New (k, l, m)-verifiable multi-secret sharing schemes based on XTR public key system

Jing Yang and Fang-Wei Fu

Abstract—Secret sharing was proposed primarily in 1979 to solve the problem of key distribution. In recent decades, researchers have proposed many improvement schemes. Among all these schemes, the verifiable multi-secret sharing (VMSS) schemes are studied sufficiently, which share multiple secrets simultaneously and perceive malicious dealer as well as participants. By pointing out that the schemes presented by Dehkordi and Mashhadi in 2008 cannot detect some vicious behaviors of the dealer, we propose two new VMSS schemes by adding validity check in the verification phase to overcome this drawback. Our new schemes are based on XTR public key system, and can realize $GF(p^6)$ security by computations in $GF(p^2)$ without explicit constructions of $GF(p^6)$, where p is a prime. Compared with the VMSS schemes using RSA and linear feedback shift register (LFSR) public key cryptosystems, our schemes can achieve the same security level with shorter parameters by using trace function. What's more, our schemes are much simpler to operate than those schemes based on Elliptic Curve Cryptography (ECC). In addition, our schemes are dynamic and threshold changeable, which means that it is efficient to implement our schemes according to the actual situation when participants, secrets or the threshold needs to be changed.

Index Terms—Verifiable multi-secret sharing, XTR public key system, trace function, shorter key parameters, fast key generation, dynamism, threshold changeable.

1 Introduction

It is well-known that secret sharing schemes [1], [2] are significant to protect secret keys, which are critical components in many applications of modern cryptography, such as threshold cryptography [3], commitment scheme [4], secure multiparty computation [5], [6], blockchain [7] and so on.

In 2004, an efficient VMSS scheme was proposed by Yang et al. [8]. Based on this scheme, in 2005, Shao and Cao [9] presented an improved scheme. However, this scheme still requires a private channel. In 2006, Zhao et al. (ZZZ) [10] proposed a new VMSS scheme, where participants choose their own shadows by themselves so that this scheme does not need a security channel. Then, in 2007, Dehkordi and Mashhadi [11] introduced RSA public key cryptosystem [20] into VMSS schemes for the first time to make their verifiable property more efficient. However, all these schemes [8], [9], [10], [11] still use Lagrange interpolation to distribute secrets, which are similar to Shamir's secret sharing scheme [1].

Further, in 2008, Dehkordi and Mashhadi presented two new types of efficient VMSS schemes (DM1 [12] and DM2 [13]), which employ the homogeneous and nonhomogeneous linear recursions [14] to increase the efficiency of the construction and reconstruction phase, respectively. In order to reduce the operating time, Hu et al. (HLC) [15] utilized LFSR sequence and LFSR public key cryptosystem [17], [18] to verify the validity of the data. Then, in 2015, Dehkordi and Mashhadi (DM3) [16] used LFSR public key cryptosystem and new nonhomogeneous linear recursions to make their schemes have shorter private and public key length.

Nevertheless, Liu et al. (LZZ) [19] found that ZZZ and DM1 schemes cannot detect some dealer's hostile behaviors and presented new schemes by RSA public key cryptosystem. Similarly, DM2 [13] and DM3 [16] have the same drawback as mentioned in [19]. We have proposed modified schemes (YF) [21] based on DM3 schemes by using LFSR public key cryptosystem. YF schemes can not only perceive the deception of both participants and the dealer, but also use one-third of the private and public key length of LZZ to achieve the same security level.

In this work, we propose two novel VMSS schemes to improve DM2 schemes by XTR public key system [22], which make full use of trace function to reduce the storage of data, computation cost and communication cost. In fact, XTR public key system can realize the security level in $GF(p^6)$ by computations in $GF(p^2)$ where p is a prime. Compared with RSA, LFSR public key cryptosystems and ECC, XTR public key system needs shorter key length than RSA and LFSR public cryptosystems to achieve the same security level, and has simpler procedure of parameter and key generation than ECC. Further, XTR public key system can be considered as a special case of the optimization of LFSR public key cryptosystem . Therefore, our proposed VMSS schemes have many good properties which will be discussed later.

The rest of this paper is organized as follows. In Section 2, we review the nonhomogeneous linear recursion, XTR

- *Jing Yang is with the Chern Institute of Mathematics and LPMC, and Tianjin Key Laboratory of Network and Data Security Technology, Nankai University, Tianjin, 300071, China.*
  E-mail: yangjing0804@mail.nankai.edu.cn
- *Fang-Wei Fu is with the Chern Institute of Mathematics and LPMC, and Tianjin Key Laboratory of Network and Data Security Technology, Nankai University, Tianjin, 300071, China.*
  E-mail: fwfu@nankai.edu.cn

Manuscript received April 19, 2015; revised August 16, 2015. (Corresponding author: Jing Yang.)

public key system, and give the review and attack on DM2 schemes. In Sections 3 and 4, we propose our two new VMSS schemes respectively. We present the security analysis in Section 5, and in Section 6 we give the performance analysis. Finally, we conclude our schemes in Section 7.

2 Preliminaries

2.1 Nonhomogeneous linear recursion

In this subsection, we firstly introduce the linear recurring sequence [14].

Definition 1. Let k be a positive integer, and $c, a_1, a_2, \cdots, a_k$ be given elements of a finite field $GF(q)$ where q is a prime. If $\{u_i\}_{i\geq 0}$ satisfies the relation

$$u_{i+k} = a_k u_{i+k-1} + \cdots + a_1 u_i + c \quad (i = 0, 1, \cdots) \quad (*),$$

then $\{u_i\}_{i\geq 0}$ is called a kth-order linear recurring sequence in $GF(q)$.

Remark 1. Note that the terms $u_0, u_1, \cdots, u_{k-1}$, which can determine the rest of the sequence uniquely, are referred to as the initial values of the sequence. A relation of the form $(*)$ is called a kth-order linear recurrence relation. If c=0, we call $(*)$ a homogeneous linear recursion. Otherwise, $(*)$ is a nonhomogeneous linear recursion (NLR).

For a kth-order linear recurring sequence $\{u_i\}_{i\geq 0}$, $x^k + a_1 x^{k-1} + \cdots + a_k = 0$ is called its auxiliary equation, and $U(x) = \Sigma_{i=0}^{\infty} u_i x^i$ is called its generating function.

Lemma 1. Let $GF(q)$ be a finite field, where q is a prime. Suppose that $(x-\alpha_1)^{m_1}(x-\alpha_2)^{m_2}\cdots(x-\alpha_l)^{m_l} = 0$ is the auxiliary equation of a kth-order linear recurring sequence $\{u_i\}_{i\geq 0}$ in $GF(q)$, where $m_1 + m_2 + \cdots + m_l = k$. Then the generating function of $\{u_i\}_{i\geq 0}$ is

$$U(x) = \frac{R(x)}{(1-\alpha_1 x)^{m_1}(1-\alpha_2 x)^{m_2}\cdots(1-\alpha_l x)^{m_l}},$$

where $R(x)$ is a polynomial of x with $\deg(R(x)) < k$ in $GF(q)[X]$.

Further, $u_i = p_1(i)\alpha_1^i + p_2(i)\alpha_2^i + \cdots + p_l(i)\alpha_l^i$, where $p_j(i) = A_0 + A_1 i + A_2 i^2 + \cdots + A_{m_j-1} i^{m_j-1}$, $j = 1, 2, \cdots, l$. Notice that $A_0, A_1, \cdots, A_{m_j-1}$ are undetermined constants in $GF(q)$ which can be calculated from $a_1, a_2, \cdots, a_k$.

Corollary 1. Let $GF(q)$ be a finite field, where q is a prime. Consider a typical fraction $\frac{R(x)}{(1-\alpha x)^m}$, where $\alpha \in GF(q)$, and $R(x)$ is a polynomial of x with $\deg(R(x)) < m$ in $GF(q)[X]$. Then,

$$\frac{R(x)}{(1-\alpha x)^m} = \sum_{i=0}^{\infty} u_i x^i$$

and $u_i = p(i)\alpha^i$, where $p(i) = A_0 + A_1 i + \cdots + A_{m-1} i^{m-1}$ and $A_0, A_1, \cdots, A_{m-1}$ are in $GF(q)$.

