Journal of Information Processing Vol.25 875–883 (Sep. 2017)

[DOI: 10.2197/ipsjjip.25.875]

Recommended Paper

A Hierarchical Secret Sharing Scheme over Finite Fields of Characteristic 2

Koji Shima[1,a)] Hiroshi Doi[1,b)]

Received: November 21, 2016, Accepted: June 6, 2017

Abstract: Hierarchical secret sharing schemes are known for the way the secret is shared among a group of participants that is partitioned into levels. We examine these schemes in terms of how easily they delete a secret after it is distributed or namely for cases where the reliability of data deletion depends on deletion of the indispensable participants' share. In this paper, we consider Tassa's idea of using formal derivatives and Birkhoff interpolation so that his method will work well even over finite fields of characteristic 2, then we devise a method for derivatives. As a result, we propose a fast $(\mathbf{k}, n)$ hierarchical secret sharing scheme applicable to any level and report the software implementation evaluation. Moreover, taking practical use into consideration, we cover the optimization specialized for a $(\{1, 3\}, n)$ hierarchical secret sharing scheme.

Keywords: Secret sharing scheme, hierarchical secret sharing scheme, derivative, Birkhoff interpolation, finite fields of characteristic 2

1. Introduction

There is a strong need to securely store large amounts of secret information in our information society for both preventing information theft or leakage and preventing information loss. Secret sharing schemes are known to simultaneously satisfy the need to distribute and manage secret information so as to prevent information theft and information loss. Blakley [1] and Shamir [2] independently introduced the basic idea of a (k, n) threshold secret sharing scheme in 1979. Shamir's (k, n) threshold secret sharing scheme has the following feature: n shares are generated from the secret and each of the shares is distributed to each participant, and after that, the secret can be recovered with any k out of n shares, but cannot be recovered with less than k shares and also every subset of less than k participants cannot obtain any information about the secret. Therefore, the original secret is secure even if some of the shares leak, and it can be recovered even if some of the shares are missing.

On the other hand, several hierarchical secret sharing schemes are known for the way the secret is shared among a group of participants who are partitioned into levels. For one of those schemes, a minimal number of higher-level participants are needed in any recovery of the secret as seen in a scenario in which 3 employees are needed and at least one of them must be a department manager in order to open a bank vault. We will say the example of this scenario is called a $(\{1, 3\}, n)$ hierarchical secret sharing scheme. Tassa [3], [4] introduced polynomial derivatives to generate shares and was devoted to the question related to Birkhoff interpolation problems.

Since this hierarchical secret sharing scheme needs indispensable participants to recover the secret, from another perspective, this can be used for the purpose of easily deleting the secret. In other words, the deletion of the secret is guaranteed by the deletion of shares possessed by the indispensable participants. Taking practical and strategic use into consideration, this scheme has advantages where the reliability of data deletion depends on the deletion of the shares of the indispensable participants. In 2015, Shima et al. [5] focused on fast methods and described the possibility of a $(\{1, 3\}, n)$ hierarchical secret sharing scheme over finite fields of characteristic 2 through Tassa's idea of using derivatives. In 2016, they [6], [7] proposed a $(\{1, k\}, n)$ hierarchical secret sharing scheme over finite fields of characteristic 2 and the more general method of a hierarchical secret sharing scheme over finite fields of characteristic 2, respectively. In that year, they [8] also proposed an XOR-based $(\{1, 3\}, n)$ hierarchical secret sharing scheme specially made for 3 participants including 1 or 2 indispensable participants to recover the secret through Fujii et al.'s scheme [9] using only XOR operations.

1.1 Fast Secret Sharing Schemes and the Surroundings

Shamir's (k, n) threshold secret sharing scheme is both *perfect* and *ideal*, and applicable for arbitrary values of k and n with $k \leq n$. However, the scheme requires expensive computational costs to generate n shares and recover the secret from k shares since it needs to deal with the polynomial of degree $k - 1$.

In 2005, Fujii et al. [9] proposed a fast $(2, n)$ threshold scheme using only XOR operations to distribute and recover the secret. In 2008, Kurihara et al. [10] proposed a $(3, n)$ threshold scheme using only XOR operations. Their scheme is much more efficient than Shamir's in terms of computational cost provided that n is not too large. Moreover, Kurihara et al. [11] proposed a (k, n) threshold scheme using only XOR operations in that year and

[1] Institute of Information Security, Yokohama, Kanagawa 221–0835, Japan

[a)] dgs164101@iisec.ac.jp

[b)] doi@iisec.ac.jp

The preliminary version of this paper was published at Computer Security Symposium 2016 (CSS 2016), October 2016. The paper was recommended to be submitted to Journal of Information Processing (JIP) by the program chair of the symposium.

their scheme is more efficient than Shamir's in terms of computational cost provided that n is not extremely large. Kurihara et al. [12] briefly introduced a *ramp* scheme [13] based on their XOR-based (k, n) threshold scheme, and they [14] proposed a fast (k, L, n) *ramp* scheme in 2009. Following up on the fact that the computational cost of both multiplications and divisions over $\mathrm{GF}(2^n)$ is high compared to that of additives, in 2011 Kurihara et al. [15] presented a faster technique realizing the field operations not over $\mathrm{GF}(q^n)$ but over $\mathrm{GF}(q)$ by using the construction of Feng et al. [16] and Blömer et al. [17] for the matrix representation of finite fields. As for other secret sharing schemes, Matsuo et al. [18] presented a technique using XOR-operations. Suga [19], [20], Ke et al. [21], and Ozaki et al. [22] presented their studies and proposals, respectively. For a hierarchical secret sharing scheme, Tassa [3], [4] shows a construction where the characteristic is a large prime number, but does not clearly include the introduction to the characteristic 2 that is desirable for a faster method. Actually, there is an issue to be solved in characteristic 2 which we will describe in more detail in Section 3.2.

1.2 Our Contributions

Secret sharing schemes have been studied from various perspectives. We sought a fast method taking into account that the conditions that big data and high performance are required along with secure storage of the information. We also look at a hierarchical secret sharing scheme as an appropriate method for the ease of deleting the secret after it is distributed because the reliability of data deletion depends on the deletion of the shares of the indispensable participants, while the reliability of data deletion depends on the deletion of more than $n - k$ shares for (k, n) threshold secret sharing schemes.

In this paper, we present a hierarchical secret sharing scheme applicable to any level over finite fields of characteristic 2 through Tassa's idea of using derivatives and Birkhoff interpolation. Our contribution can be summarized as follows:

- Our contribution provides the missing piece for the case of characteristic 2 which the original Tassa scheme lacks. As we will see in more detail in Section 3.2, meaningful shares cannot be generated as long as we apply Tassa's method [3], [4] as is to the finite fields of characteristic 2 because the k-th derivative of $p(x)$ always results in $p^{(k)}(x) = 0$ where $k \geq 2$. We introduce a new technique.
- Our scheme has a firm mathematical basis, so that the Birkhoff interpolation works with modification where the derivative of a polynomial is replaced with our function.
- In practice, our scheme achieves a high throughput or speed due to the binary operations in characteristic 2. Taking practical and strategic use into consideration, a $(\{1, k\}, n)$ hierarchical secret sharing scheme, satisfying at least 1 authority in the highest level, will be useful, especially with $k = 3$. Our scheme also achieves the same effect of Ref. [6].

2. Preliminaries

2.1 A Perfect Secret Sharing Scheme

Beimel [23] shows in its Definition 2 and Definition 3 that a *perfect* secret sharing scheme requires the following conditions:

Correctness, Accessibility Every authorized set B in an access structure gets the information on the secret.

Perfect privacy, Perfect security Every unauthorized set T out of an access structure gets no information on the secret.

In other words, let S be a random variable in a given probability distribution on the secret, S_B be a random variable in a given probability distribution on the shares for every authorized set B, and S_T be a random variable in a given probability distribution on the shares for every unauthorized set T. $H(X)$ denotes Shannon's entropy of a random variable X. A *perfect* secret sharing scheme requires the following conditions:

Correctness, Accessibility $H(S|S_B) = 0$.

