Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 3150714, 9 pages
https://doi.org/10.1155/2022/3150714

Research Article

A New Code-Based Linkable Threshold Ring Signature Scheme

Fang Ren,[1,2] Haiyan Xiu,[1,2] Chuxin Ji,[1] Dong Zheng,[1,2] and Ziyi Wu[1]

[1]*School of Cyberspace Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China*
[2]*National Engineering Laboratory for Wireless Security, Xi'an University of Posts and Telecommunications, China*

Correspondence should be addressed to Haiyan Xiu; 13210145429@163.com

Received 4 August 2022; Revised 7 September 2022; Accepted 14 September 2022; Published 3 October 2022

Academic Editor: Lei Liu

In this paper, we construct a new code-based linkable threshold signature scheme whose security is based on the hardness of the syndrome decoding problem and general syndrome decoding problem. We show that our scheme is secure in terms of existential unforgeability, anonymity, nonslanderability, and linkability. The complexity of the signature size proposed in this paper is $O(t)$. Our method is particularly well suited for large free group voting systems. The greater the number of members in the ring, the more pronounced the advantage of the signature length is when compared to other schemes. That is, our scheme achieves a fixed length signature, independent of the number of ring members. In addition, our scheme has a very short public key, and the size is $O(N)$.

1. Introduction

The ring signature scheme is a special type of digital signature that was presented in 2001 by Rivest et al. [1]. It may also be regarded as a generalization of group signatures, while it varies from group signatures [2] in that ring signatures do not have a group administrator. Ring signature schemes enable a ring member to sign a message on the ring's behalf while maintaining the anonymity of this ring's members. In other words, although the actual message's signer is unknown to the verifier, the verifier is aware that the signer belongs to this ring.

However, in some applications, such as e-voting, in order to limit the ability of an individual to sign, t members must be willing to cooperate to generate a valid signature. Bresson et al. [3] used the notion of partitioning and combining functions to extend the ring signature scheme into a threshold ring signature scheme. A subset of t-out-of-N users in such a scheme cooperates to construct a signature while keeping the identity of the subgroup of t users secret.

The first effective code-based threshold ring signature system was proposed by Melchor et al. [4]. Comparatively speaking, to other threshold ring signature schemes based on number theory, this scheme is designed based on the coded-Stern's signature protocol, which is extremely quick and resistant to quantum attacks. The complexity of the signature length in this scheme is $O(N)$. The hardness of the q-ary syndrome decoding problem was used by Cayrel et al. [5] to construct a code-based threshold ring signature scheme in 2012. A zero-knowledge identification protocol based on coding theory called AGS was the foundation for the threshold ring signature scheme introduced by Assidi et al. [6] in 2019. This scheme utilizes a double circulant code, which reduces the size of the public key. At the same time, the cheating probability of the AGS protocol is asymptotically equal to 1/2; thus, the number of repeated runs of the protocol becomes less, making the signature length significantly smaller, with a complexity of $O(Nt)$. The above schemes all use the Fiat-Shamir method to convert the identity authentication protocol into the threshold ring signature scheme, while another way to construct threshold ring signature is to combine the CFS signature algorithm with the Bresson threshold ring signature. In 2009, Dallot and Vergnaud [7]. proposed a provable security threshold ring signature scheme based on the CFS scheme. The Goppa parameterized bounded decoding problem and Goppa code distinguisher are the key to ensure the security of this scheme. Its signature length is 675N-228l, where N is the total number of ring members and l denotes how many signers there are.

In some applications, it may be desirable to keep the signer's anonymity in a ring and provide the verifier the opportunity to determine whether two signatures on the same issues were issued by the same person. This is the driving force for the development of linkable ring signature schemes [8].

There are many linkable ring schemes based on the factoring and discrete logarithm problem [8–13]. Therefore, with the development of quantum computers, Shor's algorithm poses a danger to their security [14]. In 2018, two quantum-resistant lattice-based linkable ring signatures schemes [15, 16] were proposed. The first code-based linkable ring signature scheme was presented by Branco and Mateus [17] in the same year. The complexity of this scheme is $O(N)$ for both the public key size and the signature length. The public key size and signature length in [17] are slightly larger than those in [15, 16], but the private key size in [17] is much smaller than those in both schemes [15, 16].

To solve the problem of the multicandidate voting problem, Yuen et al. [18] proposed a linkable threshold ring signature scheme in 2013. Specifically, the N members of the management committee can vote for multiple candidates, and each candidate can only become a candidate if he gets at least d votes. This scheme achieves a signature size of $O(d\sqrt{n})$. A lattice-based linkable threshold ring signature scheme is proposed in [19], which is suitable for applications that can be applied to multicategory voting systems while having a small signature length.

However, the linkable threshold ring signature proposed in this paper is different from all of the above schemes. We improve on that of [17] by taking into account the following scenarios. For instance, in free group voting, there are a total of N people in the class; any t students in the class can form a group to participate in voting, but each person in the class can only vote in at most one group. For one issue, each group can only participate in at most one vote; if someone participates in more than one group to vote multiple times, the signature will be linked. At the same time, we can know how many dishonest group members there are. To solve this problem, we propose a linkable threshold ring signature scheme. In the above example, N is the number of ring members and t is the threshold value.