Through Corollary 1, the following two main theorems are given, which have been proved in [13].

Theorem 1. Let $GF(q)$ be a finite field, where q is a prime. Assume that the sequence $(u_i)_{i\geq 0}$ is defined by the following NLR equations:

$$[NLR1] = \begin{cases} u_0 = c_0, u_1 = c_1, \cdots, u_{k-1} = c_{k-1}, \\ \displaystyle\sum_{j=0}^{k} \left(\begin{array}{c} k \\ j \end{array}\right) u_{i+k-j} = c(-1)^i i \quad (i \geq 0), \end{cases} \quad (1)$$

where $c, c_0, c_1, \cdots, c_{k-1}$ are constants in $GF(q)$. Therefore, $u_i = p(i)(-1)^i$, where $p(i) = A_0 + A_1 i + \cdots + A_{k+1} i^{k+1}$ and $A_0, A_1, \cdots, A_{k+1}$ are in $GF(q)$.

Theorem 2. Let $GF(q)$ be a finite field, where q is a prime. Assume that the sequence $(u_i)_{i\geq 0}$ is defined by the following NLR equations:

$$[NLR2] = \begin{cases} u_0 = c_0, u_1 = c_1, \cdots, u_{k-1} = c_{k-1}, \\ \displaystyle\sum_{j=0}^{k} \left(\begin{array}{c} k \\ j \end{array}\right) (-1)^j u_{i+k-j} = ci \quad (i \geq 0), \end{cases} \quad (2)$$

where $c, c_0, c_1, \cdots, c_{k-1}$ are constants in $GF(q)$. Therefore, $u_i = p(i)$, where $p(i) = A_0 + A_1 i + \cdots + A_{k+1} i^{k+1}$, and $A_0, A_1, \cdots, A_{k+1}$ are in $GF(q)$.

2.2 The XTR public key system

In 2000, Lenstra and Verheul proposed the XTR public key system [22], utilizing the third-order LFSR sequence. Actually, XTR public key system belongs to LFSR sequence public key system [17], [18], [22].

Firstly, in 1999, Gong and Harn proposed LFSR public key cryptosystem [17], [18], i.e., GH public key system, which is based on a third-order LFSR sequence generated by an irreducible polynomial $f(x) = x^3 - ax^2 + bx - 1$, where $a, b \in GF(p)$ and p is a prime.

Compared with LFSR public key cryptosystem, XTR public key system requires $b = a^p$ where $a \in GF(p^2)$, and a special group with order q in $GF(p^6)^*$ where $q|(p^2 - p + 1)$ and $q > 3$. In other words, the irreducible polynomial used in XTR public key system is $f(x) = x^3 - ax^2 + a^p x - 1$, where $a \in GF(p^2)$. Actually, XTR is the first method which utilizes computations on $GF(p^2)$ to achieve $GF(p^6)$ security without requiring explicit construction of $GF(p^6)$.

Then we review some basic knowledge about the XTR public key system, which can be found in [22], [23].

Definition 2. Let $p > 3$ and $q > 3$ be two primes, satisfying $p \equiv 2 \pmod 3$, and $q|(p^2 - p + 1)$. Let g be an element with order q in $GF(p^6)^*$, where $GF(p^6)^*$ is the multiplicative group of the finite field $GF(p^6)$. Then, we refer to the subgroup $< g >$ as the XTR group.

Definition 3. The conjugates of $g \in GF(p^6)^*$ over $GF(p^2)$ are g, g^{p^2} and g^{p^4}. Then the trace function $Tr(g)$ of $g \in GF(p^6)^*$ over $GF(p^2)$ is the sum of the conjugates of g over $GF(p^2)$, which means that

$$Tr(g) = g + g^{p^2} + g^{p^4}.$$

Proposition 1. We have $Tr(g)^{p^2} = Tr(g)$, so that $Tr(g) \in GF(p^2)$.

Proposition 2. The conjugates of g with order q satisfying $q|(p^2 - p + 1)$ are g, g^{p-1} and g^{-p}, then we have $Tr(g) = g + g^{p-1} + g^{-p} \in GF(p^2)$.

Lemma 2. The roots of $X^3 - Tr(g)X^2 + Tr(g)^p X - 1$ are the conjugates of g, where p is a prime.

By trace function, we can not only represent the elements of the XTR group by elements in $GF(p^2)$, but also compute the powers of g efficiently by performing the computations on $GF(p^2)$ while avoiding operations in $GF(p^6)$.

Definition 4. Let $c = Tr(g) \in GF(p^2)$, we define

$$F(c, X) = X^3 - cX^2 + c^p X - 1,$$

which is a polynomial in $GF(p^2)[X]$. Let $g_0, g_1, g_2 \in GF(p^6)$ be three roots of $F(c, X)$, and for an integer $n \in Z$, we define $c_n = g_0^n + g_1^n + g_2^n$, which means that $c_n = Tr(g^n)$. Obviously, $c = c_1$.

Theorem 3. $F(c, X) \in GF(p^2)[X]$ is irreducible if and only if its roots have order q, where $q|(p^2 - p + 1)$ and $q > 3$.

Next, we introduce the definition of the XTR-discrete logarithm (XTR-DL) problem.

Definition 5. Given $c = Tr(g)$, $c_n \in Tr(< g >)$, the XTR-DL problem is to find $0 \leq n < q$ such that $c_n = Tr(g^n)$.

The following theorem has been proved in [22], [23].

Theorem 4. The XTR-DL problem is equivalent to the discrete logarithm problem in $< g >$.

The security of XTR public key system is based on constructing a one-way trapdoor function through XTR-DL problem. In order to achieve this goal, when we know the values of $Tr(g)$ and n, we need to compute the values of $Tr(g^n)$ efficiently, which has been solved by Lenstra and Verheul in [22].

Finally, we give the definition of XTR public key system.

Definition 6. Let $p > 3$ and $q > 3$ be two primes such that $p \equiv 2 \pmod 3$ and $q|(p^2 - p + 1)$. Let g be an element in $GF(p^6)^*$ with order q, and $\{p, q, g, Tr(g)\}$ be public parameters. All the computations here are implemented in $GF(p^2)$:

(1) Public key: $Tr(g^k)$, where $1 < k < q$.

(2) Secret key: k, where $1 < k < q$.

(3) Encryption: Given the plaintext $M \in GF(p^2)$ and a secret random integer b $(1 < b < q - 2)$, the ciphertext is $c = e_k(M, b) = (Tr(g^b), E)$, where $E = Tr(g^{bk}) * M$, and $Tr(g^{bk})$ can be computed by Algorithm 2.3.7 [22] using b and $Tr(g^k)$.

(4) Decryption: Given the ciphertext $c = (Tr(g^b), E)$, and the secret key k, and $Tr(g^{bk})$ can be computed by Algorithm 2.3.7 [22] using k and $Tr(g^b)$. Then, the plaintext is $M = E * Tr(g^{bk})^{-1}$.

2.3 Review and attack on DM2 schemes

In this subsection, we review DM2 schemes [13] simply which are based on ECC, and then provide a kind of attack on DM2 schemes. Because the two schemes in [13] are similar, we take the type 1 scheme as an example.

2.3.1 Review of DM2 schemes

Initialization phase

Let $S_1, S_2, \cdots, S_l$ be l shared secrets among m participants $P_1, P_2, \cdots, P_m$. Let q be a prime number such that $q > \left(\begin{array}{c} k \\ j \end{array} \right)$ for $j = 1, 2, \cdots, k$, where k is the threshold of this scheme.

Firstly, the dealer D chooses two large primes p_1 and p_2, and calculates $N = p_1 p_2$. Let v be an integer such that $gcd(27v^2, N) = 1$. The elliptic curve $E_N(0, v)$ over the ring $\mathbb{Z}_N$ is the set of points $(x, y) \in \mathbb{Z}_N \times \mathbb{Z}_N$ satisfying the equation $y^2 \equiv x^3 + v \pmod N$ together with the point at infinity $\mathcal{O}_N$.

Then, D considers $Q \in E_N(0, v)$ such that the discrete logarithm problem is infeasible in cyclic group $< Q >$. Finally, D publishes $\{N, Q\}$.