Perfect privacy, Perfect security $H(S|S_T) = H(S)$.

2.2 An Ideal Secret Sharing Scheme

We refer to each paper of Blundo et al. [24], [25] and Kurihara et al. [10], [11], [12]. Let $\mathcal{P} = \{P_1, \cdots, P_n\}$ be a set of n participants. The dealer selects a secret $s \in \mathcal{S}$ and gives a share $w_i \in \mathcal{W}_i$ to every participant $P_i \in \mathcal{P}$ where $\mathcal{S}$ denotes the set of secrets and $\mathcal{W}_i$ denotes the set of possible shares that P_i might receive. The information rate is defined as $\rho = \dfrac{H(S)}{\max\limits_{P_i \in \mathcal{P}} H(W_i)}$ where S and W_i denote the random variables induced by $s \in \mathcal{S}$ and $w_i \in \mathcal{W}_i$, respectively. When the probability distributions over $\mathcal{S}$ and the shares $\mathcal{W}_i$ are uniform, the information rate

$$\rho = \frac{\log_2 |\mathcal{S}|}{\max\limits_{P_i \in \mathcal{P}} \log_2 |\mathcal{W}_i|}$$

will be measured and a secret sharing scheme is called *ideal* if it is *perfect* and $\rho = 1$. In other words, if every bit size of shares equals the bit size of the secret, the scheme is *ideal*. As Tassa [4] mentioned in its Definition 1.1, we may apply the information rate to a hierarchical secret sharing scheme.

3. Related Work

Tassa [3], [4] defines a $(\mathbf{k}, n)$ hierarchical secret sharing scheme where a minimal number of higher-level participants are needed for any recovery of the secret. Let $\mathcal{U}$ be a set of n participants and assume that $\mathcal{U}$ is composed of levels, that is, $\mathcal{U} = \bigcup_{i=0}^{m} \mathcal{U}_i$ where $\mathcal{U}_i \bigcap \mathcal{U}_j = \emptyset$ for all $0 \leq i < j \leq m$. The $(\mathbf{k}, n)$ hierarchical secret sharing scheme with $\mathbf{k} = \{k_i\}_{i=0}^{m}$ where $0 < k_0 < \cdots < k_m$ generates each share of the participants $u \in \mathcal{U}$ satisfying the access structure

$$\Gamma = \left\{ \mathcal{V} \subset \mathcal{U} : \left| \mathcal{V} \cap \left(\bigcup_{j=0}^{i} \mathcal{U}_j \right) \right| \geq k_i, \forall i \in \{0, 1, \cdots, m\} \right\}. \quad (1)$$

Given $\mathbf{k} = \{1, 3\}$ as an example, it means a $(\{1, 3\}, n)$ hierarchical secret sharing scheme that consists of two levels in the hierarchy and that requires at least 1 indispensable participant from $\mathcal{U}_0$ and 3 or more participants from $\mathcal{U}_0 \bigcup \mathcal{U}_1$ to recover the secret.

Tassa's scheme is both *perfect* and *ideal* and does not allow just the lower-level participants to recover the secret. The secret is the free coefficient or constant term of some polynomial $p(x)$ of degree $k - 1$ over a large finite field in the same manner as Shamir's (k, n) threshold scheme and $k = k_m$ is the maximal threshold. Each participant $u \in \mathcal{U}$ is given an identity in the field, denoted also by u, and a share that equals $p^{(j)}(u)$ for some derivative order j that depends on the position of u in the hierarchy. The more important participants belong to levels with a lower index and they will get shares with lower derivative orders. We are able to meet the access structure Eq. (1) by selecting the derivative orders properly.

Tassa also shows other hierarchical settings studied by other authors before. Shamir [2] suggested accomplishing this by giving the participants of the more capable levels a greater number of shares. Levels are represented in the subset of participants, however, the necessary number of participants for recovery is determined by a weighted average of the thresholds that are associated with each of the levels. In other words, when any subset of lower-level participants is sufficiently large, only the lower-level participants can recover the secret. Simmons [26] and Brickell [27] considered other hierarchical settings, respectively. However, the necessary number of participants is the highest of the thresholds that are associated with the levels. Therefore, their hierarchical settings cannot meet the scenario in which a minimal number of higher-level participants need to be involved in any recovery of the secret.

Selçuk et al. [30] proposed a function called the truncated version instead of the derivative of a polynomial to achieve the hierarchy. The truncated version truncates the polynomial from the lowest order term depending on the level. For example, we obtain $p_1(x) = a_2x^2 + a_1x$ from $p(x) = a_2x^2 + a_1x + a_0$. Our function is different from the truncated version as we will describe in detail in our function later on in Section 4.1. Another difference is that we have shown that our function has a firm mathematical basis as shown later in Section 4.2, so that the Birkhoff interpolation works with modification where the derivative of a polynomial is replaced with our function. In other words, our function works by Birkhoff interpolation to recover the secret.

Käsper [31] investigated the *multiplicativity* of hierarchical schemes. Multiplicativity allows participants, holding shares of two secrets s_0 and s_1, to privately compute shares of the product s_0s_1 without revealing the original secrets.

3.1 Polynomial Interpolation

When we use a polynomial interpolation in order to recover the secret instead of solving the simultaneous equations, we contribute to a smaller calculation load. However, when a derivative value is included as a share, the secret cannot be recovered with either Lagrange interpolation or Newton's interpolation. Hermite interpolation can deal with the derivative value, but using Hermite interpolation puts a restriction on the distribution of the share because not only $p'(x_1)$ but also $p(x_1)$ needs to be given as a share. Birkhoff interpolation can resolve that restriction.

3.1.1 Birkhoff Interpolation

Let $G = \{g_0, g_1, \cdots, g_N\}$ be a system of linearly independent, n times continuously differentiable real-valued functions in an interval $[a, b]$. A linear combination $P = \sum_{k=0}^{N} a_k g_k$ with real a_k will be called a polynomial in the system G. There exists an $m\times(n+1)$ interpolation matrix

$$E = [e_{i,j}]_{i=1,}^{m}{}_{j=0}^{n}, \quad m \geq 1, n \geq 0.$$

Its elements $e_{i,j}$ are 0 or 1 and $\sum e_{i,j} = N + 1$ but no empty rows or namely an i for which $e_{i,j} = 0, j = 0, \cdots, n$. Then, let $X = \{x_1, \cdots, x_m\}$ be a given set of m distinct points in $[a, b]$, where $x_1 < \cdots < x_m$. The Birkhoff interpolation problem [28], [29] that corresponds to the triplet $\langle E, X, G\rangle$ and given data $c_{i,j}$ must find a polynomial p of degree at most n, that satisfies the conditions

$$p^{(j)}(x_i) = c_{i,j}, \quad e_{i,j} = 1. \tag{2}$$

The system Eq. (2) consists of $N + 1$ linear equations. The triplet $\langle E, X, G\rangle$ has a unique solution for each given set of $c_{i,j}$ if and only if the determinant of the system

$$D(E, X, G) = \det[g_0^{(j)}(x_i), \cdots, g_N^{(j)}(x_i); e_{i,j} = 1] \tag{3}$$

is different from 0. Equation (3) displays only one row of the determinant or namely the row corresponding to a pair (i, j) with $e_{i,j} = 1$. We denote the $(N + 1) \times (N + 1)$ matrix that appears in Eq. (3) as $A(E, X, G)$, the determinant is equal to $D(E, X, G) = |A(E, X, G)|$. When $c_{i,j} = p^{(j)}(x_i)$ are given, the interpolation polynomial is given by

$$p(x) = \sum_{j=0}^{N} \frac{D(E, X, G_j)}{D(E, X, G)} \cdot g_j(x) \tag{4}$$

where G_j is the set of functions obtained from G by replacing g_j with p, e.g., $G_1 = \{g_0, p, g_2, \cdots, g_N\}$.

