of Secrets

Suppose, all but one participant in $\mathcal{A}_q^{s_i}$ come to get s_i. They have to guess the corresponding value $\mathcal{U}_{P_b^{i_q}}$ of the missing participant. As $\mathcal{U}_{P_b^{i_q}}$ is a $[\log_2 p] + 1$ bit long binary string, the forbidden set of participants will have no extra privilege over an outsider who knows only the value of p. Moreover, it is computationally infeasible to get back s_i from publicly available entity S_i.

From the above discussions, it is clear that the proposed scheme is secure based on the assumption of the hardness of secure collision resistant one-way hash function.

Remark 4.1. (1) *One of the key parameters concerning the efficiency of a scheme is the size of the share space. In the proposed scheme, the size of the share space is same as that of the secret space.*

(2) *In the proposed scheme all the participants and the dealer are assume to be honest. But, it may happen that some of the participants are corrupted. In that scenario verification is required. Using intractability of discrete logarithm problem or using one-way hash function, we can incorporate verifiability in the proposed scheme.*

4.2. Multi-use. The proposed scheme is a multi-use one in the sense that the same share of a particular participant may be used to reveal different secrets corresponding to the different qualified sets of participants. This follows form the fact that the pseudo share, submitted to the combiner to reveal a particular secret, of a participant changes for different secrets and even for different qualified subsets of the same secret. This prevents the combiner from misusing the share of a participant to construct some other secrets. To incorporate this, successive use of one way hash function was used in [4], [6] and [5]. But in the proposed scheme, successive use of hash function is replaced by the concatenation of suitably chosen binary strings to achieve the same property. This replacement makes the proposed scheme more efficient than above mentioned schemes with respect to the computational cost.

4.3. Performance Evaluation. In the proposed scheme, hash function plays the most important role. So, for a particular secret s_i and one of its corresponding qualified sets $\mathcal{A}_q^{s_i}$, we count the number of times that the hash function is used by the dealer, the combiner and the members of $\mathcal{A}_q^{s_i}$. In the initialization stage by the dealer, there is no use of hash function. At the time of computing pseudo share, the dealer has to operate hash function m_{i_q} times. In the third stage of the dealer Phase, the dealer has to operate hash function $m_{i_q} + 1$ times. So, in the Dealer Phase, hash function is operated $(2m_{i_q} + 1)$ times. Each participant operates hash function only once in the Participant Phase (I). When the combiner is going to verify participants,

the combiner has to operate hash function m_{i_q} times. Finally, when participants verify the combiner, then each of them has to operate hash function only once. Note that throughout the protocol, each participant has to operate only hash function just two times and no other operation has required for them. Whereas, in [4], [6] and [5] hash function is used successively to reveal more than one secrets when each participant carry only one share. But avoiding the successive use of hash function, we achieve the same goal.

In the Table 1, we highlight the main features of the schemes [4], [5], [6], [8], [11] along with the proposed scheme in a compact form. Moreover, all the above mentioned schemes are computationally secure.

TABLE 1. Comparison among [6], [4], [5], [8], [11] and the proposed scheme with respect to the various parameters. In the following table GAS and PO denote respectively general access structure and predetermined order.

Scheme	*Multi Use*	*Threshold or GAS*	*Secret revealing order*
He et al.	No	Threshold	PO
Geng et al.	Yes	Threshold	Any order
Han et al.	Yes	Threshold	Any order
Pang et al.	Yes	GAS	All secrets at a time
Wei et al.	No	GAS	PO
Proposed Scheme	Yes	GAS	Any order

5. Conclusion

In current scenario, it is important for a secret sharing scheme to be a multi-use, multi-secret, renewable and verifiable for general access structure. The proposed scheme has all the above mentioned properties. Up to the best of our knowledge, this is the first algorithm for general access structure for Lagranges' interpolation based multi-use multi-secret sharing scheme. Moreover, analysis shows that the proposed scheme is an efficient one and it can provide great capabilities for many applications.

References

[1] A. Adhikari, DNA Secret Sharing, *IEEE World Congress on Evolutionary Computation*, CEC 2006, July 16-21 (2006), 1407–1411.

[2] J.C. Benaloh and J. Leichter, Generalized Secret Sharing and Monotone Functions, *Advances in Cryptology- CRYPTO'88, Lecture Notes in Computer Science* **403** (1990), 190–199.

[3] G.R. Blakley, Safeguarding cryptographic keys, In: *Proceedings of AFIPS'79* **48** (1979), 313–317.

[4] Y.J. Geng, X.H. Fan and F. Hong, A new multi-secret sharing scheme with multi-policy, *The 9th International Conference on Advanced Communication Technology* **3** (2007), 1515–1517.

[5] Y.L. Han and S.Y. Yi, Dynamic Multi-Secret Sharing Scheme, *Int. J. Contemp. Math. Sciences* **3** (2008), no. 1, 37–42.

[6] J. He and E. Dawson, Multi-stage secret sharing based on one-way function, *Electronic Letters* **30** (1994), no. 19, 1591–1592.
[7] M. Ito, A. Saito and T. Nishizeki, Secret Sharing Scheme Realizing General Access Structure, *Proceedings of IEEE Global Telecommunications Conference, Globecom 87*, Tokyo, Japan (1987), 99–102.
[8] L.J. Pang, H. Li and Y. Wang, An efficient and secure multi-secret sharing scheme scheme with general access structure, *Wuhan University Journal of Natural Sciences* **11** (2006), no. 6.
[9] A. Shamir, How to share a secret, *Comm. ACM* **22** (1979), no. 11, 612–613.
[10] D.R. Stinson, *CRYPTOGRAPHY: Theory and Practice*. 2nd ed. Chapman and Hall (2002).
[11] Y. Wei, P. Zhong and G. Xiong, *A Multi-stage Secret Sharing Scheme with General Access Structures*, Wireless Communications, Networking and Mobile Computing, 2008.

(Partha Sarathi Roy and Avishek Adhikari) Department of Pure Mathematics, University of Calcutta, 35, Ballygunge Circular Road, Kolkata 700019, India
E-mail address: royparthasarathi0@gmail.com, aamath@caluniv.ac.in