Every participant P_i chooses an integer s_i randomly as secret shadow and calculates $R_i = s_i Q$. Then P_i sends (R_i, i) to the dealer D. D must ensure that for all $i \neq j$, $R_i \neq R_j$. Finally, D releases $(R_1, R_2, \cdots, R_m)$.

Construction phase

The following steps need to be performed by D:

(1) D chooses a random integer e such that $gcd(e, n_N) = 1$ and calculates d such that $ed \equiv de \equiv 1 \pmod{n_N}$, where $n_N = lcm(\#E_{p_1}(0, v), \#E_{p_2}(0, v))$, and $\#E_p(0, v)$ denotes the order (i.e., the number of points) of the elliptic curve $E_p(0, v)$.

(2) For $i = 1, 2, \cdots, m$, D calculates $R_0 = dQ$ and $B_i = dR_i$ over $E_N(0, v)$.

(3) For $i = 1, 2, \cdots, m$, D calculates $I_i = x_{B_i} + y_{B_i}$, where x_{B_i} and y_{B_i} are the x-coordinate and the y-coordinate of the point B_i over $E_N(0, v)$ respectively.

(4) D chooses an integer c $(c < q)$ and considers a NLR defined by the following equations:

$$\begin{cases} u_0 = I_1, u_1 = I_2, \cdots, u_{k-1} = I_k, \\ \sum_{j=0}^{k} \left(\begin{array}{c} k \\ j \end{array} \right) (-1)^j u_{i+k-j} = ci \pmod q \quad (i \geq 0). \end{cases} \tag{3}$$

(5) For $k \leq i \leq m + l + 3$, D calculates u_i.

(6) D calculates $y_i = I_i - u_{i-1}$ for $k < i \leq m$ and $r_i = S_i - u_{m+i}$ for $1 \leq i \leq l$.

(7) D releases $(R_0, e, r_1, r_2, \cdots, r_l, y_{k+1}, y_{k+2}, \cdots, y_m, u_{m+l+2}, u_{m+l+3})$.

Verification phase

Every participants P_i can compute $s_i R_0$ to obtain his or her share B_i as follows:

$$s_i R_0 = s_i dQ = ds_i Q = dR_i = B_i.$$

Assume that at least k participants $\{P_i\}_{i=1}^k$ use their shares $\{B_i\}_{i=1}^k$ to recover the secrets $S_1, S_2, \cdots, S_l$. A participant P_i can check the validity of the secret shares provided by the other authorized participants by the steps as follows:

$eB_j = R_j$ over $E_N(0, v)$ for $j = 1, 2, \cdots, k$ and $j \neq i$.

Recovery phase

Assume that any k participants $\{P_i\}_{i \in I}$ use their shares $\{B_i\}_{i \in I}$ to recover the secrets:

(1) Calculate $I_i = x_{B_i} + y_{B_i}$ for $i \in I$.

(2) Calculate k terms $\{u_{i-1}\}_{i \in I}$ in the equations (3) using the formulas as follows:

$$u_{i-1} = \begin{cases} I_i & if\, 1 \leq i \leq k, \\ I_i - y_i & if\, k < i \leq m. \end{cases}$$

(3) Utilize $k+2$ pairs $(i-1, u_{i-1})_{i \in I}$, $(m+l+2, u_{m+l+2})$, and $(m + l + 3, u_{m+l+3})$ to construct the polynomial $p(x)$ with degree $k + 1$:

$$\begin{aligned} p(x) &= \sum_{i \in I'} Y_i \prod_{j \in I', j \neq i} \frac{x - X_j}{X_i - X_j} \pmod q, \\ &= A_0 + A_1 x + \cdots + A_{k+1} x^{k+1} \pmod q. \end{aligned}$$

Notice we use (X_i, Y_i) for $i \in I'$ where $I' = I \cup \{m + l + 2, m + l + 3\}$ to denote these $k + 2$ pairs, respectively.

(4) Calculate $u_j = p(j)$ for $j = m + 1, m + 2, \cdots, m + l$.

(5) Recover $S_j = u_{m+j} + r_j$ for $j = 1, 2, \cdots, l$.

2.3.2 *Attack on DM2 schemes*

Notice that when authorized participants recover the secrets, these participants only check the validity of B_i by whether eB_i equals to R_i, while the consistence between B_i and $\{u_i\}$ is not verified. Thus when the sequence $\{u_i\}$ or $\{y_i\}$ is generated in the construction phase, a malicious D can substitute the true $B_i = dR_i$ with a fake $B'_i = dR'_i$ $(R'_i \neq R_i)$ over $E_N(0, v)$, which means that:

(1) D chooses a random integer e such that $gcd(e, n_N) = 1$ and calculates d such that $ed \equiv de \equiv 1 \pmod{n_N}$.

(2) For $i = 1, 2, \cdots, m$, D calculates $R_0 = dQ$ and $B_i = dR_i$ over $E_N(0, v)$.

When $1 \leq i \leq k$,

(3) D replaces B_i with B'_i over $E_N(0, v)$ to calculate a new $I'_i = x_{B'_i} + y_{B'_i}$, where $x_{B'_i}$ and $y_{B'_i}$ are the x-coordinate and the y-coordinate of the point B'_i respectively.

(4) D selects an integer c $(c < q)$ and considers the following formulas:

$$\begin{cases} u_0 = I_1, u_1 = I_2, \cdots, u_{i-1} = I'_i, \cdots, u_{k-1} = I_k, \\ \sum\limits_{j=0}^{k} \begin{pmatrix} k \\ j \end{pmatrix} (-1)^j u_{i+k-j} = ci \pmod{q} \quad (i \geq 0). \end{cases}$$

Then D calculates u_i for $k \leq i \leq m + l + 3$.

(5) D calculates $y_i = I_i - u_{i-1}$ for $k < i \leq m$, and $r_i = S_i - u_{m+i}$ for $1 \leq i \leq l$.

(6) D releases $(R_0, e, r_1, \cdots, r_l, y_{k+1}, y_{k+2}, \cdots, y_m, u_{m+l+2}, u_{m+l+3})$.

When $k < i \leq m$,

(3′) For $i = 1, 2, \cdots, m$, D calculates $I_i = x_{B_i} + y_{B_i}$, where x_{B_i} and y_{B_i} are the x-coordinate and the y-coordinate of the point B_i over $E_N(0, v)$ respectively.

(4′) D selects an integer c $(c < q)$ and considers the following formulas:

$$\begin{cases} u_0 = I_1, u_1 = I_2, \cdots, u_{k-1} = I_k, \\ \sum\limits_{j=0}^{k} \begin{pmatrix} k \\ j \end{pmatrix} (-1)^j u_{i+k-j} = ci \pmod{q} \quad (i \geq 0). \end{cases}$$

Then D calculates u_i for $k \leq i \leq m + l + 3$.

(5′) D replaces the I_i with I'_i to calculate $y'_i = I'_i - u_{i-1}$, where $I'_i \neq I_i$, then calculates other $y_j = I_j - u_{j-1}$ $(k < j \leq m, j \neq i)$ and $r_i = S_i - u_{m+i}$ $(1 \leq i \leq l)$ correctly.

(6′) D releases $(R_0, e, r_1, r_2, \cdots, r_l, y_{k+1}, y_{k+2}, \cdots, y'_i, \cdots, y_m, u_{m+l+2}, u_{m+l+3})$.

In the recovery phase, since P_i can not discover the replacement, P_i still offers the true B_i that conflicts with the sequence $\{u_i\}$ or $\{y_i\}$ produced by the dealer as above. So the recovered secrets are not valid. Nonetheless, at least k participants without P_i can reconstruct secrets successfully. Actually, it is difficult to verify which I_i is substituted. So DM2 schemes [13] cannot prevent this kind of malicious behavior of the dealer. In addition, if more than one I_i is replaced by the dealer with some invalid I'_i, the circumstance is even more complex.

3 Scheme 1

In this section, in order to get rid of the drawback as mentioned in Section 2.3.2, we propose a novel VMSS scheme by using $[NLR1]$, XTR public key system, discrete logarithm problem and XTR-DL problem.

3.1 Initialization phase

D represents the dealer. Let $P = \{P_1, P_2, \cdots, P_m\}$ be the collection of participants, and k $(k \leq m)$ be the threshold.