3.1.2 An Example for Birkhoff Interpolation

$g_0(x) = 1, g_1(x) = x, g_2(x) = x^2$, that is, $G = \{1, x, x^2\}$ are given. X and E are also given as follows:

$$X = \{1, 2, 3\}, \quad E = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

When the data $c_{i,j}$ are actually given as $p(1) = 15$, $p(2) = 29$, $p'(3) = 23$, we look for a polynomial $p(x) = \sum_{j=0}^{2} a_j x^j$ satisfying $p(1) = c_{1,0} = 15$, $p(2) = c_{2,0} = 29$, $p'(3) = c_{3,1} = 23$.

$$D(E, X, G) = \begin{vmatrix} g_0(x_1) & g_1(x_1) & g_2(x_1) \\ g_0(x_2) & g_1(x_2) & g_2(x_2) \\ g_0'(x_3) & g_1'(x_3) & g_2'(x_3) \end{vmatrix} = \begin{vmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 0 & 1 & 6 \end{vmatrix} = 3,$$

$$D(E, X, G_0) = \begin{vmatrix} p(x_1) & g_1(x_1) & g_2(x_1) \\ p(x_2) & g_1(x_2) & g_2(x_2) \\ p'(x_3) & g_1'(x_3) & g_2'(x_3) \end{vmatrix} = \begin{vmatrix} 15 & 1 & 1 \\ 29 & 2 & 4 \\ 23 & 1 & 6 \end{vmatrix} = 21,$$

$$D(E, X, G_1) = \begin{vmatrix} g_0(x_1) & p(x_1) & g_2(x_1) \\ g_0(x_2) & p(x_2) & g_2(x_2) \\ g_0'(x_3) & p'(x_3) & g_2'(x_3) \end{vmatrix} = \begin{vmatrix} 1 & 15 & 1 \\ 1 & 29 & 4 \\ 0 & 23 & 6 \end{vmatrix} = 15,$$

$$D(E, X, G_2) = \begin{vmatrix} g_0(x_1) & g_1(x_1) & p(x_1) \\ g_0(x_2) & g_1(x_2) & p(x_2) \\ g_0'(x_3) & g_1'(x_3) & p'(x_3) \end{vmatrix} = \begin{vmatrix} 1 & 1 & 15 \\ 1 & 2 & 29 \\ 0 & 1 & 23 \end{vmatrix} = 9,$$

$$p(x) = \sum_{j=0}^{2} \frac{D(E, X, G_j)}{D(E, X, G)} \cdot g_j(x) = 7 + 5x + 3x^2.$$

We consider a hierarchical secret sharing scheme over a prime field GF(p) where the prime number is sufficiently large. For example, the shares are given by $p(x) = 3x^2 + 5x + 7 \bmod p$ and $f'(x) = 6x + 5 \bmod p$, and also $p(1) = 15, p(2) = 29, p'(3) = 23$

are available for recovering the secret, we obtain the secret s with

$$s = p(0) = \frac{D(E, X, G_0)}{D(E, X, G)} = \frac{21}{3} = 7.$$

3.2 An Issue with Tassa's Method for Finite Fields of Characteristic 2

As stated in Ref. [6], when we differentiate a polynomial $f(x)$ over finite fields of characteristic 2, every term that is an even degree will disappear since differentiating x^i over the extension field such that i is an even number results in 0. Given $p(x) = a_2x^2 + a_1x + a_0 \in \mathrm{GF}(2^L)[x]$, where $a_2, a_1, a_0 \in \mathrm{GF}(2^L)$, as an example, we get $p'(x) = a_1$ and $p''(x) = 0$. For that reason, Tassa's method does not work as expected over finite fields of characteristic 2. Therefore, in order to realize a $(\{1, k\}, n)$ hierarchical secret sharing scheme, we choose a polynomial $p(x) \in \mathrm{GF}(2^L)[x]$ that consists of the free coefficient and $k - 1$ terms that are an odd degree with random coefficients, that is,

$$p(x) = \sum_{i=1}^{k-1} a_i x^{2(i-1)+1} + s \in \mathrm{GF}(2^L)[x].$$

However, when we consider the more general method of a $(\{k_0, k_1\}, n)$ hierarchical secret sharing scheme where $2 \leq k_0 < k_1$, we cannot realize such a hierarchical secret sharing scheme effectively because the k_0-th derivative is needed and the k_0-th derivative of $p(x)$ always results in $p^{(k_0)}(x) = 0$.

4. Our Proposed Scheme

We describe the proposed $(\mathbf{k}, n)$ hierarchical secret sharing scheme over finite fields of characteristic 2 where $\mathbf{k} = \{k_i\}_{i=0}^{m}$, $0 < k_0 < \cdots < k_m$, and $k = k_m$ is the maximal threshold. The access structure is the same as Eq. (1).

4.1 Generalization

We consider the issue shown in Section 3.2, or namely, a $(\{k_0, k_1\}, n)$ hierarchical secret sharing scheme where $k_0 \geq 2$ again. The issue is that the k_0-th derivative of $p(x)$ always results in $p^{(k_0)}(x) = 0$. Therefore, what we need to look at is whether we can give a polynomial such that meaningful shares can be generated in the k_0-th derivative where $k_0 \geq 2$, and we reconsider the need to use derivatives. In order to realize the hierarchy, it is important that the constant term of the polynomial disappears every time we differentiate it, but we begin to realize that there is no need to use the definition of the derivative itself as follows:

$$f^{(n)}(x) = \begin{cases} \displaystyle\sum_{i=0}^{k-1-n} {}_{i+n}\mathrm{P}_n \cdot a_{i+n}x^i & (k-1 \geq n) \\ 0 & (k-1 < n) \end{cases},$$

$${}_n\mathrm{P}_r = \frac{n!}{(n-r)!} = n(n-1)\cdots(n-r+1),$$

where $n \geq 0$ here and $k = k_m$. Therefore, we do not use the n-th derivative $f^{(n)}(x) \in \mathrm{GF}(2^L)[x]$ of the polynomial $f(x)$ of degree $k-1$, but we instead define a function $f^{[n]}(x) \in \mathrm{GF}(2^L)[x]$ that is used to reduce each exponent of the variable x in the polynomial $f(x)$ n times, and the function $f^{[n]}(x)$ is hereinafter referred to as the n-th order reduction of $f(x)$,

$$f^{[n]}(x) = \begin{cases} \displaystyle\sum_{i=0}^{k-1-n} a_{i+n}x^i & (k-1 \geq n) \\ 0 & (k-1 < n) \end{cases}, \tag{5}$$

where $n \geq 0$ here and $k = k_m$. This function is used to realize the hierarchy. For example, given $f(x) = a_2x^2 + a_1x + a_0$, we get $f^{[1]}(x) = a_2x + a_1$ and $f^{[2]}(x) = a_2$.

4.2 Birkhoff Interpolation Using $f^{[n]}(x)$

We consider whether Birkhoff interpolation can be applied to the hierarchical secret sharing scheme using the n-th order reduction $f^{[n]}(x)$ to recover the secret. If Eq.(4) holds even when $f^{[n]}(x)$ is used, we may apply to it Birkhoff interpolation to recover the secret. We then introduce Theorem 4.1 from Lemma 4.1 and Lemma 4.2.

Lemma 4.1. *Suppose $F = [a(i, j)]$ is an $n \times n$ matrix and fix any $i, j \in \{1, 2, \cdots, n\}$. The following equation then holds for any $1 \leq i \leq n$, $1 \leq k \leq n$.*

$$\sum_{j=1}^{n} (-1)^{i+j} |\tilde{F}(i, j)| a(k, j) = \begin{cases} |F| & (k = i) \\ 0 & (k \neq i) \end{cases} \tag{6}$$

where $\tilde{F}(i, j)$ is the $(n-1) \times (n-1)$ submatrix of F formed by deleting the i-th row and the j-th column.

Proof. When an i is chosen from any $1 \leq i \leq n$, we prove that Eq. (6) holds for $k = i$ and $k \neq i$, respectively.

