Infinite Group Theory - From the Past to the Future, World Scientific Publishing, Feb 2018, pp. 19-41. ISBN: 978-981-3204-04-1.

Chapter 2

A Survey: Shamir Threshold Scheme and Its Enhancements

Chi Sing Chum[1], Benjamin Fine[2], and Xiaowen Zhang[1,3]

[1] *Computer Science Dept., Graduate Center, CUNY*
365 Fifth Ave., New York, NY 10016, U.S.A.
E-mail: cchum@gradcenter.cuny.edu

[2] *Mathematics Dept., Fairfield University*
1073 North Benson Road, Fairfield, CT 06824, U.S.A.
E-mail: fine@fairfield.edu

[3] *Computer Science Dept., College of Staten Island, CUNY*
2800 Victory Blvd, Staten Island, NY 10314, U.S.A.
E-mail: xiaowen.zhang@csi.cuny.edu

ABSTRACT. This paper serves as an introduction to secret sharing scheme, and it provides the fundamental understandings to the scheme from various aspects. We first review the basics of a Shamir threshold scheme, and discuss various enhancements so that the scheme can be proactive and verifiable. We then show how a Shamir scheme can be extended to realize any general access structure. We also point out the relationship between a Shamir scheme and other topics such as error correction code, ramp scheme, information disposal algorithm and multiparty computation. Finally, we briefly discuss other platforms for its implementation.

1. Introduction

A **secret sharing scheme** is a method to distribute a secret among a group of participants by giving a share of the secret to each. The secret can be recovered only if a sufficient number of participants combines their shares.

Formally we have the following. We have a secret K and a group of n participants. This group is called the **access control group**. A **dealer**

Keywords: Secret sharing, threshold scheme, access structure, Reed-Solomon code, ramp scheme, information disposal algorithm, multiparty computation.

allocates shares to each participant under given conditions. If a sufficient number of participants combine their shares, then the secret can be recovered. If $t \leq n$ then an (t, n)-**threshold scheme** is one with n total participants and in which any t participants can combine their shares and recover the secret but not fewer than t. The number t is called the **threshold**. It is a **secure secret sharing scheme** if given less than the threshold there is no chance to recover the secret. If a measure is placed on the set of secrets, and on the set of shares, security can be made precise by saying that when given less than the threshold, all secrets are equally likely, but when given the threshold, there is a unique secret. Secret sharing is an old idea but was formalized mathematically in independent papers in 1979 by Adi Shamir [26] and George Blakley [2].

Shamir [26] proposed a beautiful (t, n) threshold scheme, based on polynomial interpolation, that has many desirable properties. We describe this in Section 3. It is now a standard method for solving the (t, n) secret sharing problem, although there are modifications for different situations that we will discuss in this paper. Blakley [2] in his original paper proposed a geometric solution based on hyperplanes that is less space efficient, for computer storage, than Shamir's. In Blakley's scheme the distributed shares are larger than the secret, whereas in Shamir's scheme they are the same size.

The protection of a private key in an encryption protocol provides strong motivation for the ideas of secret sharing. Based on Kerchhoffs' principle [18], only the private key in an encryption scheme is the secret and not the encryption method itself. When we examine the problem of maintaining sensitive information, we will consider two issues: availability and secrecy. If only one person keeps the entire secret, then there is a risk that the person might lose the secret or the person might not be available when the secret is needed. Hence, it is often wise to allow several people to have access to the secret. On the other hand, the higher the number of people who can access the secret, the higher the chance the secret will be leaked. A secret sharing scheme is designed to solve these issues by splitting a secret into multiple shares and distributing these shares among a group of participants. The secret can only be recovered when the participants of an authorized subset join together to combine their shares.

A secret sharing scheme is a cryptographic primitive with many applications, such as in security protocols, multiparty computation (MPC), Pretty Good Privacy (PGP) key recovering, visual cryptography, threshold cryptography, threshold signature, etc.

The remainder of this paper is organized as follows. In Section 2, we give a brief review on entropy which is related to secret sharing schemes. In Section 3, we discuss the principles of share distribution and secret recovery of a Shamir threshold scheme and its properties. We further talk about different enhancements which make the original threshold scheme proactive or verifiable. In Section 4 we further show how to extend a Shamir threshold scheme to realize any general access structure. In Sections 5 to 8, we discuss the relationship between a Shamir threshold scheme and Reed-Solomon code, ramp scheme, information disposal algorithm, and multiparty computation, respectively. In Section 9, we gave an alternative to Shamir threshold scheme. In Section 10, we discuss another platform for its implementation. We conclude the paper in Section 11.

2. Entropy

In information theory, developed by Shannon [27, 28], entropy is a measure of information or uncertainty. Also see [4, 14, 30] for the details. Let X be a random variable with possible outcomes $\mathcal{X}$ and probability distribution $p(x)$, where $p(x) \geq 0$, $\sum_{x\in\mathcal{X}} p(x) = 1$. Then, the entropy of X is defined as

$$Ent(X) = -\sum_{x\in\mathcal{X}} p(x)\log_2 p(x). \tag{1}$$

In probabilistic terms this is the expected value of $-\log_2 p(x)$. We assume $p(x)\log_2 p(x) = 0,$ if $p(x) = 0$. This is justified because

$$\lim_{p(x)\to 0} p(x)\log_2 p(x) = 0. \tag{2}$$

Example: Let X be a random variable of the event of an unbiased fair coin flipping with the possible outcomes of $\mathcal{X} = \{\text{Head, Tail}\}$, with $p(X = \text{Head}) = p(X = \text{Tail}) = 1/2$, then:

$$\begin{aligned} Ent(X) &= -p(X = \text{Head})\log_2 p(X = \text{Head}) \\ &\quad - p(X = \text{Tail})\log_2 p(X = \text{Tail}) = \frac{1}{2} + \frac{1}{2} = 1. \end{aligned} \tag{3}$$

If the coin is biased with $p(X = \text{Head}) = 1$ and $p(X = \text{Tail}) = 0$, then $Ent(X) = 0$. In this case there is no uncertainty. We can use $Ent(X) = 0$ to infer that $\exists x_i \in \mathcal{X}$ such that $p(x_i) = 1$ and $p(x_j) = 0$ for $j \neq i$. $\square$

Let X and Y be two random variables. The joint entropy $H(X,Y)$ is defined as:

$$H(X,Y) = -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\log_2 p(x,y). \tag{4}$$

Again, as in the case of a single random variable this is the expected value of $-\log_2(p(x,y))$

The conditional entropy $H(X|Y)$ is defined as:

$$\begin{aligned} H(X|Y) &= \sum_{y\in\mathcal{Y}} p(y)H(X|Y=y) \\ &= -\sum_{y\in\mathcal{Y}} p(y)\left(\sum_{x\in\mathcal{X}} p(x|y)\log_2 p(x|y)\right) \\ &= -\sum_{y\in\mathcal{Y}}\sum_{x\in\mathcal{X}} p(y)p(x|y)\log_2 p(x|y). \end{aligned} \tag{5}$$

However, if X and Y are independent, then

$$\begin{aligned} H(X|Y) &= -\sum_{y\in\mathcal{Y}} p(y)\left(\sum_{x\in\mathcal{X}} p(x|y)\log_2 p(x|y)\right) \\ &= \sum_{y\in\mathcal{Y}} p(y)\left(-\sum_{x\in\mathcal{X}} p(x)\log_2 p(x)\right) \\ &= 1\cdot H(X) = H(X). \end{aligned} \tag{6}$$

3. Shamir (t, n) Threshold Scheme

Given a secret K in general a (t,n) secret sharing threshold scheme is a cryptographic primitive in which a secret is split into pieces (shares) and distributed among n participants $p_1, p_2, \ldots, p_n$ so that any group of t or more participants, with $(t \leq n)$, can recover the secret. Meanwhile, any group of $t-1$ or fewer participants cannot recover the secret. By sharing a secret in this way, the availability and reliability issues can be solved. Distributing share and recovering secret [3, 14, 30] will be discussed as follows.