At first, the dealer D performs the following operations:

(1) D randomly chooses two primes p, q $(p > 3, q > 3)$ with λ bits satisfying $p \equiv 2 \pmod{3}$, $q | (p^2 - p + 1)$ and $q > \begin{pmatrix} k \\ j \end{pmatrix}$ for $j = 0, 1, \cdots, k$.

(2) D selects an element g of $GF(p^6)^*$ with order q satisfying that XTR-DL problem with the base g is infeasible. Then D computes $Tr(g)$.

(3) D chooses $b \in Z$ $(1 < b < q - 2)$ randomly, then computes $Tr(g^b)$.

(4) D releases $(\lambda, p, q, g, Tr(g), Tr(g^b))$.

Then, the authorized participants perform the following operations:

(1) Each P_i with ID_i chooses $x_i \in Z$ $(1 < x_i < q)$ for $i = 1, 2, \cdots, m$.

(2) P_i computes $y_i = Tr(g^{x_i})$ as his shadow for $i = 1, 2, \cdots, m$.

(3) P_i provides (ID_i, y_i) to D, and keeps x_i secret, where $i = 1, 2, \cdots, m$.

D must ensure that $y_i \neq y_j$ $(i \neq j)$, otherwise P_i needs to select a new x_i to recalculate y_i. Then D releases (ID_i, y_i) for $i = 1, 2, \cdots, m$.

Remark 2: After the initialization phase, all the public parameters can be reused. Since D does not get useful information from participants' shadows, these shadows can also be reused.

3.2 Construction phase

Let $S_1, S_2, \cdots, S_l \in GF(q)^*$ be l secrets. Then D generates a subshadow u_i for each participant P_i as follows:

(1) Randomly chooses $c_i \in GF(q)^*$ for $i = 0, 1, 2, \cdots, k - 1$.

(2) Chooses a random constant $c \in GF(q)^*$, considers $[NLR1]$ presented by the following equations and computes u_i for $k \leq i \leq m + l + 1$:

$$[NLR1] = \begin{cases} u_0 = c_0, u_1 = c_1, \cdots, u_{k-1} = c_{k-1}, \\ \sum\limits_{j=0}^{k} \begin{pmatrix} k \\ j \end{pmatrix} u_{i+k-j} = c(-1)^i i \pmod{q} (i \geq 0). \end{cases}$$

(3) Computes $z_i = S_i - u_{m+i-1} \pmod{q}$ for $i = 1, 2, \cdots, l$.

(4) Computes $Tr(g^{bx_i})$ by using $Tr(g^{x_i})$ and b, then $E_i = Tr(g^{bx_i}) * u_{i-1} \pmod{q}$ for $1 \leq i \leq m$.

(5) Computes $T_i = g^{u_{i-1}} \pmod{p^2}$ for $1 \leq i \leq m$.

(6) Releases $(E_1, E_2, \cdots, E_m, T_1, T_2, \cdots, T_m, z_1, z_2, \cdots, z_l, c, u_{m+l}, u_{m+l+1})$.

3.3 Verification phase

Each P_i can get its subshadow u_{i-1} by the following way. At first, P_i can compute $Tr(g^{bx_i})$ by using x_i and $Tr(g^b)$ for $1 \leq i \leq m$. Then P_i will get u_{i-1} by

$$u_{i-1} = E_i * Tr(g^{bx_i})^{-1} \pmod{q}, \quad 1 \leq i \leq m.$$

The validity and consistence of P_i's subshadow u_{i-1} with public messages can be checked as follows:

$$\prod_{j=0}^{k}(T_{i+1+k-j})^{\left(\begin{array}{c}k\\j\end{array}\right)} \overset{?}{=} g^{c(-1)^i i} \pmod{p^2},$$

$$T_i \overset{?}{=} g^{u_{i-1}} \pmod{p^2}.$$

If the verification succeeds, P_i thinks that its subshadow u_{i-1} is true and is consistent with public messages. If every verification succeeds, participants think that D is honest.

3.4 Recovery phase

Suppose that at least k participants $\{P_i\}_{i\in I}$ $(I \subseteq \{1,2,\cdots,m\})$ use these subshadows $\{u_{i-1}\}_{i\in I}$ to recover the shared secrets. Every P_i can check the validity of $\{u_{j-1}|j\in I, j\neq i\}$ as follows:

$$g^{u_{j-1}} \overset{?}{=} T_j \pmod{p^2}, \quad j\in I \; and \; j\neq i.$$

There are two ways to recover the secrets. From these two ways, we can see that Scheme 1 is a (k,l,m)-threshold secret sharing schemes.

Way 1: Owning k true subshadows $\{u_{i-1}|i \in J \subseteq I, |J| = k\}$ and the published $\{u_{m+l}, u_{m+l+1}\}$, they can use Theorem 1 to get the following equations, where $i \in J' = J \cup \{m+l+1, m+l+2\}$:

$$z_0+z_1(i-1)+\cdots+z_{k+1}(i-1)^{k+1} = u_{i-1}(-1)^{i-1} \pmod{q}.$$

Solving these $k+2$ equations or using Lagrange interpolation formulas, they have $z_0 = A_0, z_1 = A_1, \cdots, z_{k+1} = A_{k+1}$ in $GF(q)$.

Next, they get

$$u_{i-1} = (A_0+A_1(i-1)+\cdots+A_{k+1}(i-1)^{k+1})(-1)^{i-1} \pmod{q}$$

where $i \in \{1,2,\cdots,m+l+2\}\backslash J'$.

Finally, they recover the secrets: $S_i = z_i + u_{m+i-1}$ $\pmod{q}$, $i = 1,2,\cdots,l$.

Way 2: If owning k successive $\{u_{i-1}, u_i, \cdots, u_{i+k-2}\}$, these participants can get u_j $(j = i+k-1, i+k, \cdots, m+l-1)$ by the following equations:

$$\sum_{j=0}^{k}\left(\begin{array}{c}k\\j\end{array}\right) u_{n+k-j} = c(-1)^n n \pmod{q} \quad (n\geq 0).$$

Finally, they can recover the secrets: $S_i = z_i + u_{m+i-1}$ $\pmod{q}$, $i = 1,2,\cdots,l$.

4 Scheme 2

In this section, in order to get rid of the drawback as mentioned in Section 2.3.2, we propose a novel VMSS scheme by using $[NLR2]$, XTR public key system, discrete logarithm problem and XTR-DL problem.

4.1 Initialization phase

The initialization phase in Scheme 2 is the same as Scheme 1.

4.2 Construction phase

In this phase, we replace $[NLR1]$ with $[NLR2]$, and the rest is identical to Scheme 1.

$$[NLR2] = \begin{cases} u_0 = c_0, u_1 = c_1, \cdots, u_{k-1} = c_{k-1}, \\ \displaystyle\sum_{j=0}^{k}\left(\begin{array}{c}k\\j\end{array}\right)(-1)^j u_{i+k-j} = ci \pmod{q} (i\geq 0). \end{cases}$$

4.3 Verification phase

Each P_i can get its subshadow u_{i-1} by computing $u_{i-1} = E_i * Tr(g^{bx_i})^{-1} \pmod{q}$ for $1 \leq i \leq m$. The validity and consistence of P_i's subshadow u_{i-1} with public messages can be checked as follows:

$$\prod_{j=0}^{k}(T_{i+1+k-j})^{(-1)^j\left(\begin{array}{c}k\\j\end{array}\right)} \overset{?}{=} g^{ci} \pmod{p^2},$$

$$T_i \overset{?}{=} g^{u_{i-1}} \pmod{p^2}.$$

If the verification succeeds, P_i thinks its subshadow u_{i-1} is true and is consistent with public messages. If every verification succeeds, participants think that D is honest.

4.4 Recovery phase

Assume that at least k participants $\{P_i\}_{i\in I}$ $(I \subseteq \{1,2,\cdots,m\})$ use these subshadows $\{u_{i-1}\}_{i\in I}$ to recover the shared secrets. Each P_i can check the validity of $\{u_{j-1}|j\in I, j\neq i\}$ as follows:

$$g^{u_{j-1}} \overset{?}{=} T_j \pmod{p^2}, \quad j\in I \; and \; j\neq i.$$

There are two ways to recover the secrets. From these two ways, we can see that Scheme 2 is also a (k,l,m)-threshold secret sharing schemes.