For the case of $k = i$, the left-hand side is the value of the cofactor expansion, also called the Laplace expansion, of $|F|$ along the i-th row. Thus the equation holds for $k = i$.

For the case of $k \neq i$, when we do not care about the sign of the whole expression of the left-hand side itself, it is the value of the cofactor expansion along the i-th row for the determinant of an $n \times n$ matrix

$$F = \begin{array}{c} \left[\begin{array}{ccc} a(1,1) & \ldots & a(1,n) \\ \vdots & & \vdots \\ a(k,1) & \ldots & a(k,n) \\ \vdots & & \vdots \\ a(n,1) & \ldots & a(n,n) \end{array} \right] \end{array} \begin{array}{l} \text{the 1st row} \\ \vdots \\ \text{the } i\text{-th row} \\ \vdots \\ \text{the } n\text{-th row} \end{array},$$

where the i-th row is $[a(k, 1), \cdots, a(k, n)]$ and the $l(\neq\ i)$-th row is $[a(l, 1), \cdots, a(l, n)]$. However, the k-th row is $[a(k, 1), \cdots, a(k, n)]$. Thus the equation holds for $k \neq i$ since whenever two rows of a matrix are identical, its determinant is 0. □

Lemma 4.2. *Suppose $F = [a(i, j)]$ is an $n \times n$ matrix and $FQ(j)$ is an $n \times n$ matrix where the j-th column of the matrix F is replaced with $[q(1), \cdots, q(n)]^{\mathrm{T}}$, that is,*

$$FQ(j) = \begin{bmatrix} a(1,1) & \ldots & q(1) & \ldots & a(1,n) \\ \vdots & & \vdots & & \vdots \\ a(n,1) & \ldots & q(n) & \ldots & a(n,n) \end{bmatrix}.$$

Then the following equation holds for any $1 \leq k \leq n$.

$$\sum_{j=1}^{n} |FQ(j)| a(k, j) = q(k) \times |F|. \tag{7}$$

Proof. The cofactor expansion along the j-th column for $|FQ(j)|$

yields $\sum_{i=1}^{n}(-1)^{i+j}|\tilde{F}(i,j)|q(i)$. The left-hand side of Eq. (7) is expanded by using Lemma 4.1 as follows:

$$\begin{aligned}\sum_{j=1}^{n}|FQ(j)|a(k,j) &= \sum_{j=1}^{n}\sum_{i=1}^{n}(-1)^{i+j}|\tilde{F}(i,j)|q(i)a(k,j)\\ &= \sum_{i=1}^{n}\sum_{j=1}^{n}(-1)^{i+j}|\tilde{F}(i,j)|q(i)a(k,j)\\ &= \sum_{i=1}^{n}q(i)\sum_{j=1}^{n}(-1)^{i+j}|\tilde{F}(i,j)|a(k,j)\\ &= q(k)\times|F|\end{aligned}$$

The proof is therefore complete. □

Theorem 4.1. *Assume that $D(E,X,G) \neq 0$. Equation (4), Birkhoff interpolation, holds even when the n-th order reduction $f^{[n]}(x)$ is used instead of the n-th derivative $f^{(n)}(x)$.*

Proof. The equation that needs to be satisfied is

$$p(x)\cdot D(E,X,G) = \sum_{j=0}^{N} D(E,X,G_j)\cdot g_j(x). \tag{8}$$

Note that Eq. (8) is equivalent to

$$p^{[j]}(x)\cdot D(E,X,G) = \sum_{j=0}^{N} D(E,X,G_j)\cdot g_j^{[j]}(x)$$

since $p^{[j]}(x)$ is uniquely determined by the polynomial of $p(x)$ under the n-th order reduction.

In the $(\mathbf{k},n)$ hierarchical secret sharing scheme with $\mathbf{k}=\{k_i\}_{i=0}^{m}$, Eq. (8) must hold for $x = x_1,\cdots,x_{k_m}$, where $N = k_m - 1$. Without loss of generality, we may assume that the 1st, $\cdots$, k_0-th participants in the highest level $\mathcal{U}_0$, the (k_0+1)-th, $\cdots$, k_1-th participants in the level $\mathcal{U}_1$, and participants up to the level $\mathcal{U}_m$ in the same manner are assigned. Then the relation between $q(i)$ of Lemma 4.2 and a share passed into Birkhoff interpolation can be represented as follows:

$$\begin{aligned}&q(1) = p(x_1),\cdots,q(k_0) = p(x_{k_0}),\\ &q(k_0+1) = p^{[k_0]}(x_{k_0+1}),\cdots,q(k_1) = p^{[k_0]}(x_{k_1}),\\ &\quad\vdots\\ &q(k_{m-1}+1) = p^{[k_{m-1}]}(x_{k_{m-1}+1}),\cdots,q(k_m) = p^{[k_{m-1}]}(x_{k_m}).\end{aligned}$$

Moreover, let F be a $k_m \times k_m$ matrix as follows:

$$F = \begin{bmatrix} a(1,j)\\ \vdots\\ a(k_0,j)\\ a(k_0+1,j)\\ \vdots\\ a(k_1,j)\\ \vdots\\ a(k_{m-1}+1,j)\\ \vdots\\ a(k_m,j)\end{bmatrix} = \begin{bmatrix} g_{j-1}(x_1)\\ \vdots\\ g_{j-1}(x_{k_0})\\ g_{j-1}^{[k_0]}(x_{k_0+1})\\ \vdots\\ g_{j-1}^{[k_0]}(x_{k_1})\\ \vdots\\ g_{j-1}^{[k_{m-1}]}(x_{k_{m-1}+1})\\ \vdots\\ g_{j-1}^{[k_{m-1}]}(x_{k_m})\end{bmatrix}$$

where only the $j(=1,\cdots,k_m)$-th column is shown. $D(E,X,G)$ then equals the determinant of the matrix F and $D(E,X,G_j)$ equals $|FQ(j+1)|$, where $j = 0,\cdots,k_m-1$. When we note that $k = 1,\cdots,k_m$ correspond to $x = x_1,\cdots,x_{k_m}$, respectively, we obtain

$$\begin{aligned}\sum_{j=0}^{N=k_m-1} D(E,X,G_j)\cdot g_j(x) &= \sum_{j=1}^{k_m}|FQ(j)|\cdot a(k,j)\\ &= q(k)\times|F|\\ &= p(x)\cdot D(E,X,G).\end{aligned}$$

The proof is thus complete since we have confirmed that Birkhoff interpolation works well by using Lemma 4.2. □

Table 1 Participants and the shares for recovery.

Participant	Share
$u_1 \in \mathcal{U}_0$	$p(u_1)$
$u_2 \in \mathcal{U}_0$	$p(u_2)$
$u_3 \in \mathcal{U}_0$	$p(u_3)$
$u_4 \in \mathcal{U}_1$	$p^{[3]}(u_4)$
$u_5 \in \mathcal{U}_1$	$p^{[3]}(u_5)$
$u_6 \in \mathcal{U}_2$	$p^{[4]}(u_6)$

Theorem 4.1 means that there is no need for shares to be generated with the definition of the derivative itself and that the n-th order reduction $f^{[n]}(x)$ works in Birkhoff interpolation to recover the secret.

4.3 Distribution and Recovery

We describe the $(\mathbf{k},n)$ hierarchical secret sharing scheme satisfying the access structure Eq. (1), where $\mathbf{k} = \{k_i\}_{i=0}^{m}, 0 < k_0 < \cdots < k_m$. The total number of the participants required to recover the secret is $k = k_m$, the maximal threshold.

4.3.1 Distribution Algorithm

The dealer selects a random polynomial

$$p(x) = \sum_{i=0}^{k-1} a_i x^i \in \mathrm{GF}(2^L)[x], \quad a_0 = s.$$

The dealer identifies each participant $u \in \mathcal{U}$ with an element of $\mathrm{GF}(2^L)$. For simplicity, the element that corresponds to $u \in \mathcal{U}$ will be also denoted by u.