The general idea of a Shamir (t,n) threshold scheme is the following. Let F be any field and $(x_1,y_1),\ldots,(x_n,y_n)$ be n points in F^2 with distinct x_i. We say that a polynomial $P(x)$ of degree less than or equal to $n-1$ over F **interpolates** these points if $P(x_i) = y_i$ for $i = 1,\ldots,n$. The relevant theoretical result that we need is the following. We can see Atkinson [1] for a reference and for a proof.

Theorem 3.1. *Let F be any field and $x_1, \ldots, x_n$ be n distinct elements of F and $y_1, \ldots, y_n$ any elements of F. Then there exists a* **unique** *polynomial of degree $\leq n-1$ that interpolates the n points $(x_i, y_i), i = 1, \ldots, n$.*

Using this theorem, a Shamir (t, n) threshold scheme is roughly this. We choose a field F. The secret is $K \in F$ and we choose a polynomial $P(x)$ of degree at most $t-1$ with K as its constant term. We choose distinct $x_1, \ldots, x_n$ with no $x_i = 0$ and distribute to each of the n participants a point $(x_i, P(x_i))$,$i = 1, \ldots, n$. By the theorem above any t people can determine the interpolating polynomial $P(x)$ and hence recover the secret K. Given an infinite field and fewer that t people there are infinitely polynomials of degree t that can interpolate the given points and hence finding the correct polynomial has probability zero.

We now present a more explicit version of the Shamir scheme using the finite field $\mathbb{Z}_q$ where q is a large prime. By using a finite field Shamir was able to place a finite measure on the set of plaintexts and ciphertexts and showed that with this scheme if there are fewer than t people all secrets are equally likely.

Distributing share: Let K be the secret. The dealer generates a polynomial $P(x)$ of degree at most $t-1$ over $\mathbb{Z}_q$, where q is a prime number $> n$ as follows:

$$P(x) = a_0 + a_1 x + \ldots + a_{t-1} x^{t-1} \pmod{q} \tag{7}$$

where $a_0 = K$ is the secret, $a_1, \ldots, a_{t-1} \in \mathbb{Z}_q$ and are generated randomly.

The dealer arbitrarily chooses different $x_i \in \mathbb{Z}_q - \{0\}$, $i = 1, 2, \ldots, n$. Usually, $x_i = i$ will be chosen for simplicity. The values $x_1, x_2, \ldots, x_n$ are stored in a public area. The dealer calculates $y_i = P(x_i) \pmod{q}, i = 1, 2, \ldots, n$, and distributes to the n participants via a secure channel so that each participant p_i gets one share y_i. For the rest of the paper, we will not repeat the criteria of the generation of the coefficient a_i of the polynomial $P(x)$ and the calculation of the shares $P(x_i)$.

Recovering secret (i): When any t participants join together, we have the following system of t equations. For simplicity, we assume $p_1, p_2, \ldots, p_t$

join together.

$$\begin{aligned} y_1 &= P(x_1) = a_0 + a_1 x_1 + \ldots + a_{t-1} x_1^{t-1} \pmod{q}, \\ y_2 &= P(x_2) = a_0 + a_1 x_2 + \ldots + a_{t-1} x_2^{t-1} \pmod{q}, \\ &\ldots, \\ y_t &= P(x_t) = a_0 + a_1 x_t + \ldots + a_{t-1} x_t^{t-1} \pmod{q}. \end{aligned} \tag{8}$$

In matrix representation, it will be:

$$\begin{bmatrix} 1 & x_1 & \cdots & x_1^{t-1} \\ 1 & x_2 & \cdots & x_2^{t-1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & x_t & \cdots & x_t^{t-1} \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{t-1} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_t \end{bmatrix} \pmod{q}. \tag{9}$$

Let M be the above $t \times t$ Vandermonde matrix. Its determinant is

$$\det(M) = \prod_{1 \le j < k \le t}^{t} (x_k - x_j) \pmod{q}. \tag{10}$$

Since we choose different points for the participants, i.e., different x_i's, $\det(M) \neq 0$, and this guarantees a unique solution. We can solve the system of equations by Gaussian elimination or Crammer's rule. Hence the secret can be recovered.

Recovering secret (ii): Another method is to use Lagrange interpolation. We can construct the polynomial of degree at most $t-1$ by any t different points $(x_1, y_1), \ldots, (x_t, y_t)$ as

$$P(x) = \sum_{i=1}^{t} y_i l_i(x), \text{ where } l_i(x) = \prod_{j=1, j \neq i}^{t} \frac{x - x_j}{x_i - x_j} \pmod{q}. \tag{11}$$

So, the secret a_0 will be

$$a_0 = P(0) = \sum_{i=1}^{t} y_i \prod_{j=1, j \neq i}^{t} \frac{-x_j}{x_i - x_j} \pmod{q}. \tag{12}$$

3.1. *Access structure*

In a (t, n) threshold scheme, any group of t or more participants forms an authorized subset, since we assume it has the monotone property. A group of participants, which can recover the secret when they join together, is called an **authorized subset**. On the other hand, any group of participants that cannot recover the secret is called an unauthorized subset. An **access structure** $\mathcal{A}$ is a set of all authorized subsets.

Given any access structure $\mathcal{A}$, $A \in \mathcal{A}$ is called a minimal authorized subset if $A' \subsetneq A$ then $A' \notin \mathcal{A}$.

We use $\mathcal{A}_0$ to denote the set of the minimal authorized subsets of $\mathcal{A}$. In a (t, n) threshold scheme, let P be the set of the participants:

$$\mathcal{A} = \{A | A \subseteq P \text{ and } |A| \geq t\}, \tag{13}$$

$$\mathcal{A}_0 = \{A | A \subseteq P \text{ and } |A| = t\}. \tag{14}$$

In secret sharing, we first define the access structure. Then, we realize the access structure by a secret sharing scheme.