Way 1: Owning k true subshadows $\{u_{i-1}|i \in J \subseteq I, |J| = k\}$ and the published $\{u_{m+l}, u_{m+l+1}\}$, they can use Theorem 2 to get the following equations, where $i \in J' = J \cup \{m+l+1, m+l+2\}$:

$$z_0 + z_1(i-1) + \cdots + z_{k+1}(i-1)^{k+1} = u_{i-1} \pmod{q}.$$

Solving these $k+2$ equations or using Lagrange interpolation formulas, they get $z_0 = A_0, z_1 = A_1, \cdots, z_{k+1} = A_{k+1}$ in $GF(q)$.

Next, they get

$$u_{i-1} = A_0 + A_1(i-1) + \cdots + A_{k+1}(i-1)^{k+1} \pmod{q}$$

where $i \in \{1,2,\cdots,m+l+2\}\backslash J'$.

Finally, they recover the secrets: $S_i = z_i + u_{m+i-1}$ $\pmod{q}$, $i = 1,2,\cdots,l$.

Way 2: If owning k successive $\{u_{i-1}, u_i, \cdots, u_{i+k-2}\}$, these participants can get u_j $(j = i+k-1, i+k, \cdots, m+l-1)$ by the following equations:

$$\sum_{j=0}^{k}\left(\begin{array}{c}k\\j\end{array}\right)(-1)^j u_{n+k-j} = cn \pmod{q} \quad (n\geq 0).$$

Finally, they can recover these secrets: $S_i = z_i + u_{m+i-1}$ $\pmod{q}$, $i = 1,2,\cdots,l$.

5 Security analysis

The security of presented schemes is based on the nonhomogeneous linear recursion, XTR public key system, discrete logarithm problem and XTR-DL problem. Then we analyze our schemes from three aspects.

5.1 Correctness

In this subsection, we discuss the correctness of our schemes.

Theorem 5. If the dealer and correlated participants behave honestly, any k participants can reconstruct the shared secrets.

Proof. We can utilize two ways mentioned in Sections 3.4 and 4.4 to recover those secrets.

The correctness of Way 1 is based on solving $k+2$ simultaneous equations or using Lagrange interpolation polynomials with random k subshadows $\{u_{i-1}|i \in J, |J| = k\}$ and the published $\{u_{m+l}, u_{m+l+1}\}$. In Theorem 1 and Theorem 2, there are $k+2$ uncertain coefficients in $p(x)$. Therefore, $p(x)$ can be uniquely defined, which means that any authorized k participants can reconstruct the secrets.

The correctness of Way 2 is based on a nonhomogeneous linear recursion with degree k. Note that we require that the indices of these subshadows are successive. Since $[NLR1]$ and $[NLR2]$ are both nonhomogeneous linear recursions with degree k, these participants have to compute k terms u_j $(j = i-1, i, \cdots, i+k-2)$ to obtain other $u_{j'}$ $(j' = i+k-1, i+k, \cdots, m+l-1)$, which means that they can recover the shared secrets.

Remark 3. Next, we will discuss the reason why we publish $\{u_{m+l}, u_{m+l+1}\}$ instead of other subshadows.

At first, $\{u_0, u_1, \cdots, u_{m-1}\}$ are subshadows of participants $\{P_1, P_2, \cdots, P_m\}$ respectively. Besides, $\{u_m, u_{m+1}, \cdots, u_{m+l-1}\}$ are correlated to the shared secrets $\{S_1, S_2, \cdots, S_l\}$. Then only u_{m+l} and u_{m+l+1} not only can satisfy the requirement, but also will not disclose any information about subshadows and secrets.

5.2 Verifiability

Theorem 6. In the construction phase, it is impossible for the dealer to cheat participants.

Proof. From the Figure 1, in DM2 schemes [13], we know that each P_i chooses its secret shadow s_i, calculates $R_i = s_iQ$, and sends R_i to D. After that, D selects its secret shadow d to compute $R_0 = dQ$, then transforms R_0 to P_i. Hence, both D and P_i can calculate the secret share by $B_i = s_iR_0 = dR_i$. Nevertheless, whether B_i used in the generation of $\{u_{i-1}\}$ is identical to that offered by P_i is not checked.

In contrast, in our schemes, P_i chooses x_i and keeps it secret from D. P_i sends $y_i = Tr(g^{x_i})$ to the dealer, where y_i is the public key of x_i. Then D gets $E_i = Tr(g^{bx_i}) * u_{i-1}$ by choosing a random number b from $(1, q-2)$. After that, P_i can compute $Tr(g^{bx_i})^{-1}$ by using its secret shadow x_i and $Tr(g^b)$. Finally, P_i obtains its subshadow by $u_{i-1} = E_i * Tr(g^{bx_i})^{-1}$. Notice that we add consistence check between u_{i-1} and public information so that the malicious dealer can be found.

We assume that the dealer can provide a false P_i's subshadow u'_{i-1} $(u'_{i-1} \neq u_{i-1})$ successfully in the construction phase, where u_{i-1} is P_i's valid subshadow. This implies that $T_i = g^{u_{i-1}} \equiv g^{u'_{i-1}} \pmod{p^2}$, and

$$\prod_{j=0}^{k} (T_{i+1+k-j})^{\left(\begin{array}{c} k \\ j \end{array}\right)} = g^{c(-1)^i i} \pmod{p^2}$$

or

$$\prod_{j=0}^{k} (T_{i+1+k-j})^{(-1)^j \left(\begin{array}{c} k \\ j \end{array}\right)} = g^{ci} \pmod{p^2}.$$

Because $u_{i-1}, u'_{i-1} \in GF(q)$, the probability of $u'_{i-1} \neq u_{i-1}$ is negligible in the equations mentioned above, which means that it is impossible for the dealer to cheat participants successfully in the construction phase.

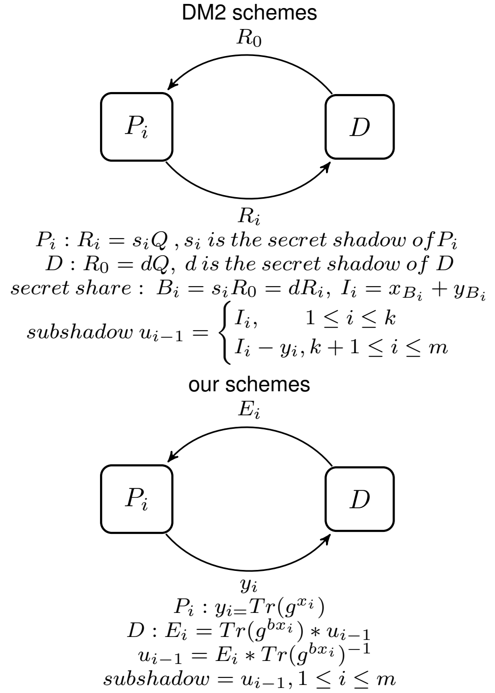
DM2 schemes
R_0
P_i
D
R_i
$P_i : R_i = s_iQ$, s_i is the secret shadow of P_i
$D : R_0 = dQ$, d is the secret shadow of D
secret share : $B_i = s_iR_0 = dR_i$, $I_i = x_{B_i} + y_{B_i}$
subshadow $u_{i-1} = \begin{cases} I_i, & 1 \leq i \leq k \\ I_i - y_i, & k+1 \leq i \leq m \end{cases}$
our schemes
E_i
P_i
D
y_i
$P_i : y_i = Tr(g^{x_i})$
$D : E_i = Tr(g^{bx_i}) * u_{i-1}$
$u_{i-1} = E_i * Tr(g^{bx_i})^{-1}$
subshadow $= u_{i-1}, 1 \leq i \leq m$

Fig. 1: The difference between DM2 schemes and our schemes

Theorem 7. In the recovery phase, it is impossible for the participant P_i to cheat other participants and the dealer.

Proof. When a malicious P_i provides u'_{i-1} $(u'_{i-1} \neq u_{i-1})$ in the recovery phase, it implies that other participants can obtain $T_i = g^{u_{i-1}} \neq g^{u'_{i-1}} \pmod{p^2}$, which means that the malicious participant can be found in the recovery phase.