The dealer securely distributes shares to all participants in the following manner: Each participant $u \in \mathcal{U}_i$ receives the share $p^{[k_{i-1}]}(u)$, where $k_{-1} = 0$ and $p^{[n]}(x)$ is the definition Eq. (5).

We describe the $(\{3,4,6\},n)$ hierarchical secret sharing scheme as an example, where $k_0 = 3, k_1 = 4, k_2 = 6$. As $k = k_2 = 6$, the dealer selects a random polynomial of degree 5, $p(x) = \sum_{i=1}^{5} a_i x^i + s \in \mathrm{GF}(2^L)[x]$. Then the dealer distributes the shares, that is, each participant $u \in \mathcal{U}_0$ will get the share $p(u)$, each participant $u \in \mathcal{U}_1$ will get the share $p^{[3]}(u) = \sum_{i=0}^{2} a_{i+3}u^i$, and each participant $u \in \mathcal{U}_2$ will get the share $p^{[4]}(u) = \sum_{i=0}^{1} a_{i+4}u^i$.

4.3.2 Recovery Algorithm

k participants that satisfy the access structure Eq. (1) cooperate to recover the secret s. The secret is recovered from the shares, including the n-th order reduction $p^{[n]}(x)$, by using Theorem 4.1 and Birkhoff interpolation as follows:

$$s = p(0) = \frac{D(E,X,G_0)}{D(E,X,G)}.$$

We describe the $(\{3,4,6\},n)$ hierarchical secret sharing scheme as an example, where $k_0 = 3, k_1 = 4, k_2 = 6$. **Table 1** shows 6 participants that includes 3 participants from $\mathcal{U}_0$ and 5 participants from $\mathcal{U}_0 \cup \mathcal{U}_1$ agree to recover the secret. Note that at least 4 participants from $\mathcal{U}_0 \cup \mathcal{U}_1$ are required to recover the secret. We obtain the secret s with the following $D(E, X, G)$ and $D(E, X, G_0)$,

$$D(E, X, G) = \begin{vmatrix} 1 & u_1 & u_1^2 & u_1^3 & u_1^4 & u_1^5 \\ 1 & u_2 & u_2^2 & u_2^3 & u_2^4 & u_2^5 \\ 1 & u_3 & u_3^2 & u_3^3 & u_3^4 & u_3^5 \\ 0 & 0 & 0 & 1 & u_4 & u_4^2 \\ 0 & 0 & 0 & 1 & u_5 & u_5^2 \\ 0 & 0 & 0 & 0 & 1 & u_6 \end{vmatrix},$$

$$D(E, X, G_0) = \begin{vmatrix} p(u_1) & u_1 & u_1^2 & u_1^3 & u_1^4 & u_1^5 \\ p(u_2) & u_2 & u_2^2 & u_2^3 & u_2^4 & u_2^5 \\ p(u_3) & u_3 & u_3^2 & u_3^3 & u_3^4 & u_3^5 \\ p^{[3]}(u_4) & 0 & 0 & 1 & u_4 & u_4^2 \\ p^{[3]}(u_5) & 0 & 0 & 1 & u_5 & u_5^2 \\ p^{[4]}(u_6) & 0 & 0 & 0 & 1 & u_6 \end{vmatrix}.$$

4.4 Perfect Privacy

We describe that the perfect security shown in Section 2.1 holds for the $(\mathbf{k}, n)$ hierarchical secret sharing scheme, where $\mathbf{k} = \{k_i\}_{i=0}^m$ and $k = k_m$. The proof is based on Tassa's approach in his Section 3.1 [4]. The main difference is that the n-th order reduction instead of the n-th order derivative is used in the coefficient matrix $M_{\mathcal{V}}$ to generate each participant's share that agrees to recover the secret.

Theorem 4.2. *Assume that the corresponding square matrix $M_{\mathcal{V}}$ including values of the n-th order reduction is regular,* $\det(M_{\mathcal{V}}) \neq 0$*, for any minimal authorized subset $\mathcal{V} \in \Gamma$, namely, $|\mathcal{V}| = k$. Then perfect security holds.*

Proof. First, a square matrix is regular, also called nonsingular, if and only if its determinant is nonzero. Equivalently, the rows of $M_{\mathcal{V}}$ are linearly independent. Let $\mathcal{V}_u \notin \Gamma$ be an unauthorized subset and $M_{\mathcal{V}_u}$ be the corresponding matrix. We aim at showing that even if all participants in $\mathcal{V}_u$ pool their shares together, they cannot reveal anything about the secret s. This also implies that any value of s is accepted from their shares. The proof is that the secret is not included in the row space of $M_{\mathcal{V}_u}$, in the set of all possible linear combinations of the rows of $M_{\mathcal{V}_u}$.

Without loss of generality, we may assume that $\mathcal{V}_u$ is missing only one participant in order to become authorized and we may boil the process down to adding to $\mathcal{V}_u$ the phantom participant $0 \in \mathcal{U}_0$ so that we can get an authorized subset. Then the square matrix corresponding to the authorized subset is regular by the assumption, the rows of the square matrix are linearly independent. Consequently, the share of the participant $0 \in \mathcal{U}_0$ cannot be generated from $\mathcal{V}_u$, and the share is equivalent to the secret itself. In addition, even when $\mathcal{V}_u$ is missing only one participant in the j-th level, the access structure holds for adding one higher-level participant, that is, the highest-level participant $0 \in \mathcal{U}_0$.

Next, let $\mathcal{V} = \{v_1, \cdots, v_{|\mathcal{V}|}\} \subset \mathcal{U} = \bigcup_{i=0}^m \mathcal{U}_i$ and assume that all the participants are assigned as follows:

$$\begin{aligned} &v_1, \cdots, v_{l_0} \in \mathcal{U}_0, \\ &v_{l_0+1}, \cdots, v_{l_1} \in \mathcal{U}_1, \\ &\quad \vdots \\ &v_{l_{m-1}+1}, \cdots, v_{l_m} \in \mathcal{U}_m, \end{aligned}$$

where $0 \leq l_0 \leq \cdots \leq l_m = |\mathcal{V}|$. $\mathcal{V}$ satisfies the access structure Eq. (1) if and only if $l_i \geq k_i$ for all $0 \leq i \leq m$. The distribution of shares in Section 4.3 is represented as

$$\sigma(u) = \mathbf{r}^{(k_{i-1})}(u) \cdot \mathbf{a}$$

where the share $\sigma(u)$ of the participant $u \in \mathcal{U}_i$ and $\mathbf{a} = (s, a_1, \cdots, a_{k-1})^{\mathrm{T}}$ is the vector of coefficients of $p(x)$. Also, let $\mathbf{r}(x) = (1, x, x^2, \cdots, x^{k-1})$ and let $\mathbf{r}^{(i)}(x)$ for all $i \geq 0$ denote the i-th order reduction Eq. (5) of that vector $\mathbf{r}(x)$. For example, $\mathbf{r}^{(1)}(x) = (0, 1, x, \cdots, x^{k-2})$. When all participants $v_1, \cdots, v_{l_m}$ of $\mathcal{V}$ pool together their shares of $\boldsymbol{\sigma} = (\sigma(v_1), \cdots, \sigma(v_{l_m}))^{\mathrm{T}}$, they need to solve $\boldsymbol{\sigma} = M_{\mathcal{V}} \cdot \mathbf{a}$ in the unknown vector $\mathbf{a}$, or in other words, the secret s is obtained from

$$\begin{bmatrix} \sigma(v_1) \\ \vdots \\ \sigma(v_{l_m}) \end{bmatrix} = M_{\mathcal{V}} \begin{bmatrix} s \\ a_1 \\ \vdots \\ a_{k-1} \end{bmatrix}, M_{\mathcal{V}} = \begin{bmatrix} \mathbf{r}(v_1) \\ \vdots \\ \mathbf{r}(v_{l_0}) \\ \mathbf{r}^{(k_0)}(v_{l_0+1}) \\ \vdots \\ \mathbf{r}^{(k_0)}(v_{l_1}) \\ \cdots \\ \mathbf{r}^{(k_{m-1})}(v_{l_{m-1}+1}) \\ \vdots \\ \mathbf{r}^{(k_{m-1})}(v_{l_m}) \end{bmatrix}.$$

The proof is thus complete and $\det(M_{\mathcal{V}}) \neq 0$ is required for perfect security. □

4.5 Rate of Unrecoverable Identities

The division by $D(E, X, G)$ is required for recovery, or in other words, $D(E, X, G) \neq 0$ is required for the unique solution. Tassa [4] describes the probability in its Section 3.2. We look into the probability of $D(E, X, G) = 0$ for our method with experiments.