3.2. *Perfect and ideal scheme*

A Shamir (t, n) threshold scheme allows no partial information to be given out even up to $t - 1$ participants joined together [9, 29]. In other words, any group of up to $t - 1$ participants cannot get more information about the secret than any outsider. A secret sharing scheme with this property is called a **perfect scheme**.

In terms of entropy in information theory, we have

$$H(S|A) = 0, \text{ if } A \in \mathcal{A} \text{ (correctness)}, \tag{15}$$

$$H(S|A) = H(S), \text{ if } A \notin \mathcal{A} \text{ (privacy)}. \tag{16}$$

The Eq. (15) says that for an authorized subset A the entropy is equal to zero (i.e., no uncertainty) and the secret S can be determined/recovered. The Eq. (16) says that for an unauthorized subset A the entropy remains unchanged and no information about the secret S is leaked out even if the participants pool all their shares together.

Based on the information theory, the length of any share must be at least as long as the secret itself in order to have perfect secrecy. The argument is that up to $t - 1$ participants have zero information about the secret under perfect sharing scheme, but when one extra participant joins the group, the secret can be recovered. That means any participant has his share at least as long as the secret.

Following [30], the information rate for participant $p_i, i = 1, \ldots, n$, is defined as

$$\rho_i = \frac{\log_2 |\mathbb{K}|}{\log_2 |S_i|}, \tag{17}$$

where $\mathbb{K}$ is the key space, $S_i \subseteq S$ is the set of shares that p_i has. The information rate of the scheme is defined as

$$\rho = \min \{\rho_i : 1 \leq i \leq n\}. \tag{18}$$

For a perfect scheme, the information rate will be less than or equal to 1. If the shares and the secret come from the same domain, we call it an **ideal scheme**. In this case, the shares and the secret have the same size, i.e., the information rate is equal to 1.

3.3. *Proactive scheme*

In a secret sharing scheme, we need to consider the possibility that a smart adversary may find out all the shares in an authorized set to discover the secret eventually if he is given a very long time to gather the necessary information. This means that if the adversary can successfully break in t servers, in a (t, n) threshold scheme he can steal the secret. In order to prevent this from happening, we may try to reset the shares. We re-fresh and re-distribute all the shares to all the participants periodically. After finishing this phase, the old shares are erased safely and the secret remains unchanged. By doing so, an adversary has to get enough information of the shares within any two periodic resets in order to break the system. This would make it more difficult to achieve.

Based on Shamir scheme, Herzberg, Jarecki, Krawczyk, and Yung [13] derived a proactive scheme, which uses the following method to reset the shares.

Let $P(x)$ be an arbitrary polynomial of degree at most $t-1$ over $\mathbb{Z}_q$, same as in the Shamir scheme,

$$P(x) = a_0 + a_1 x + \ldots + a_{t-1}x^{t-1} \pmod q, \tag{19}$$

where q is a prime number, a_0(secret) $a_1, \ldots, a_{t-1} \in \mathbb{Z}_q$. For simplicity, let $P(1), \ldots, P(n)$ be the shares of the participants $p_1, \ldots, p_n$. The dealer generates another polynomial $Q(x)$ of degree at most $t-1$ over $\mathbb{Z}_q$ without a constant term,

$$Q(x) = b_1 x + \ldots + b_{t-1}x^{t-1} \pmod q, \tag{20}$$

where $b_1, \ldots, b_{t-1} \in \mathbb{Z}_q$. The dealer sends out $Q(1), \ldots, Q(n)$ to the participants $p_1, \ldots, p_n$, respectively. Each participant p_i will update/renew his share as $S(i) = P(i) + Q(i)$ and destroy his old share $P(i)$ safely. Here

$$S(x) = P(i) + Q(i) = a_0 + c_1 x + \ldots + c_{t-1}x^{t-1} \pmod q, \tag{21}$$

where $c_i = a_i + b_i \pmod q$ for $i = 1, \ldots, t-1$. The scheme remains a (t, n) threshold scheme with the same original secret a_0.

The above technique can be extended so that each participant p_i, by turn, generates a polynomial $P_i(x)$ of degree at most $t-1$ without a constant term and sends values of $P_i(1), \ldots, P_i(i-1), P_i(i+1), \ldots, P_i(n)$ to participants $p_1, \ldots, p_{i-1}, p_{i+1}, \ldots, p_n$, respectively. That means participant p_j will get $P_i(j)$ from participant p_i. After the above exchange process, each participant p_i resets his new shares as follows:

$$\text{newshare} = \text{oldshare} + P_1(i) + \ldots + P_n(i). \tag{22}$$

After the calculation of the new shares, all participants will destroy their old shares safely. In other words, all the participants can engage in the share renewing process. This method can eliminate all the work done by the dealer and be more secure.

3.4. *Verifiable scheme*

Shamir's original sharing scheme assumes the dealer and all the participants are honest. However, in reality, we need to consider the situation that the dealer or some of the participants might be malicious. In this case, we need to set up a verifiable scheme so that the shares of the participants can be verified to be valid. In order to make this possible, additional information is required for the participants to verify their shares' consistency.

Feldman [8] presented a simple verifiable scheme that is based on Shamir scheme. It is based on the homomorphic properties of the exponentiation function $x^{a+b} = x^a \cdot x^b$.

The idea is to find a cyclic group G of order q, where q is a prime. Since it is cyclic a generator of G, say g, exists. As other cryptographic protocols, we assume the parameters of G are carefully chosen so that the discrete logarithm problem is hard to solve in G.

Let p, q be primes such that $q | p-1$, $g \in Z_p^*$ of order q. A polynomial over Z_q of degree at most $t-1$ as a Shamir (t, n) threshold scheme is generated as

$$P(x) = a_0 + a_1 x + \ldots + a_{t-1} x^{t-1} \pmod q, \tag{23}$$

where $a_0, a_1, \ldots, a_{t-1} \in \mathbb{Z}_q$.

The dealer sends out $P(i)$ to participant i as before. In addition, he broadcasts in a public channel the commitments $g^{a_0} \pmod p$, $g^{a_1} \pmod p, \ldots, g^{a_{t-1}} \pmod p$ for the participants to verify.

Each participant P_i can verify if the following equation is true.

$$g^{P(i)} = (g^{a_0})(g^{a_1})^i(g^{a_2})^{i^2} \dots (g^{a_{t-1}})^{i^{t-1}} (\text{mod } p), i = 1, \dots, n. \tag{24}$$

Based on the homomorphic properties of the exponentiation, the above condition will hold true if the dealer sends out consistent information. If this is the case, we conclude that the dealer is honest, and the scheme is verifiable. Later, when the participants return their shares for secret recovering, the dealer can verify their shares' validity by the same method.

Feldman's scheme is not a perfect scheme since partial information, $g^{a_0}(\text{mod } p)$, is leaked out. However, we assume it is difficult to get the secret a_0 from $g^{a_0}(\text{mod } p)$ if the discrete logarithm problem is hard to solve under G.

3.5. *Enhancements by one-way function and RSA*

In order to make secret sharing schemes practical, researchers have proposed to apply one-way functions [20], hash functions [17, 31] and RSA [7, 12, 23] cryptosystems in Shamir threshold scheme. These enhancements add proactive-ness, verifiability and other desired features to Shamir scheme.