Theorem 8. In the recovery phase, it is impossible for two conspirators P_i and P_j to collude to cheat other participants and the dealer.

Proof. If P_i and P_j conspire, they exchange their secret key x_i and x_j privately. Thus P_i gets u_{j-1} and P_j gets u_{i-1}, which means that they can pass through the verification phase. Nevertheless, all the participants have transmitted (ID_i, y_i) to the dealer D and D has released them in the initialization phase. In consequence, the published message pair can guarantee that other participants and the dealer will detect this conspiracy owing to the mismatch between ID_i and u_{j-1} or ID_j and u_{i-1} in the recovery phase.

5.3 Privacy

Theorem 9. We assume that discrete logarithm problem, XTR-DL problem with the base $g \in GF(p^6)^*$ is infeasible, and XTR public key system is secure. Then, the adversary cannot get anything about the secrets and subshadows.

Proof. From the description of our schemes, the public messages are as follows.

(1) (ID_i, y_i) for $i = 1, 2, \cdots, m$.

First, D has released y_i $(1 \leq i \leq m)$. If participants P_j $(j \neq i)$ or the dealer wants to derive secret key x_i from $y_i = Tr(g^{x_i})$, which means that the XTR-DL problem can be solved, it is impossible under our assumption. The adversary cannot decrypt u_{i-1} from E_i without x_i, so they cannot recover the secrets.

(2)$E_1, E_2, \cdots, E_m$.

We have $E_i = Tr(g^{bx_i}) * u_{i-1}$ $(1 \leq i \leq m)$, where E_i is the XTR encryption of P_i's subshadow u_{i-1}. When the adversary want to obtain u_{i-1} from E_i, they need to break XTR public key system. Therefore, the adversary cannot get any useful information of subshadows and secrets under the assumption.

(3)$T_1, T_2, \cdots, T_m$.

$T_i = g^{u_{i-1}}$ $(1 \leq i \leq m)$ have been released in the construction phase. Because the security of our schemes is based on the intractability of discrete logarithm problem with the base g in the finite field $GF(p^6)^*$, it is impossible to get u_{i-1} from T_i under the assumption.

Theorem 10. Any $k-1$ or fewer participants cannot reconstruct these secrets.

Proof. We might consider the worst case. Assume that there are exactly $k-1$ participants $\{P_i, P_{i+1}, \cdots, P_{i+k-2}\}$, which means that they only have $k+1$ terms, i.e., subshadows $\{u_{i-1}, u_i, \cdots, u_{i+k-3}\}$ and the published $\{u_{m+l}, u_{m+l+1}\}$. However, there are $k+2$ undefined coefficients of $p(x)$ mentioned in Theorem 1 and Theorem 2. Then we cannot determine the polynomial $p(x)$ uniquely. Therefore, it is impossible for them to recover other subshadows by using the corresponding nonhomogeneous linear recursions. Consequently, the secrets shared by the dealer cannot be derived from $k-1$ or fewer participants.

6 Performance analysis

6.1 Public values

At first, we compare other proposed schemes [12], [13], [15], [16], [19], [21] with our schemes from the perspective of the amount of public values, which is the main index to measure the efficiency of VMSS schemes.

In Table 1, we use abbreviation OS to represent our schemes. In order to compare these schemes, we assume that there are m participants, l shared secrets, and the threshold is k.

From Table 1, we know that LZZ, YF and our schemes need more public values than the other schemes. However, except these three schemes, the other presented schemes cannot resist some malicious behaviors of the dealer as mentioned in Section 2.3.2. Further, our schemes make use of XTR public key system, then we can achieve the same security level as LZZ and YF schemes with shorter key length. Therefore, our schemes are relatively efficient among all these schemes.

6.2 Computational complexity

Next, we compare the computational complexity of the presented schemes [12], [13], [15], [16], [19], [21] and our schemes. And we utilize the notations below in Table 2.

TABLE 2: Notations

Symbol	Explanation
T_e	cost of one modular exponentiation on some finite field
T_m	cost of one modular multiplication on some finite field
$T_L(i)$	cost of the Lagrange basis of i points, where $i \geq 0$
T_M	average cost of a scalar multiplication on the elliptic curve
$T_{le}(i)$	cost of obtaining a solution of i linear equations, where $i \geq 0$

Because all the schemes listed here are multi-use secret sharing schemes, which means that the initialization phase of every scheme needs to be performed only once, we ignore the cost of this phase in the following part. To save space, in Table 3, we use Con, Ver, Rec to represent the construction phase, verification phase, and recovery phase, respectively. Similarly, OS is on behalf of our schemes. In addition, we also assume that there are m participants, l shared secrets, and the threshold is k.

6.2.1 Construction phase

In the Scheme 1 of HLC and LZZ, they employ the polynomials of degree $k-1$ or $l-1$ to share secrets. However, DM1, DM2, DM3, YF, our schemes and the Scheme 2 of HLC, LZZ utilize the linear recursion. Because LZZ, YF and our schemes can detect the malicious behavior of the dealer, all these three schemes need more computations than the others.

The differences of computations among these three schemes lie in the different public key system used in them. Because the trace function used in the XTR public key system needs less time than modular multiplication computation used in RSA and LFSR public key cryptosystems, our schemes are faster to implement than LZZ and YF schemes in this phase.

What's more, the Scheme 1 of HLC and LZZ need two ways to deal with different cases, which are more complex to operate than the other schemes. Therefore, YF and our schemes are easier to run than LZZ.

Remark 4. Notice that we do not consider the cost of trace function in Table 3.

6.2.2 Verification phase

Except LZZ, YF and our schemes, the other schemes need less computations. Since these three schemes overcome the drawback mentioned before, they need more computations to verify the validity of shares. The serious consequences of the lack of these verification have been shown in Section 2.3.2.

TABLE 1: Comparison of the amount of the public values in proposed schemes

Scheme	Amount of public values		Public values
DM1 [12]	Type1 $2(m+3)+l-k$		$\{e, N, g, q, \alpha\}$, $(r, \{G_i\}_{i=1}^{m}, \{r_i\}_{i=1}^{l}, \{y_i\}_{i=k+1}^{m})$
	Type2 $2(m+3)+l-k$		$\{N, g, q, \alpha\}$, $\{R_i\}_{i=1}^{m}$ $(R_0, f, \{r_i\}_{i=1}^{l}, \{y_i\}_{i=k+1}^{m})$
DM2 [13]	Type1&2 $2(m+3)+l-k$		$\{N, Q\}$, $\{R_i\}_{i=1}^{m}$ $(R_0, e, \{r_i\}_{i=1}^{l}, \{y_i\}_{i=k+1}^{m},$ $u_{m+l+2}, u_{m+l+3})$
HLC [15]	Scheme1 $2m+l-k+6$		$\{N, a, b\}, \{ID_i, s_{e_i}(a, b)\}_{i=1}^{m}$ $(s_{e_0}(a, b), s_{-e_0}(a, b), d),$ $(\{Y_i\}_{i=1}^{m}, \{h(i)\}_{i=1}^{l-k})$
	Scheme2 $2m+l-k+7$		$\{N, a, b, \alpha, q_1\}$, $\{ID_i, s_{e_i}(a, b)\}_{i=1}^{m}$ $(s_{e_0}(a, b), d, \{r_i\}_{i=1}^{l}, \{Y_i\}_{i=k+1}^{m})$
DM3 [16]	Type1&2 $3m+l-k+7$		$\{N, a, b, q_1\}, \{ID_i, s_{e_i}(a, b)\}_{i=1}^{m},$ $(s_{e_0}(a, b), d, \{r_i\}_{i=1}^{l}, \{y_i\}_{i=k}^{m})$
LZZ [19]	Scheme1	$3m+k+5$, when $l \leq k$	$(\lambda, N, Q, q, g), \{ID_i, e_i, N_i\}_{i=1}^{m},$ $(\{C_i\}_{i=1}^{m}, \{H_i\}_{i=1}^{m}, \{A_i\}_{i=1}^{k})$
		$3m+3l-2k+5$, when $l > k$	$(\lambda, N, Q, q, g), \{ID_i, e_i, N_i\}_{i=1}^{m},$ $(\{C_i\}_{i=1}^{m}, \{H_i\}_{i=1}^{m}, \{\eta_i\}_{i=1}^{l-k},$ $\{f(\eta_i)\}_{i=1}^{l-k}, \{A_i\}_{i=1}^{l})$
	Scheme2 $3m+l+6$		$(\lambda, N, Q, q, g, \alpha)$, $\{ID_i, e_i, N_i\}_{i=1}^{m}$ $(\{H_i\}_{i=1}^{m}, \{T_i\}_{i=1}^{m}, \{Y_i\}_{i=1}^{l})$
YF [21]	Scheme1&2 $3m+l+7$		(λ, N, Q, q, g), $\{ID_i, e_i, N_i\}_{i=1}^{m}$ $(\{H_i\}_{i=1}^{m}, \{T_i\}_{i=1}^{m}, \{y_i\}_{i=1}^{l},$ $c, u_{m+l})$
OS	Scheme1&2 $3m+l+9$		$(\lambda, p, q, g, Tr(g), Tr(g^b))$, $\{ID_i, y_i\}_{i=1}^{m}$ $(\{E_i\}_{i=1}^{m}, \{T_i\}_{i=1}^{m}, \{z_i\}_{i=1}^{l},$ $c, u_{m+l}, u_{m+l+1})$