For recovery of 3 participants including 2 indispensable participants in a $(\{1, 3\}, n)$ hierarchical secret sharing scheme over GF(2^8), we observe 10,795 patterns of $D(E, X, G) = 0$ in the combination of 2,731,135(= $\binom{255}{3}$) patterns. This percentage is approximately $1/2^8$. Under the same observation in GF(2^{16}), the percentage is approximately $1/2^{16}$. Tassa shows that the probability of $\det(M_{\mathcal{V}}) = 0$ is less than $1/(q - 3)$ in a finite field of size q. Thus it is reasonable to think the same probability will be obtained for our proposed method and that there are almost no issues when a large finite field of characteristic 2 is used.

5. Software Implementation

For the $(\mathbf{k}, n)$ hierarchical secret sharing scheme with $\mathbf{k} = \{k_i\}_{i=0}^m$, we evaluate the method with one general purpose machine. We use a file size of 888,710 bytes for recovery and give some parameters of $\mathbf{k}$. **Table 2** shows the test environment and the GCC options related to performance. For operations with GF(2^L), the additive operation is replaced with the XOR operation, the multiplication operation uses the Russian peasant multiplication, the division operation uses $x^{-1} = x^{2^L-2}$, and the shift operation uses only the left shift by 1 bit. However, Shima et

Table 2 Test environment.

CPU	Intel ® Celeron ® Processor G1820 2.70 GHz × 2, 2 MB cache
RAM	3.6 GB
OS	CentOS 7 Linux 3.10.0-229.20.1.el7.x86_64
Programing language	The C language
Compiler system	gcc 4.8.3 (-O3 -flto -DNDEBUG)

Table 3 Experiment results.

Level **k**	Shares and speed for recovery
{1,3}	$p(7), p^{[1]}(14), p^{[1]}(17)$ 100.92 Mbps
	$p(2), p(3), p^{[1]}(8)$ 97.07 Mbps
{2,4}	$p(6), p(7), p^{[2]}(14), p^{[2]}(17)$ 63.19 Mbps
	$p(1), p(2), p(3), p^{[2]}(8)$ 59.79 Mbps
{2,3,5}	$p(6), p(7), p^{[2]}(14), p^{[3]}(24), p^{[3]}(27)$ 35.40 Mbps
	$p(1), p(2), p(3), p^{[2]}(8), p^{[3]}(27)$ 34.84 Mbps
{2,4,6,10}	$p(6), p(7), p^{[2]}(14), p^{[2]}(17), p^{[4]}(24),$ $p^{[4]}(27), p^{[6]}(34), p^{[6]}(35), p^{[6]}(37), p^{[6]}(39)$ 7.84 Mbps
	$p(1), p(2), p(3), p^{[2]}(8), p^{[2]}(9),$ $p^{[4]}(24), p^{[4]}(27), p^{[6]}(34), p^{[6]}(37), p^{[6]}(39)$ 7.57 Mbps
{3,7,11,14,17}	$p(5), p(6), p(7), p^{[3]}(14), p^{[3]}(15), p^{[3]}(17), p^{[3]}(19),$ $p^{[7]}(24), p^{[7]}(25), p^{[7]}(27), p^{[7]}(29), p^{[11]}(34),$ $p^{[11]}(37), p^{[11]}(39), p^{[14]}(44), p^{[14]}(47), p^{[14]}(49)$ 1.62 Mbps
	$p(1), p(2), p(3), p(5), p^{[3]}(8), p^{[3]}(9), p^{[3]}(14),$ $p^{[3]}(17), p^{[7]}(24), p^{[7]}(25), p^{[7]}(27), p^{[7]}(29),$ $p^{[11]}(34), p^{[11]}(37), p^{[11]}(39), p^{[14]}(44), p^{[14]}(47)$ 1.53 Mbps

Elements in GF(2^L) can be identified with polynomials $f(X) = \sum_{i=0}^{L-1} f_i X^i$, $f_i \in$ GF(2). Identities are represented by decimal numbers of $f_{L-1} \cdots f_1 f_0$ binary.

al. [6] have recognized that the computational cost of multiplication and division operations is high regardless of the value of L. In this experiment, we use only GF(2^8) and a lookup table that is precalculated for the multiplication and division operations over GF(2^8). Specifically, all the results of the multiplication operation are stored into an array of 2^{16} bytes and those of the division operation are stored into another array of 2^{16}. When each of the multiplication and division operations actually takes place, its operation refers to the lookup table of each array. Moreover, each determinant of $D(E, X, G)$ and $D(E, X, G_0)$ is calculated with its triangular matrix since the determinant of a triangular matrix equals the product of the diagonal entries. The primitive polynomial used for GF(2^8) is $x^8 + x^4 + x^3 + x^2 + 1$. **Table 3** shows the experiment results.

With a $(\{1, 3\}, n)$ hierarchical secret sharing scheme where the total number of participants is 60, we have got a 97 Mbps recovery speed. However, that shows 1/10th the speed of the optimization version [6] of 970 Mbps in a $(\{1, k\}, n)$ hierarchical secret sharing scheme with $k = 3$ over GF(2^8). The reason for the speed is mainly that the determinant calculation of a $k \times k$ matrix is generically implemented. Actually, we have got a 970 Mbps recovery speed with the optimization version shown later in Section 5.1.

Table 4 The number of recovery operations in 888,710 bytes.

	1 indispensable participant		2 indispensable participants	
	w/ table	w/o table	w/ table	w/o table
add	1,777,423	98,660,932	1,777,424	99,534,805
mul.	2,666,132	15,108,072	2,666,134	15,108,074
div.	888,710	888,710	888,710	888,710
shift	0	120,864,576	0	120,864,592
copy	2,666,133	26,661,305	2,666,134	26,661,308

Table 5 The number of the first 8-bit recovery operations.

	1 indispensable participant		2 indispensable participants	
	w/ table	w/o table	w/ table	w/o table
add	5	122	6	132
mul.	5	19	7	21
div.	1	1	1	1
shift	0	152	0	168
copy	6	35	7	38

Table 6 3 participants including 1 indispensable participant.

Participant	Share
$u_1 \in \mathcal{U}_0$	$p(u_1)$
$u_2 \in \mathcal{U}_1$	$p^{[1]}(u_2)$
$u_3 \in \mathcal{U}_1$	$p^{[1]}(u_3)$

In addition, **Table 4** shows the number of recovery operations in 888,710 bytes for the optimized $(\{1, 3\}, n)$ hierarchical secret sharing scheme over GF(2^8). When we look at the case in which the lookup table is not used (w/o table), we see that the multiplication operation used in the division operation and both the additive operation and the shift operation used in the multiplication operation are increased by the algorithm used in the implementation. Also, **Table 5** shows the number of the first $L = 8$-bit recovery operations, including the operations related to the identities that are processed only once.