3.5.1. *Applying one-way function in Shamir scheme*

Liu *et al.* [20] enhanced the Shamir (t, n) threshold scheme by applying a one-way function. Their scheme works as follows.

Scheme setup: Suppose $f : \mathbb{Z}_q \to \mathbb{Z}_q$ is a collision-free one-way function, where $\mathbb{Z}_q$ is a finite field and $q > n$ is a large prime. (a) The dealer $\mathcal{D}$ randomly chooses n distinct elements $s_1, \dots, s_n$ in $\mathbb{Z}_q$ as shares for n participants, sends s_i to p_i via a secure channel. (b) $\mathcal{D}$ randomly chooses an element $\alpha \in \mathbb{Z}_p$ and a polynomial $P(x)$ of degree $t-1$, such that $P(0) = K$ is the secret. Dealer computes $y_i = P(f(\alpha + s_i))$, $i = 1, 2, \dots, n$. (c) $\mathcal{D}$ publishes f, α and the sequence $(y_1, y_2, \dots, y_n)$ in a public area (such as a bulletin board). All evaluations for $P(x)$ and $f(x)$ are reduced by $\text{mod}\, q$.

Secret recovery: Any t participants, say $p_1, p_2, \dots, p_t$, can recover the secret K. Every p_i gets α and their corresponding y_i from the public area. With his private share s_i (only known to him), p_i computes $x_i = f(\alpha + s_i)$ and presents x_i, the masked share, to a trusted agent $\mathcal{T}_\mathcal{A}$. After collecting t pairs of (x_i, y_i), $i = 1, \dots, t$, $\mathcal{T}_\mathcal{A}$ uses Lagrange interpolation method to recover $P(x)$, hence the secret $K = P(0)$.

The collision-free property of the one-way function f guarantees that $x_i = f(\alpha + s_i)$ will be distinct for distinct s_i, therefore $\mathcal{T_A}$ will surely get t distinct points to recover the polynomial $P(x)$. One-way function f also keeps share s_i private, a participant p_i only needs to present his masked share x_i. When the secret K needs to be replaced by a new secret K', $\mathcal{D}$ chooses element α' ($\alpha' \neq \alpha$) and a new polynomial $P'(x)$ of degree $(t-1)$ such that $K' = P'(0)$, and new $y'_i = P'(f(\alpha' + s_i))$, s_i remains the same and can be used unlimited number of times.

The scheme can be made verifiable simply adding a verifying message $v_i = f(x_i)$ in the public area for every participant p_i. $\mathcal{T_A}$ or a participant can verify the validity of any participants by this. When a new participant, say p_{n+1}, is admitted to the scheme, $\mathcal{D}$ only needs to generate s_{n+1} and appends $y_{n+1} = f(\alpha + s_{n+1})$ to the y_i sequence. When a participant p_i needs to be removed from the scheme, $\mathcal{D}$ generates another polynomial $P'(x)$ of the same degree and let $P'(0) = K$, and update the y_i sequence with the new $P'(x)$.

3.5.2. *Using one-way functions and RSA in a Shamir scheme*

Fei and Wang [7] enhanced Shamir (t, n) threshold scheme by applying one-way function and RSA cryptosystem. Their scheme works as follows.

Scheme setup: Suppose $q > n$ is a big prime, g is a primitive element of finite field $\mathbb{Z}_q$, u, w are two RSA prime numbers and $m = uw$, and f is a one-way function. (a) Dealer $\mathcal{D}$ chooses a polynomial $P(x)$ of degree $t-1$ over $\mathbb{Z}_q$, such that $K = P(0)$ is the secret to be shared among n participants $p_1, p_2, \ldots, p_n$. (b) $\mathcal{D}$ chooses an e, such that $gcd(e, \phi(m)) = 1$, and computes $d = e^{-1} \bmod \phi(m)$ (here ϕ is Euler's totient function), and publishes e. (c) $\mathcal{D}$ computes $s_i = P(g^i), v_i = (f(s_i))^d \bmod m$, and sends s_i and v_i to participant p_i as his share and verifying message.

Secret recovery: When a trusted agent $\mathcal{T_A}$ receives t points (g^1, s_1), $(g^2, s_2), \ldots, (g^t, s_t)$ from any t participants, $\mathcal{T_A}$ uses Lagrange interpolation method to reconstruct the polynomial $P(x)$, and hence the secret $K = P(0)$. Participant p_i can be verified by $v_i^e = f(s_i) \bmod m$.

4. Extension to Any General Access Structure

Ito, Saito and Nishizeki [15, 16] showed how to extend a Shamir threshold scheme to a multiple assignment scheme to realize any general access structure which fulfills the following monotone property:

$$A' \in \mathcal{A} \text{ and } A' \subseteq A'' \subseteq P \Longrightarrow A'' \in \mathcal{A}, \tag{25}$$

$$B' \in \beta \text{ and } B'' \subseteq B' \Longrightarrow B'' \subseteq \beta \tag{26}$$

where P is the set of the participants, $\mathcal{A}$ is the access structure. $\beta = 2^P - \mathcal{A}$ will be the set of all unauthorized subsets.

Following the notations in [15, 16], we give a brief discussion here. For details, please refer to [15, 16]. The family of maximal sets in $\mathcal{A}$ is defined as

$$\partial^+\mathcal{A} = \{A \subset \mathcal{A} : A \nsubseteq A' \ \forall A' \in \mathcal{A} - \{A\}\}. \tag{27}$$

Let S be the set of shares. A multiple assignment scheme assigns a subset $S_i \subseteq S$ to participant $p_i \in P$ as follows:

$$g : P \ \rightarrow \ 2^S \text{ or } g(p_i) = S_i, \forall i = 1, \ldots, n. \tag{28}$$

Define

$$\mathcal{A}(S, g, k) = \{Q \subseteq P \mid \ |\bigcup_{p \in Q} g(p)| \geq k\}. \tag{29}$$

That means if the number of distinct shares of the union of the participants in a subset Q of P is more than the threshold k, it is an authorized subset.

For any access structure $\mathcal{A} \subseteq 2^P$ satisfying the monotone property, there exist a set of shares S, an assignment function $g : P \rightarrow 2^S$ and a non-negative integer k such that $\mathcal{A}(S, g, k) = \mathcal{A}$.

<u>Proof</u>: Let $\beta = 2^P - \mathcal{A}$. We determine $\partial^+\beta$ and set up a (k, k) threshold scheme, where $k = |\partial^+\beta|$.