TABLE 3: Comparison of the computational complexity in presented schemes

Scheme	Con	Ver	Rec
DM1 [12]	$T_e + kT_m$	T_e	$kT_e + T_L(k)$
DM2 [13]	$2T_M + (k+1)T_m$	T_M	Way1 $T_L(k+2)$ Way2 $T_{le}(k)$
HLC [15]	Scheme1 $3T_e + kT_m$ $(l \leq k)$ $3T_e + lT_m$ $(l > k)$	T_e	Scheme1 $T_L(k)$ $(l \leq k)$ $T_L(l)$ $(l > k)$
	Scheme2 $3T_e + kT_m$		Scheme2 $T_L(k)$
DM3 [16]	$3T_e + kT_m$	T_e	Way1,2 $T_L(k)$ Way3 $T_{le}(k)$
LZZ [19]	Scheme1 $2T_e + kT_m$ $(l \geq k)$ $2T_e + lT_m$ $(l > k)$	Scheme1 T_e $(l \leq k)$ $2T_e$ $(l > k)$	Scheme1 $T_L(k)$ $(l \leq k)$ $T_L(l)$ $(l > k)$
	Scheme2 $2T_e + kT_m$	Scheme2 $2T_e$	Scheme2 $T_L(k)$ *or* $T_{le}(k)$
YF [21]	$2T_e + (k+1)T_m$	$2T_e$	Way1 $T_L(k+1)$ Way2 $T_{le}(k)$
OS	$T_e + (k+1)T_m$	$2T_e$	Way1 $T_L(k+2)$ Way2 $T_{le}(k)$

6.2.3 *Recovery phase*

The recovery phase is the most time-consuming phase in these phases. In fact, all schemes mentioned here can use Lagrange interpolation polynomial to recover the shared secrets. However, DM1, DM2, DM3, YF, our schemes and the Scheme 2 of HLC, LZZ can make use of linear recursions to reconstruct secrets, which are much easier and faster to construct than Lagrange interpolation polynomial.

Because a polynomial of degree n needs $O(n^2)$ time to construct by Lagrange interpolation, the recovery phase of Scheme 1 of HLC and LZZ can be operated within $O(k^2)$ $(l \leq k)$ or $O(l^2)$ $(l > k)$ time.

YF and our schemes have two ways to recover the secrets. As for the Way 1, YF schemes need $O(k^2)$ time, and our schemes need $O((k+1)^2)$ time. Because we use different nonhomogeneous linear recursions from YF schemes, our schemes need more computations. The Way 2 is easier to implement than the first way, however it has stricter condition, which means that corresponding participants' indices must be consecutive. Since the nonhomogeneous linear recursions used in YF and our schemes are both k-th order, these two schemes need k terms of subshadows to determine the nonhomogeneous linear recursions used in the construction phase, which means that they have the same computational complexity in the recovery phase for Way 2.

Remark 5. Compared with LZZ schemes using homogeneous linear recursions, YF and our schemes need more computations in Way 1, because these two schemes utilize nonhomogeneous linear recursions which are more complex than homogeneous linear recursions. Nevertheless, if we utilize the same linear recursion in these three schemes, they will cost the same time in the recovery phase for Way 1.

6.3 Dynamic attribute

Then, we will show a dynamic update, deletion, addition of the participants, the values of secrets and the threshold according to the actual situation.

Participants:

When a participants P_{new} needs to be added in the scheme, P_{new} selects an integer x_{new} $(1 < x_{new} < q)$ randomly and computes $y_{new} = Tr(g^{x_{new}})$, then transmits (ID_{new}, y_{new}) to the dealer. Next, D can compute $E_{new} = Tr(g^{bx_{new}}) * u_{new-1} \pmod{q}$ and $T_{new} = g^{u_{new-1}}$ $(\bmod\ p^2)$ where $q|(p^2 - p + 1)$, and it releases them later. Similarly, when the scheme needs to delete a participant P_{del}, D only erases (ID_{del}, y_{del}) from its list. Therefore, if P_{del} wants to attack the scheme by its subshadow u_{del-1}, the dealer will detect this malicious behavior.

Secrets:

Once D wants to add a secret S_{l+1} to the scheme, D can obtain $z_{l+1} = S_{l+1} - u_{m+(l+1)-1} = S_{l+1} - u_{m+l}$. Likewise, D can delete a secret S_i by erasing $z_i = S_i - u_{m+i-1}$ from the list. If D wants to update the secrets, D only erases the old secrets and then add the new one into the scheme by corresponding operations mentioned above.

Threshold:

Our schemes are secure (k, l, m)-VMSS schemes, because our schemes make use of a NLR of degree k. Therefore, if D wants to change the threshold, D can replace the original NLR with a new degree, which means that our schemes are threshold changeable multi-secret sharing schemes.

6.4 Performance feature

Finally, we analyze performance features of the schemes in [12], [13], [15], [16], [19], [21] and our schemes in Table 4.

• Feature 1: Reconstruct multiple secrets at the same time

• Feature 2: Utilize the public channel

• Feature 3: Resist the conspiracy attack

• Feature 4: Update the secrets after an unsuccessful recovery

• Feature 5: Reuse the shadows with different access structure

• Feature 6: Reuse the shadows with different D

• Feature 7: Perceive D's deception

• Feature 8: Perceive P_i's deception

• Feature 9: The bit length of private key in a 1024-bit finite field

• Feature 10: The bit length of public key in a 1024-bit finite field

From Table 4, we know that DM1 and DM2 schemes cannot resist conspiracy attack as analyzed in Theorem 8, because the dealer does not construct the links between the identity messages and corresponding secret shadows of the specific participants.

Notice that, except LZZ, YF and our schemes, the other schemes cannot perceive malicious dealer, since they lack verification between their participants' subshadows and public messages.

Nevertheless, in a 1024-bit finite field, the length of our private key can be one-sixth of LZZ schemes, and one-third of YF schemes, which is about 170 bits. And the length of the public key can be one-third of LZZ schemes, and equal to YF schemes, which is about 340 bits. This is because the private key x_i is in $GF(q)$, and the public key $Tr(g^{x_i})$ is in $GF(p^2)$. What's more, the XTR public key system used in our schemes can realize the security level in $GF(p^6)$ by computations in $GF(p^2)$. It has been proved that the security level of a 170-bit XTR is equivalent to a 340-bit LFSR public key cryptosystem or a 1024-bit RSA public key cryptosystem, which means that our schemes can achieve the same security level as LZZ and YF schemes with shorter key size.

Therefore, our schemes are better schemes than the other schemes mentioned in this section.

Remark 6. If we can use LFSR sequences with a higher order, such as sixth-order, to construct a new LRSR sequence public key system applied to our schemes, then we will use shorter key size. However, there is not necessarily a fast way to get the required parameters in the whole scheme. So we choose XTR public key system generated by a third-order LFSR sequence to construct our new VMSS schemes.

7 CONCLUSION

In this paper, we utilize XTR public key system to construct two new efficient VMSS schemes which are improved versions of the VMSS schemes proposed by Dehkordi and Mashhadi in 2008.