Here we note that the optimization version and that of Ref. [6] are different implementations and that their speeds are the same. In Ref. [6], the dealer selects a random polynomial $p(x) = \sum_{i=1}^{k-1} a_i x^{2(i-1)+1} + s \in \mathrm{GF}(2^L)[x]$, where $a_i, s \in \mathrm{GF}(2^L)$, and gets $p'(x) = \sum_{i=1}^{k-1} a_i x^{2(i-1)} \in \mathrm{GF}(2^L)[x]$. The reason for the same speed is that the dominant computational cost of the optimization version and that of Ref. [6] are the same in a large file size of 888,710 bytes. Specifically, their operations related to the identities are different, but those are processed only once and are not dominant in a large file. In the optimization version of Ref.[6], the shares $p(u_1)$, $p'(u_2)$, $p'(u_3)$ instead of $p(u_1)$, $p^{[1]}(u_2)$, $p^{[1]}(u_3)$ shown in **Table 6** are given and let $c_1 = p(u_1)$, $c_2 = p'(u_2)$, $c_3 = p'(u_3)$. We obtain

$$D(E, X, G) = u_2^2 + u_3^2,$$

$$D(E, X, G_0) = \begin{vmatrix} c_1 & u_1 & u_1^3 \\ c_2 & 1 & u_2^2 \\ c_3 & 1 & u_3^2 \end{vmatrix}$$

$$= c_1(u_2^2 + u_3^2) + c_2 u_1(u_1^2 + u_3^2) + c_3 u_1(u_1^2 + u_2^2)$$

since $D(E, X, G)$ is calculated with $c_1 = 1$, $c_2 = 0$, $c_3 = 0$ of $D(E, X, G_0)$. Calculating the $D(E, X, G)$, $u_2^2 + u_3^2$, $u_1(u_1^2 + u_2^2)$, $u_1(u_1^2 + u_3^2)$ only once each is sufficient. The dominant computational cost is the calculation that requires c_1, c_2, c_3.

Table 7 3 participants including 2 indispensable participants.

Participant	Share
$u_1 \in \mathcal{U}_0$	$p(u_1)$
$u_2 \in \mathcal{U}_0$	$p(u_2)$
$u_3 \in \mathcal{U}_1$	$p^{[1]}(u_3)$

5.1 A ({1, 3}, n) Hierarchical Secret Sharing Scheme and Its Optimization

Taking practical use into consideration, we cover the optimization specialized for a $(\{1, 3\}, n)$ hierarchical secret sharing scheme and describe the recovery by 3 participants including 1 indispensable participant and ones including 2 indispensable participants.

5.1.1 3 Participants Including 1 Indispensable Participant

The shares shown in Table 6 are given and let $c_1 = p(u_1)$, $c_2 = p^{[1]}(u_2)$, $c_3 = p^{[1]}(u_3)$. Since $D(E, X, G)$ is calculated with $c_1 = 1$,$c_2 = 0$,$c_3 = 0$ of $D(E, X, G_0)$, we obtain

$$D(E, X, G) = u_2 + u_3,$$
$$D(E, X, G_0) = \begin{vmatrix} c_1 & u_1 & u_1^2 \\ c_2 & 1 & u_2 \\ c_3 & 1 & u_3 \end{vmatrix}$$
$$= c_1(u_2 + u_3) + c_2u_1(u_1 + u_3) + c_3u_1(u_1 + u_2).$$

When the size of the secret exceeds L bits in GF(2^L), the secret is divided into L-bit pieces and each share is generated from each of the pieces with each random polynomial, the random values can then be deleted after the shares are distributed. Generally, the size of the secret, such as a file size of 1 MB, exceeds L bits so the recovery procedures will be repeated in the combination of u_1, u_2, u_3. Therefore, it is sufficient that the additive operations and the multiplications of u_1, u_2, u_3 are performed only once for the file, and the dominant computational cost is the calculation that requires c_1, c_2, c_3. Specifically, it is sufficient that $D(E, X, G)$, $u_2 + u_3$, $u_1(u_1 + u_2)$, $u_1(u_1 + u_3)$ are calculated only once, respectively. Thus this recovery requires 3 multiplications in $c_1(u_2+u_3)$, $c_2u_1(u_1 + u_3)$, $c_3u_1(u_1 + u_2)$, 2 additive operations in $D(E, X, G_0)$, and 1 division operation in the calculation to derive the secret.

5.1.2 3 Participants Including 2 Indispensable Participants

The shares shown in **Table 7** are given and let $c_1 = p(u_1)$, $c_2 = p(u_2)$, $c_3 = p^{[1]}(u_3)$. Since $D(E, X, G)$ is calculated with $c_1 = 1$,$c_2 = 1$,$c_3 = 0$ of $D(E, X, G_0)$, we obtain

$$D(E, X, G) = u_2(u_2 + u_3) + u_1(u_1 + u_3)$$
$$= (u_1 + u_2)(u_1 + u_2 + u_3),$$
$$D(E, X, G_0) = \begin{vmatrix} c_1 & u_1 & u_1^2 \\ c_2 & u_2 & u_2^2 \\ c_3 & 1 & u_3 \end{vmatrix} = c_1u_2(u_2 + u_3)$$
$$+c_2u_1(u_1 + u_3) + c_3u_1u_2(u_1 + u_2).$$

Given the dominant computational cost in the same manner, this recovery requires 3 multiplications, 2 additive operations, and 1 division operation.

5.1.3 Identities to Always Recover the Secret

Since each participant has a different identity, for the case of 1 indispensable participant, $D(E, X, G)$ is always nonzero, and the secret is recovered by any combination of the participants in an authorized subset. For the case of 2 indispensable participants, $D(E, X, G) \neq 0$ if $u_1 + u_2 + u_3 \neq 0$. Therefore, every participant has an identity such that, for example, its least significant bit is always 1, the secret is recovered by any combination of the participants in an authorized subset, and our scheme can be operated over GF(2^8) up to 128 identities. Moreover, if every participant has some sequence of characters or some string corresponding to their identification, a 225-bit identity is generated from a SHA-224 hash value of the string and the least significant bit of 1. In doing so, our scheme can be operated over GF(2^{256}).

5.2 Comparison with Other Schemes

We compare other hierarchical secret sharing schemes with our optimization version over GF(2^8).

5.2.1 Tassa's Scheme

We consider the fastest implementation over GF(257), where the closest prime field to 2^8. Each of the multiplication and division operations refers to the lookup table in the same manner as our scheme. The computational cost is equivalently low, but the memory consumption is higher because each lookup table needs an array of 16-bit values stored from 0 to 256. Each of the additive and subtractive operations also refers to the lookup table. The computational cost is close to our scheme using only XOR operations, but the memory consumption is much higher because our scheme needs no lookup tables.

To achieve the *ideal* hierarchical secret sharing scheme, we must generate shares for s_i, $0 \leq i \leq n$ where the secret $s = \sum_{i=0}^{n} s_i * 257^i$. That requires complicated operations in distribution and recovery such that the multiple precision arithmetic is required and such that every bit size of shares equals the bit size of the secret. Our scheme requires simple 8-bit shares. Therefore, our scheme has a theoretical advantage in implementation, performance, and memory consumption.

We actually reach only 0.15 Mbps under the same test conditions because the computational cost of the multiple precision arithmetic is high, while the GNU multiple precision arithmetic library (GMP) is used under GMPbench 0.2 score: 2,813.9.

5.2.2 Selçuk et al.'s Scheme

When the truncated version is applied to our recovery procedure of 3 participants including 1 indispensable participant, shown in Chapter 5.1, we obtain

$$D(E, X, G) = u_2u_3(u_2 + u_3),$$
$$D(E, X, G_0) = \begin{vmatrix} c_1 & u_1 & u_1^2 \\ c_2 & u_2 & u_2^2 \\ c_3 & u_3 & u_3^2 \end{vmatrix} = c_1u_2u_3(u_2 + u_3)$$
$$+c_2u_1u_3(u_1 + u_3) + c_3u_1u_2(u_1 + u_2).$$

As for the computational cost, we will state that our proposed method has a very small advantage, while there is no difference between the two functions for the dominant computational cost. The closer to L bits in GF(2^L) the file size is, the more advantageous our scheme is, due to the smaller operations related to the identities. In addition, we consider a $(\{1, 2\}, n)$ hierarchical secret sharing scheme. All shares of non-indispensable participants are not the same for the truncated version since a multiplication with the participant identity takes place, while all shares of non-indispensable participants are the same for our function.