Construct a set of shares S so that $|S| = k$. We have $\partial^+\beta = \{\beta_1, \ldots, \beta_k\}$ and $S = \{s_1, \ldots, s_k\}$. There exists a one-to-one correspondence between S and $\partial^+\beta$, say $s_1 \leftrightarrow \beta_1, \ s_2 \leftrightarrow \beta_2, \ \ldots, s_k \leftrightarrow \beta_k$. That means $S = \{S_i, \beta_i \in \partial^+\beta, i = 1, \ldots, k\}$. We also define $g : P \rightarrow 2^S$ as follows:

$$g(p) = \{S_i, \beta_i \in \partial^+\beta, p \notin \beta_i, i = 1, \ldots, k\}. \tag{30}$$

(i) $\mathcal{A} \subseteq \mathcal{A}(S, g, k)$.

Assume there exists $Q \in \mathcal{A}$ such that $Q \notin \mathcal{A}(S, g, k)$, then $|\bigcup_{p \in Q} g(p)| < k$ and hence $\bigcup_{p \in Q} g(p) \neq S$. There exists $s_i \in S - \bigcup_{p \in Q} g(p)$ for some i. So, for every $p \in Q, s_i \notin g(p)$ and therefore $p \in \beta_i$. Hence $Q \subseteq \beta_i \in \partial^+\beta$.

By monotone property, $Q \in \beta$. This contradicts $Q \in \mathcal{A}$, since $\beta = 2^P - \mathcal{A}$.

(ii) $\mathcal{A}(S,g,k) \subseteq \mathcal{A}$.

Assume there exists $Q \in \mathcal{A}(S,g,k)$, but $Q \notin \mathcal{A}$. Since $Q \notin \mathcal{A}$, there exists $\beta_i \in \partial^+\beta$ such that $Q \in \beta$. By the definition of the function g, $s_i \notin g(p)$ for all $p \in Q$.

So, $s_i \notin \bigcup_{p \in Q} g(p)$ and hence $Q \notin \mathcal{A}(S,g,k)$. This contradicts the assumption.

Example: Let $P = \{p_1, p_2, p_3\}$ be the set of participants. Suppose $\mathcal{A} = \{\{p_1,p_2\},\{p_1,p_3\},\{p_1,p_2,p_3\}\}$, then $\beta = \{\{p_1\},\{p_2\},\{p_3\}, \{p_2,p_3\}\}$, then $\partial^+\beta = \{\{p_1\},\{p_2,p_3\}\}$.

Since $|\partial^+\beta| = 2$ we set up a $(2,2)$ threshold scheme with $S = \{s_1, s_2\}$ be the set of shares. s_1 will be assigned to participant(s) p_2, p_3 $[P - \{p_1\}]$; s_2 will be assigned to participant(s) p_1 $[P - \{p_2, p_3\}]$. It can be easily verified that all the following are unauthorized subsets $\{p_2\}, \{p_3\}, \{p_2, p_3\}$ (with s_1 only), $\{p_1\}$ (with s_2 only).

On the other hand, $\{p_1,p_2\},\{p_1,p_3\},\{p_1,p_2,p_3\}$ will have shares s_1 and s_2 to recover the secret.

5. Relation with Reed-Solomon Code

Here we discuss briefly error correction code, in particular, Reed-Solomon code. Then, we talk about the relationship or similarity between Reed-Solomon code and Shamir threshold scheme. Please refer to the textbooks for details in error correction code, for instance, [14, 21].

A $[m,q]$ code C is a mapping from a vector space of dimension m over a finite field F into a vector space of dimension q (here $q > m$) over the same field, i.e.,

$$C : F^m \to F^q; m < q. \tag{31}$$

That means an information word $a = (a_0, \ldots, a_{m-1}) \in F^m$ is mapped to a codeword $c = (c_0, \ldots, c_{q-1}) \in F^q$. There are $q - m$ extra symbols to detect or correct the errors occurred during the transmission. We call q and m the length and the dimension of the code C, respectively.

The Hamming distance between two codewords $c_1, c_2 \in C$ is defined as the number of the differences between the corresponding positions in c_1 and c_2. For example, let $c_1 = (0,0,1,1)$, $c_2 = (1,0,1,0)$. Since the first and fourth positions are different, the Hamming distance $d(c_1,c_2) = 2$. The minimum distance of C, d, is defined as

$$d = min\{d(c_1,c_2)|c_1,c_2 \in C, \text{and } c_1 \neq c_2\}. \tag{32}$$

d is important that it tells us the minimum of errors that will convert a codeword c_1 to another codeword c_2.

A code C can detect and correct up to t_1 and t_2 errors, respectively, if $t_1 \leq d-1$ and $2t_2+1 \leq d$. The error detection is based on the fact that fewer than d errors cannot convert a codeword to another codeword. The error correction is based on the nearest neighbor decoding principle. The received invalid word c' will be converted to the codeword c such that $d(c', c)$ is the smallest.

Reed-Solomon code, which is one type of error correcting codes with many applications such as compact disc (CD), spacecraft etc., was invented by Irving Reed and Gus Solomon in 1959 [25].

Let F be a field with q elements. There exists a primitive element α such that the q elements in F can be represented as $\{0, \alpha, \alpha^2, \ldots, \alpha^{q-1} = 1\}$.

Given an information word $a = (a_0, \ldots, a_{m-1})$, we set up a polynomial $P(x) = a_0 + a_1 x + \ldots + a_{m-1}x^{m-1}$, where $a_i \in F$. And the Reed-Solomon code is the mapping of the information word $a = (a_0, \ldots, a_{m-1})$ to a codeword $c = (P(0), P(\alpha), P(\alpha^2), \ldots, P(\alpha^{q-2}), P(1))$ as follows:

$$\begin{aligned} P(0) &= a_0, \\ P(\alpha) &= a_0 + a_1\alpha + a_2\alpha^2 + \ldots + a_{m-1}\alpha^{m-1}, \\ P(\alpha^2) &= a_0 + a_1\alpha^2 + a_2(\alpha^2)^2 + \ldots + a_{m-1}(\alpha^2)^{m-1}, \\ &\cdots \quad \cdots \quad \cdots \quad \cdots \\ P(\alpha^{q-2}) &= a_0 + a_1\alpha^{q-2} + a_2(\alpha^{q-2})^2 + \ldots + a_{m-1}(\alpha^{q-2})^{m-1}, \\ P(1) &= a_0 + a_1 + a_2 + \ldots + a_{m-1}. \end{aligned} \tag{33}$$

Any m correct equations without error from Eq. (33) will determine a correctly. On the other hand, any m equations from Eq. (33) with one or more errors will determine a incorrectly.

Suppose t errors occur during the transmission. There will be $\binom{q-t}{m}$ and $\binom{t+m-1}{m}$ sets of m equations that will give correct and incorrect results, respectively. By taking the majority vote for determination of the information word a, we can get the correct result if

$$\binom{q-t}{m} > \binom{t+m-1}{m}. \tag{34}$$

That is $t < \frac{q-m+1}{2}$. Please refer to [25] for details.

McEliece and Sarwate [22] pointed out that Shamir scheme is closely related to Reed-Solomon code. Suppose s pieces of P_i (Eq. (33)) are transmitted and t out of these s pieces are in error. Replacing q by s plus rearrangement and modifications in Eq. (34), we can recover $a = (a_0, a_1, \ldots, a_{m-1})$ as long as $s - 2t \geq m$. This is exactly a (m, s) threshold scheme with $t = 0$, and a_0 of a is the secret and $F = Z_q$ (q is a prime), $\alpha^i = i$. Recall that the original Shamir threshold scheme assumes the dealer and the participants are honest and $P(1), \ldots, P(s)$ are the shares of the participants.