TABLE 4: Performance feature

Feature	DM1 [12]	DM2 [13]	HLC [15]	DM3 [16]	LZZ [19]	YF [21]	OS
1	YES	YES	YES	YES	YES	YES	YES
2	YES	YES	YES	YES	YES	YES	YES
3	NO	NO	YES	YES	YES	YES	YES
4	NO	NO	NO	NO	NO	NO	NO
5	YES	YES	YES	YES	YES	YES	YES
6	YES	YES	YES	YES	YES	YES	YES
7	NO	NO	NO	NO	YES	YES	YES
8	YES	YES	YES	YES	YES	YES	YES
9	1024	1024	340	340	1024	340	170
10	1024	1024	340	340	1024	340	340

Compared with the previous presented schemes, our schemes can detect the malicious dealer by adding verification between participants' subshadows and public messages. Even though LZZ and YF schemes have the same advantages as our schemes, we use shorter key size to achieve the same security level. In addition, our schemes are efficient to implement because they have dynamic attributes, which means that our schemes can change the number of participants, the values of secrets and the threshold easily according to the practical situation.

In conclusion, our schemes are computationally secure (k, l, m)-VMSS schemes which can share multiple secrets simultaneously, use the public channel, have verifiability, reuse subshadows, and are both dynamic and threshold changeable with shorter parameters.

Acknowledgments

This research is supported by the National Key Research and Development Program of China (Grant No. 2018YFA0704703), the National Natural Science Foundation of China (Grant No. 61971243), the Fundamental Research Funds for the Central Universities of China, and the Nankai Zhide Foundation.

References

[1] A. Shamir, ”How to share a secret”, *Commun. ACM*, vol. 22, no. 11, pp. 612-613, 1979.

[2] G. R. Blakley, ”Safeguarding cryptographic keys”, in *Proc. 1979 AFIPS National Computer Conference*. New York, USA: AFIPS Press, 1979, pp. 313-318.

[3] Y. Desmedt and Y. Frankel, ”Threshold cryptosystems,” in *Advances in Cryptology-CRYPTO*. Berlin, Germany: Springer-Verlag, 1989, pp. 307-315.

[4] D. Boneh and M. Naor, ”Timed commitments,” in *Advances in Cryptology-CRYPTO*. Berlin, Germany: Springer-Verlag, 2000, pp. 236-254.

[5] R. Cramer, I. Damgård , U. Maurer, ”General secure multi-party computation from any Linear secret-sharing scheme,” in *Advances in Cryptology-EUROCRYPT*. Berlin, Germany: Springer-Verlag, 2000, pp. 316-334.

[6] R. Cramer, V. Daza, I. Gracia, J. J. Urroz, G. Leander, J. Martí-Farré and C. Padró, ”On codes, matroids, and secure multiparty computation from linear secret-sharing schemes,” *IEEE Trans. Inf. Theory*, vol. 54, no. 6, pp. 2644-2657, Jun. 2008.

[7] Y. Kim, R. K. Raman, Y. Kim, L. R. Varshney and N. R. Shanbhag, ”Efficient local secret sharing for distributed blockchain systems,” *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 282-285, Feb. 2019.

[8] C. C. Yang, T. Y. Chang, and M. S. Hwang, ”A (t, n) multi-secret sharing scheme,” *Appl. Math. Comput.*, vol. 151, no. 2, pp. 483-490, Apr. 2004.

[9] J. Shao and Z. Cao, ” A new efficient (t, n) verifiable multi-secret sharing (VMSS) based on YCH scheme,” *Appl. Math. Comput.*, vol. 168, no. 1, pp. 135-140, Sep. 2005.

[10] J. Zhao, J. Zhang, and R. Zhao, ” A practical verifiable multi-secret sharing scheme,” *Comput. Stand. & Interfaces*, vol. 29, no. 1, pp. 138-141, Jan. 2007.

[11] M. H. Dehkordi and S. Mashhadi, ”An efficient threshold verifiable multi-secret sharing,” *Comput. Stand. & Interfaces*, vol. 30, no. 3, pp. 187-190, Mar. 2008.

[12] M. H. Dehkordi and S. Mashhadi, ”New efficient and practical verifiable multi-secret sharing schemes,” *Inf. Sci.*, vol. 178, no. 9, pp. 2262-2274, May. 2008.

[13] M. H. Dehkordi and S. Mashhadi, ”Verifiable secret sharing schemes based on non-homogeneous linear recursions and elliptic curves,” *Comput. Commun.*, vol. 178, no. 9, pp. 2262-2274, May. 2008.

[14] N. L. Biggs, *Discrete Mathematics*, 2nd ed. New York, USA: Oxford University Press, Inc., 2002.

[15] C. Hu, X. Liao, and X. Cheng, ”Verifiable multi-secret sharing based on LFSR sequences,” *Thero. Comput. Sci.*, vol. 445, no. 11, pp. 52-62, Aug. 2012.

[16] S. Mashhadi and M. H. Dehkordi, ”Two verifiable multi secret sharing schemes based on nonhomogeneous linear recursion and LFSR public-key cryptosystem,” *Inf. Sci.*, vol. 294, no. 10, pp. 31-40, Feb. 2015.

[17] G. Gong and L. Harn, ”Public-key cryptosystems based on cubic finite field extensions,” *IEEE Trans. Inf. Theory*, vol. 45, no. 7, pp. 2601-2605, Nov. 1999.

[18] G. Gong and L. Harn, ”The GH public-key cryptosystem,” in *Selected Areas in Cryptography (Lecture Notes in Computer Science)*. Berlin, Germany: Springer-Verlag, 2001, pp. 284-300.

[19] Y. Liu, F. Zhang, and J. Zhang, ”Attacks to some verifiable multi-secret sharing schemes and two improved schemes,” *Inf. Sci.*, vol. 329, no. 1, pp. 524-539, Feb. 2016.

[20] R. L. Rivest, A. Shamir, and L. Adleman, ” A method for obtaining digital signatures and public-key cryptosystems,” *Commun. ACM*, vol. 21, no. 2, pp. 120-126, Feb. 1978.

[21] J. Yang and F. Fu, ”New dynamic and verifiable multi-secret sharing schemes based on LFSR public key cryptosystem,” *IET Inf. Secur.*, vol. 14, no. 6, pp. 783-790, Nov. 2020.

[22] A. K. Lenstra and E. R. Verheul, ”The XTR public key system,” in *Advances in Cryptology-CRYPTO*. Berlin, Germany: Springer-Verlag, 2000, pp. 1-19.

[23] A. K. Lenstra and E. R. Verheul, ”Key improvements to XTR,” in *Advances in Cryptology-ASIACRYPT*. Berlin, Germany: Springer-Verlag, 2000, pp. 220-233.

Jing Yang received the B.S. degree in mathematics and applied mathematics from Shandong Normal University, Jinan, China in 2015, and M.S. degree in Probability and Mathematical Statistics from Nankai University, Tianjin, China in 2019. She is currently a Ph.D. student advised by Prof. Fang-Wei Fu in Chern Institute of Mathematics and LPMC, and Tianjin Key Laboratory of Network and Data Security Technology, Nankai University, Tianjin, China. Her research interests include secret sharing, blockchain, and corresponding cryptography.

Fang-Wei Fu received the B. S. degree in mathematics, the M. S. degree, and the Ph.D. degree in applied mathematics from Nankai University, Tianjin, China, in 1984, 1987 and 1990, respectively. Since April 2007, he has been with the Chern Institute of Mathematics, Nankai University, Tianjin, China, where he is a Professor. From June 1987 to April 2007, he was with the School of Mathematical Science, Nankai University, Tianjin, China, and became a Professor there in 1995. From February 2002 to March 2007, he was a Research Scientist with the Temasek Laboratories, National University of Singapore, Republic of Singapore. From November 1989 to November 1990, he visited the Department of Mathematics, University of Bielefeld, Germany. From October 1996 to October 1997, he visited the Institute for Experimental Mathematics, University of Duisburg-Essen, Germany. He also visited the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong, the Department of Mathematics, University of California, Irvine, USA, the Division of Mathematical Sciences, the School of Physical and Mathematical Sciences, Nanyang Technological University, Republic of Singapore. His current research interests include coding theory, cryptography, and information theory.