6. Concluding Remarks

We focus on a fast method and the ease of deleting the secret, and we propose a $(\mathbf{k}, n)$ hierarchical secret sharing scheme over finite fields of characteristic 2, applicable to any level of $\mathbf{k}$. In order to accomplish it, we introduce the n-th order reduction $f^{[n]}(x)$ and confirm it works in Birkhoff interpolation. Our scheme is also both *perfect* and *ideal*. Under the implementation used in the experiment applicable to any level, that is, not limited to the use of a specific level, our implementation system on a general purpose PC can recover the secret with $\mathbf{k} = \{1, 3\}$ in the processing of around 97 Mbps. Moreover, under the implementation optimized for a $(\{1, 3\}, n)$ hierarchical secret sharing scheme, our implementation system on the same PC can recover the secret in the processing of around 970 Mbps.

Acknowledgments The authors thank the anonymous referees for insightful comments on the manuscript.

References

[1] Blakley, G.R.: Safeguarding cryptographic keys, *AFIPS*, Vol.48, pp.313–317 (1979).

[2] Shamir, A.: How to share a secret, *Comm. ACM*, Vol.22, No.11, pp.612–613 (1979).

[3] Tassa, T.: Hierarchical Threshold Secret Sharing, *TCC 2004*, LNCS 2951, pp.473–490 (2004).

[4] Tassa, T.: Hierarchical Threshold Secret Sharing, *Journal of Cryptology*, Vol.20, No.2, pp.237–264 (2007).

[5] Shima, K. and Doi, H.: A study on fast hierarchical secret sharing schemes, *CSS 2015*, 3C4-5, pp.1327–1334 (2015). (in Japanese)

[6] Shima, K. and Doi, H.: A study on $(\{1, k\}, n)$ hierarchical secret sharing schemes over finite fields of characteristic 2, *IPSJ SIG Technical Reports, CSEC 72*, No.4 (2016). (in Japanese)

[7] Shima, K. and Doi, H.: A study on hierarchical secret sharing schemes applicable to any level over finite fields of characteristic 2, *CSS 2016*, 2C2-2, pp.425–432 (2016). (in Japanese)

[8] Shima, K. and Doi, H.: $(\{1, 3\}, n)$ hierarchical secret sharing scheme based on XOR operations for a small number of indispensable participants, *AsiaJCIS 2016*, pp.108–114 (2016).

[9] Fujii, Y., Tada, M., Hosaka, N., Tochikubo, K. and Kato, T.: A Fast $(2, n)$-Threshold Scheme and Its Application, *CSS 2005*, pp.631–636 (2005). (in Japanese)

[10] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: A Fast $(3, n)$-Threshold Secret Sharing Scheme Using Exclusive-OR Operations, *IEICE Trans. Fundamentals*, Vol.E91-A, No.1, pp.127–138 (2008).

[11] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: On a Fast (k, n)-Threshold Secret Sharing Scheme, *IEICE Trans. Fundamentals*, Vol.E91-A, No.9, pp.2365–2378 (2008).

[12] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: A New (k, n)-Threshold Secret Sharing Scheme and Its Extension, *ISC 2008*, LNCS 5222, pp.455–470 (2008).

[13] Yamamoto, H.: On Secret Sharing System Using (k, L, n) Threshold Scheme, *IEICE Transactions*, Vol.J68-A, No.9, pp.945–952 (1985). (Japanese Edition)

[14] Kurihara, J., Kiyomoto, S., Fukushima, K. and Tanaka, T.: A Fast (k, L, n)-Threshold Ramp Secret Sharing Scheme, *IEICE Trans. Fundamentals*, Vol.E92-A, No.8, pp.1808–1821 (2009).

[15] Kurihara, J. and Uyematsu, T.: A Novel Realization of Threshold Schemes over Binary Field Extensions, *IEICE Trans. Fundamentals*, Vol.E94-A, No.6, pp.1375–1380 (2011).

[16] Feng, G.-L., Deng, R.-H. and Bao, F.: Packet-loss resilient coding scheme with only XOR operations: *IEE Proc. Communications*, Vol.151, No.4 (2004).

[17] Blömer, J., Kalfane, M., Karp, R., Karpinski, M., Luby, M. and Zuckerman, D.: An XOR-Based Erasure-Resilient Coding Scheme, *ICSI Technical Report TR-95-048* (1995).

[18] Matsuo, M. and Mutou, K.: (k, n)-Threshold Secret Sharing Scheme Using Exclusive OR, *Panasonic Technical Journal*, Vol.59, No.2 (2013). (in Japanese)

[19] Suga, Y.: New constructions of (k, n)-threshold secret sharing schemes using exclusive-OR operations and their advantages, *CSS 2012*, pp.185–192 (2012). (in Japanese)

[20] Suga, Y.: New Constructions of $(2, n)$-Threshold Secret Sharing Schemes Using Exclusive-OR Operations, *2013 7th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing* (*IMIS*), pp.837–842 (2013).

[21] Ke, C., Anada, H., Kawamoto, J., Morozov, K. and Sakurai, K.: Cross-group Secret Sharing for Distributed Storages over Providers, *SCIS 2016*, 3A1-3 (2016). (in Japanese)

[22] Ozaki, H. and Sakurai, K.: Yet another security issue around Secret Sharing Schemes – Revisiting "unfair" cryptosystems, *SCIS 2016*, 3A1-5 (2016). (in Japanese)

[23] Beimel, A.: Secret-Sharing Schemes, A Survey, *IWCC 2011*, LNCS 6639, pp.11–46 (2011).

[24] Blundo, C., De Santis, A., Gargano, L. and Vaccaro, U.: On the information rate of secret sharing schemes, *TCS*, Vol.154, pp.283–306 (1996).

[25] Blundo, C., De Santis, A., Gargano, L. and Vaccaro, U.: On the Information Rate of Secret Sharing Schemes, *CRYPTO 1992*, LNCS, Vol.740, pp.149–169 (1993).

[26] Simmons, G.J.: How to (really) share a secret, *Advances in Cryptology - CRYPTO '88*, LNCS 403, pp.390–448 (1990).

[27] Brickell, E.F.: Some ideal secret sharing schemes, *Advances in Cryptology - EUROCRYPT '89*, LNCS 434, pp.468–475 (1990).

[28] Lorentz, G.G., Jetter, K. and Riemenschneider, S.D.: Birkhoff Interpolation, Encyclopedia of Mathematics and its Applications 19 (1983, 1984).

[29] Lorentz, G.G. and Zeller, K.L.: Birkhoff Interpolation, *SIAM Journal on Numerical Analysis*, Vol.8, No.1, pp.43–48 (1971).

[30] Selçuk, A.A., Kaşkaloğlu, K. and Özbudak, F.: On Hierarchical Threshold Secret Sharing, IACR Cryptology ePrint Archive 2009/450 (2009).

[31] Käsper, E., Nikov, V. and Nikova, S.: Strongly multiplicative hierarchical threshold secret sharing, *Information Theoretic Security*, LNCS 4883, pp.148–168 (2009).

Editor's Recommendation

This paper proposes an efficient hierarchical secret sharing scheme available for any access structure. Whereas the previous schemes for any access structure are only available over the finite fields of odd characteristic, the proposed scheme works over those of characteristic 2 taking advantage of Birkhoff interpolation. The security analysis and the evaluation results presented in the paper clearly show the reliability and the practicality of the proposed scheme. The paper will give insights to readers in this research field and thus is selected as a recommended paper.

(Masayuki Terada, CSS2016 Program Chair)

Koji Shima received his B.S. degree in Information Science from Tokyo University of Science in 1997 and M.S. degree in Information Security from Institute of Information Security in 2016, respectively. His research interests include information security and network protocols.

Hiroshi Doi received his B.S. degree in Mathematics from Okayama University in 1988, M.I.S. degree in Information Science from JAIST in 1994, and D.S. degree from Okayama University in 2000, respectively. He is currently a professor of Graduate School of Information Security, Institute of Information Security. His research interests include information security and cryptography.