6. Shamir Ramp Scheme

Recall that in Shamir (t, n) threshold scheme, n shares $P(x_1), \ldots, P(x_n)$ are distributed to n participants $p_1, \ldots, p_n$ so that any t out of these n participants when joined together can recover the secret. Let q be a large prime, $x_1, \ldots, x_n \in Z_q - \{0\}$ are all different to each other ($x_i \neq x_j$ if $i \neq j, 1 \leq i, j \leq n$) and chosen arbitrarily. $a_0, \ldots, a_{t-1} \in Z_q$ are chosen randomly. For simplicity, suppose $p_1, \ldots, p_t$ join together and let $y_1 = P(x_1), y_2 = P(x_2), \ldots,$ etc. We have the following t independent equations. [Note: If y_i is not available, let y_i' be its assumed value.]

$$y_1 = a_0 + a_1 x_1 + a_2 x_1^2 + \ldots + a_{t-1} x_1^{t-1} \pmod q; \tag{35}$$

$$\ldots$$

$$y_t = a_0 + a_1 x_t + a_2 x_t^2 + \ldots + a_{t-1} x_t^{t-1} (\mathrm{mod}\ q). \tag{36}$$

From Eq. (35), rewrite a_{t-1} in terms of $a_0, \ldots, a_{t-2}$ and substitute this into other equations, we reduce t equations in t unknowns into $(t-1)$ equations in $(t-1)$ unknowns. Continuing this way, we can reduce the system of t independent equations to one equation with one unknown a_0. We can solve for a_0, which is the secret.

If only $t - 1$ participants, say $p_1, \ldots, p_{t-1}$, join together, the last equation will have 2 unknowns left, namely, y_t and a_0. Any assumed or guessed value of the secret $a_0' \in Z_q$ will lead to a corresponding valid share of the missing participant $y_t' \in Z_q$, and vice versa. In other words, we can find a unique polynomial $P'(x)$ such that it will pass through all these $t-1$ points and the assumed secrets a_0'. $P'(0) = a_0', P'(1) = y_1, \ldots, P'(t-2) = y_{t-2}, P'(t-1) = y_{t-1}$. Since we cannot rule out any possibility, the scheme is perfect. The secret a_0 and the shares $y_i (i = 1, \ldots, n)$ are elements of Z_q, so it is ideal. From Eq. (17), it is obvious that the information rate is 2.

Suppose (a_0, a_1) is the secret. If $(t-2)$ participants $p_1, \ldots, p_{t-2}$ join together, we have 2 equations left:

$$y'_{t-1} = \text{ in terms of } a_1 \text{ and } a_0,$$

$$y'_t = \text{ in terms of } a_1 \text{ and } a_0.$$

Any guessed values of the secret (a'_0, a'_1) will lead to valid shares $y'_{t-1} \in Z_q$ and $y'_t \in Z_q$ of missing participants, and vice versa. So no partial information is given out here. The scheme is perfect.

Now, assume $(t-1)$ participants $p_1, \ldots, p_{t-1}$ join together. We have one equation left:

$$y'_t = \text{ in terms of } a_0. \tag{37}$$

As before, any guessed value of the share $y'_t \in Z_q$ gives a unique $a'_0 \in Z_q$. However, once a'_0 is determined, all the $a'_1, \ldots, a'_{t-1}$ are determined. We can thus eliminate the possibilities from $|Z_q| \times |Z_q|$ to $|Z_q|$. Hence, partial information is given out.

The above can be summarized by Shamir ramp scheme. For more details, please refer to [30].

A Shamir (t_1, t_2, n) ramp scheme, where $t_1 < t_2 \leq n$, is one in which n shares of information are distributed to n participants so that

(i) if t_2 or more participants join together, the secret can be recovered.
(ii) if up to t_1 participants join together, the secret cannot be recovered and no partial information about the secret is leaked out.
(iii) if t $(t_1 < t < t_2)$ participants join together, the secret cannot be recovered. However, partial information will be leaked out. The larger the t, the more information will be leaked out.

For a Shamir (t_1, t_2, n) ramp scheme, let $l = t_2 - t_1$ be the gap. The bigger the gap l, the more efficient the size of the share would be, but the lesser the secrecy the scheme will provide (see Figure 1-Right).

One implementation for a ramp scheme is also by polynomial evaluation and interpolation. Let $s = (a_0, a_1, \ldots, a_{l-1}) \in Z_q^l$. We create a polynomial of degree of at most $t_2 - 1$ as follows:

$$P(x) = a_0 + a_1 x + \ldots + a_{l-1} x^{l-1} + a_l x^l + \ldots + a_{t_2-1} x^{t_2-1} (\text{mod } q) \tag{38}$$

where $a_i \in Z_q$ will be generated randomly, $i = l, \ldots, t_2-1$. $x_i \in Z_q - \{0\}$ will be chosen arbitrarily and $P(x_i)$ will be evaluated and sent to $P_i, i = 1, \ldots, n$ as his/her share. The information rate is equal to l.

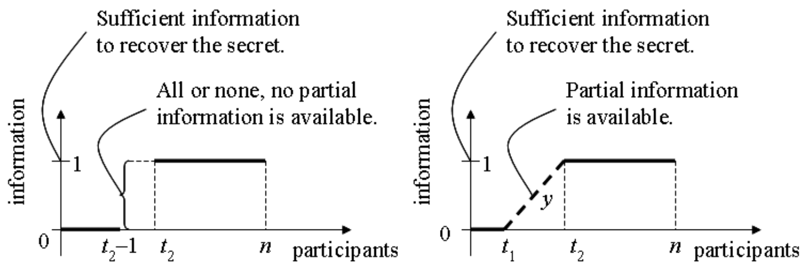

Figure 1. A Shamir (t_2, n) scheme is a $(t_2 - 1, t_2, n)$ ramp scheme.

Let us fix t_2 and n. That means any t_2 out of n participants can recover the secret. One special case is as follows: A $(t_2 - 1, t_2, n)$ ramp scheme is just the same as a (t_2, n) threshold scheme. The information rate is equal to 1 but perfect secrecy is provided. The secret will be the constant term of the polynomial. Figure 1-Left is to illustrate this.

7. Information Disposal Algorithm and Making Secret Short

Rabin [24] proposed the information disposal algorithm (IDA) in 1989. IDA is a scheme to distribute a piece of information into n participants such that any t of these participants can recover the original information while up to $(t - 1)$ participants cannot. One implementation is also by polynomial interpolation, same as the Shamir threshold scheme. In a Shamir threshold scheme, the constant term will be the secret. However, in IDA, the secret will be split into all the coefficients. In other words, the secret will be represented by the whole polynomial. This gives the optimal rate of information, but even one participant has some partial information.

A $(0, t_2, n)$ ramp scheme is an information dispersal algorithm. The information rate is optimal. But no secrecy is provided. Any participant has some partial information. The secret is made up of all the coefficients of the polynomial, as Figure 2 illustrated.

Krawczyk [19] showed a method to make the secret short and provides secrecy at the same time. Suppose we have a secure encryption (ENC_K) and decryption (DEC_K) scheme and a symmetric key K will be chosen randomly from the key space $\mathbb{K}$.

(a) We first encrypt the secret S to give a ciphertext C, i.e. $ENC_K(S) = C$. Then we use IDA to split C into $C_1, \ldots, C_n$ shares and distribute them

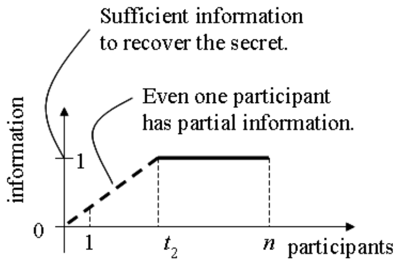

Figure 2. A (t_2, n) IDA is a $(0, t_2, n)$ ramp scheme.

to participants $p_1, \ldots, p_n$ so that each participant p_i gets one share C_i, $i = 1, \ldots, n$.

(b) We use a perfect secret sharing scheme, say a Shamir (t, n) threshold scheme, to safeguard the key K. Each participant p_i gets one share of the key K_i, $i = 1, \ldots, n$.

In this way any t participants can recover the key K and the ciphertext C. Then use K to get back the original secret S by $DEC_K(C) = S$.

The information rate is optimal. IDA helps to make the size of the share short. But it does not provide secrecy. So we need a secure encryption and decryption scheme to protect it. In turn we need a perfect secret sharing scheme to safeguard the key.

8. Secure Multiparty Computation

Secure multiparty computation (MPC), a subfield of cryptography, was first introduced in Yao's seminal two millionaire's problem [32]. The goal is to create methods for parties to jointly compute a function over their inputs while keeping those inputs private. In MPC n parties $p_1, p_2, \ldots, p_n$ join together to compute a public function $f(x_1, x_2, \ldots, x_n)$, where x_i is the private input held by party p_i, $i = 1, \ldots, n$. After the computation, each p_i will know the correct function result, the value of $f(x_1, x_2, \ldots, x_n)$, but he or she will not know the inputs of the other parties. For more MPC materials, please refer to [6].

For security reason, instead of storing a secret in a single server, we split it as shares and store in different servers. That is why secret sharing schemes are important in multiparty computation. We also want to have the computations based on the shares of the parties instead of the secrets. Let $p_1, \ldots, p_n$ be the parties and p_i holds $A(i)$ and $B(i)$ as shares for the secrets a_0 and b_0, respectively. We want to calculate $c_0 = a_0 + b_0$ based on $(A(i), B(i)),\ i = 1, \ldots, n$.

Since Shamir threshold scheme is linear, we can proceed as follows:

$$A(x) = a_0 + a_1 x + \ldots, a_{t-1} x^{t-1},\ a_i \in Z_q, \tag{39}$$

$$B(x) = b_0 + b_1 x + \ldots, b_{t-1} x^{t-1},\ b_i \in Z_q,\ \text{and} \tag{40}$$

$$\begin{aligned} C(x) = A(x) + B(x) &= c_0 + c_1 x + \ldots, c_{t-1} x^{t-1},\ \text{where} \\ c_i &= a_i + b_i, 0 \leq i \leq t-1. \end{aligned} \tag{41}$$

Any t parties (say $1, \ldots, t$) can join together to calculate $C(i) = A(i) + B(i), 1 \leq i \leq t$, and then recover c_0 which is equal to $a_0 + b_0$, the sum of the original secrets.

But for multiplication, it is different. Here,

$$D(x) = A(x)B(x) = a_0 b_0 + \ldots. \tag{42}$$

$D(x)$ will be a polynomial of degree $(t-1) + (t-1) = 2t-2$. So we need $2t-1$ parties to pull their shares to recover $a_0 b_0$, which is the product of the original secrets a_0 and b_0. Obviously, $2t-1$ can not be greater than n. So Shamir threshold scheme is multiplicative provided that $n \geq 2t-1$. Also, a linear secret sharing scheme (LSSS) is strongly multiplicative if any subset $A \subseteq P$, such that $P - A$ is not qualified, and the product $a_0 b_0$ can be computed only from the values of A. In a Shamir (t, n) threshold scheme, the maximum size of an unauthorized subset is $t-1$. So, a Shamir (t, n) threshold scheme will be strongly multiplicative if $n - (t-1) \geq 2t-1$, i.e., $3t-2 \leq n$.

9. Private Information Retrieval and Shamir Scheme

Private information retrieval (PIR) deals with the privacy of a user when he queries a public database. It was first introduced by Chor *et al.* [5] in 1995. It is formalized as follows: given a database x which consists of n bits, $x = x_1 \ldots x_n$, a user wants to inquire the ith bit without letting the database know any information about i. A trivial solution is to let the user download the entire database. In this case, the communication complexity, which is the number of bits transferred between the user and the database, is n. Chor *et al.* proved that this trivial solution turned out to be optimal for a single database in the information theoretic setting. However, Chor *et al.* further showed that if we had more than one non-colluding servers with each having a complete database, we could reduce the communication complexity and preserve the perfect privacy as well.

In PIR, a user sends out queries to a group of non-colluding databases, and then combines the answers from the databases to come up with the results. The answers from the databases act like shares from the participants, and based on that, the desired information somewhat like the secret can be obtained. In the literature, there are papers discussing the applications of secret sharing schemes to PIR. For example, Goldberg [10] proposed a Byzantine-robust PIR based on the Shamir secret sharing scheme.

10. Practical Applications

Many companies start to store their data outside their premises in cloud storage provided by various cloud providers, for instance, Amazon, Google, etc. The advantages to use cloud storage mainly include shorter setup time, lower implementation cost, easier scaling up/down, cheaper ongoing cost (pay-as-you-go). Big data has 3Vs characteristics, i.e., the velocity — the data go in and out or change very fast, the variety — different types of data (structured, semi-structured, and unstructured), and the volume - exponentially growing huge volume of data. This has been the trend for the last decade and will remain this way at least in the foreseeable future. Both cloud storage/computing and big data give rise to many big challenges to the existing data center infrastructure. They affect almost all areas to a certain extent. Here let us discuss some applications based on Shamir's secret sharing scheme and its variants.

Big data: In order to provide the data availability for the users, the traditional approach is to replicate one or more copies of data in different locations so that when one operating node goes down, the system can switch to another node so that the service will not be interrupted and is transparent to the users. However, under big data scenarios, this method is not feasible anymore. We need another efficient approach. By applying information dispersal algorithm, a large file can be separated into several smaller segments and a subset of these segments can combine to reconstruct the original file. This solves the problem of single point failure and as we saw before, the storage needed is the optimal.

Cloud storage/computing: Even if we trust a company, the data would turn out to be stored outside the premises. Privacy is a big concern to cloud storage/computing.

11. Other Platforms

Since many cryptographic protocols are based on the assumed hardness of certain mathematical problems, there is always a strong motivation to continue looking for harder problems especially after knowing that a powerful quantum computer could break RSA easily.

Since 1990, there are new proposals coming up, by using multivariate polynomials, braid group cryptography, etc. For example, Habeeb, Kahrobaei and Shpilrain [11] proposed an (n, n) secret splitting scheme construction based on non-abelian groups using n secure channels. The (n, n) scheme combined with the Shamir's idea can be further generalized to a (t, n) threshold scheme. Under this (t, n) threshold scheme, the shares of the secret are sent out to the participants over the open channels as integers in the form of tuples of words. The participants then use group-theoretic techniques to recover the integers as their shares. Then following polynomial interpolation as in Shamir's threshold scheme, any t participants can recover the polynomial and the secret.

As we mentioned earlier, Ito, Saito and Nishizeki [15, 16] showed how to extend a threshold scheme to a multiple assignment scheme to realize any general access structure, so this provides a new direction to set up any secret sharing scheme based on another platform, non-abelian groups.

12. Conclusions and Future Research

Based on a Shamir threshold scheme, many properties of secret sharing schemes can be easily demonstrated. It has a simple access structure. It is perfect and ideal. The shares distribution and secret recovery are through polynomial evaluation and polynomial interpolation, which are easy to follow. It can be further implemented as proactive or verifiable. A Shamir threshold scheme can be used as a building block to realize any general access structure. It is also closely related to Reed-Solomon code, a ramp scheme, an information dispersal algorithm and multiparty computation.

Even though the Shamir scheme was introduced more than 30 years ago, we can still use it as a building block for other cryptographic primitives and/or protocols. It has many applications in different areas such as big data and cloud storage/computing. It still remains an important active research area in the future and is worth more attention.

Another direction for research is to set up secret sharing schemes based on other alternative platforms as briefly mentioned in this paper, should this be proved more effective.

References

[1] K. Atkinson. *An Introduction to Numerical Analysis*. Wiley, 2nd edition, 1989.

[2] G.R. Blakley. Safeguarding cryptographic keys. In *Proc. of the National Computer Conference, American Federation of Information Processing Societies Proceedings 48*, pages 313–317, 1979.

[3] D. Bogdanov. Foundations and properties of Shamir's secret sharing scheme. University of Tartu, available online http://www.cs.ut.ee/~peeter_l/teaching/seminar07k/bogdanov.pdf, 2007.

[4] R.M. Capocelli, A. De Santis, L. Gargano, and U. Vaccaro. On the size of shares for secret sharing schemes. *Journal of Cryptology*, 6(3):157–167, 1993.

[5] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. In *36th Annual IEEE Symposium on Foundations of Computer Science*, pages 41–50, 1995.

[6] R. Cramer, I. Damgård, and J.B. Nielsen. Multiparty Computation, an Introduction. Lecture Notes. Available http://www.brics.dk/~jbn/smc.pdf, 2009.

[7] R. Fei and L. Wang. Cheat-proof secret sharing schemes based on rsa and one-way function. *Journal of Software*, 14(1):146–150, 2003. (In Chinese).

[8] P. Feldman. A practical scheme for non-interactive verifiable secret sharing. In *Proc. of the 28th IEEE Symposium on the Foundations of Computer Science*, pages 427–437, 1987.

[9] H. Ghodosi and R. Safavi-Naini. Remarks on the multiple assignment secret sharing scheme. In *in Proceedings of ICICS'97 –International Conference on Information and Communications Security*, pages 72–80. SpringerVerlag, 1997.

[10] I. Goldberg. Improving the robustness of private information retrieval. In *Proc. of IEEE S&P 2007*, pages 131–148, May 2007. Oakland, California.

[11] M. Habeeb, D. Kahrobaei, and V. Shpilrain. A secret sharing scheme based on group presentations and the word problem. In *Contemporary Mathematics, Volume 582 - Computational and Combinatorial Group Theory and Cryptography (American Mathematical Society)*, pages 143–150, 2012.

[12] J. He, L. Li, and X. Li. Verifiable multi-secret sharing scheme. *Acta Electronica Sinica*, 31(1):45–47, 2003. (In Chinese).

[13] A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung. Proactive secret sharing. In *Proc. of CRYPTO 1995*, volume 963 of *LNCS*, 1995.

[14] W.C. Huffman and V. Pless. *Fundamentals of Error-Correcting Codes*. Cambridge University Press, 2003.

[15] M. Ito, A. Saio, and T. Nishizeki. Multiple assignment scheme for sharing secret. *J. Cryptology*, 6:15–20, 1993.

[16] M. Ito, A. Saito, and T. Nishizeki. Secret sharing scheme realizing general access structure. In *Proc. of IEEE GLOBECOM 1987*, pages 99–102, 1987.

[17] W. Ji, S. Oh, S. Kim, and D. Won. New on-line secret sharing scheme using hash function. *Acta Electronica Sinica*, 31(1):45–47, 2003. (In Chinese).

[18] J. Katz and Y. Lindell. *Introduction to Modern Cryptography*. Chapman and Hall/CRC, 2007.
[19] H. Krawczyk. Secret sharing made short. In *CRYPTO 1993*, volume 773 of *LNCS*.
[20] H. Liu, M. Hu, B. Fang, and Y. Yang. A dynamic secret sharing scheme based on one-way function. *Journal of Software*, 13(5):1009–12, 2002. (In Chinese).
[21] F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error-Correcting Codes*. North Holland Publishing Co., 1977.
[22] R.J. McEliece and D.V. Sarwate. On sharing secrets and reed-solomon codes. *Communications of the ACM*, 24(9):583–584, 1981.
[23] L. Pang and Y. Wang. (t, n) threshold secret sharing scheme based on rsa cryptosystem. *Acta Electronica Sinica*, 31(1):45–47, 2003. (In Chinese).
[24] M.O. Rabin. Efficient dispersal of information for security, load balancing, and fault tolerance. *Journal of the ACM*, 36(2):335–348, 1989.
[25] I.S. Reed and G. Solomon. Polynomial codes over certain finite fields. *J. of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960.
[26] A. Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.
[27] C.E. Shannon. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 623–656, 1948.
[28] C.E. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28:656–715, 1949.
[29] D. Stinson. An explication of secret sharing schemes. *Design, Codes and Cryptology*, 2:357–390, 1992.
[30] D. Stinson. *Cryptography, Theory and Practice*. Chapman and Hall/CRC, 3rd edition, 2005.
[31] H. Yang and G. Lin. Security research of secret sharing schemes based on hash functions. *Computer Engineering and Design*, 27(24):4718–19, 2006. (In Chinese).
[32] A.C. Yao. Protocols for secure computations (extended abstract). In *the 21st Annual IEEE Symposium on the Foundations of Computer Science*, pages 160–164, 1982.