In this paper, we give the first construction of a code-based linkable threshold ring signature scheme whose security is based on the general syndrome decoding (GSD) problem which is an NP problem. We prove that the linkable threshold ring signature scheme proposed in this paper has the security properties of existential unforgeability, anonymity, and linkability. To construct our proposal, we also give a variant of threshold GStern's protocol, and then, we apply the Fiat-Shamir transform to it. We also prove that the threshold ring signature scheme proposed in this paper has security properties of complete, special sound, and honest-verifier-zero-knowledge (HVZK).

Overall, our protocol has a signature length linear in t and the best-known complexity on $O(t)$ when other number theory-based threshold ring signature schemes have complexity in $O(Nt)$ and those based on code theory have complexity in $O(N)$. Our protocol has a public key size linear in N, and the complexity is $O(N)$.

In our proposal, signatures can be linked to each other by the same vector r_i, which is the syndrome of the secret key e_i and a random matrix G, where G is generated by all public keys of the members in the ring and issue.

This paper is structured as follows: in the next section, the necessary preliminary knowledge needed in this paper is introduced. In Section 3, we present our threshold ring signature scheme and linkable threshold ring signature scheme. Section 4 is devoted to the security analysis of our proposed schemes. Experimental result analyses including key cost, signature size, efficiency, and property comparison are shown in Section 5. Finally, the conclusion is drawn in Section 6.

2. Preliminaries

In this section, we give the definition of the threshold ring signature scheme and linkable ring signature scheme. Then, we introduce the hard problem in coding theory that our scheme is based on. Finally, we present the security model we adopt.

2.1. Threshold Ring Signature. A (t, N)-threshold ring signature scheme [6] consists four of polynomial time algorithms, defined as follows.

(1) *T.setup(1^λ)*: this algorithm generates system parameters with global public values by using the security parameter string 1^λ as input

(2) *T.KeyGen(params)*: this is a probabilistic polynomial-time (PPT) that accepts public parameters as input and returns a pair of secret and public keys as output (sk, pk).

(3) *T.Sign(params, t, $pk^{(N)}$, $sk^{(t)}$, m)*: this is a PPT algorithm for t users of a ring R that accepts public parameters params, a set $pk^{(N)}$ of public keys, a set of t secret keys $sk^{(t)}$, and a message m as input. On message m, the algorithm generates a (t, N)-threshold ring signature

(4) *T.Verify(params, t, $pk^{(N)}$, m, σ)*: this is a deterministic polynomial-time (DPT) that accepts as input public parameter params, a threshold value t, a set $pk^{(N)}$ of public keys, and a pair message/signature (m, σ) and returns 1 if the signature on m is valid with regard to the set of public key $pk^{(N)}$ and 0 otherwise

2.2. Linkable Ring Signature. A linkable ring signature scheme [16] consists of four polynomial time algorithms, defined as follows.

(1) $(pk, sk) \longleftarrow \mathrm{KeyGen}(1^\lambda)$ is a PPT algorithm that takes a security parameter λ as input and produces a pair of public and secret keys (pk, sk) as output

(2) $\sigma \longleftarrow \mathrm{Sign}(1^\lambda, pk\, M, sk)$ is a PPT algorithm that takes a security parameter λ, actual signers' public

key pk, the corresponding secret key sk, and a message M to be signed as inputs. It generates a signature σ

(3) $b \longleftarrow \mathrm{Ver}(1^\lambda, pk\, M, \sigma)$ is a DPT algorithm that takes a security parameter λ, all ring members' public key pk, a message M, and a signature σ as input. Return "1" (indicating a valid signature) and "0" (indicating an invalid signature)

(4) $b \longleftarrow \mathrm{Link}(1^\lambda, pk, M_1, \sigma_1, M_2, \sigma_2)$ is a DPT algorithm that accepts a list of public key pk, two messages M_1 and M_2, and two signatures σ_1 and σ_2 as input, where $\mathrm{Ver}(1^\lambda, pk\, M_1, \sigma_1) = 1$ and $\mathrm{Ver}(1^\lambda, pk\, M_2, \sigma_2) = 1$, respectively. For linked signatures, it returns 1; otherwise, it returns 0

2.3. Hard Problems

2.3.1. Syndrome Decoding (SD) Problem. Let H be the parity-check matrix of a random$[n, k]$-linear code on F_2, $s \in F_2^{n-k}$, w is a positive integer. The problem is to find an $e \in F_2^n$ satisfying $wt_H(e) \le w$ and $s = He^T$.

2.3.2. General Syndrome Decoding (GSD) Problem. Given H, $G \in Z_2^{(n-k)\times n}$, $s, r \in Z_2^{(n-k)}$, and w is a positive integer. The problem is to find an $e \in F_2^n$ satisfying $wt_H(e) \le w$, $He^T = s$, $Ge^T = r$.

In [20], it is pointed out that by selecting $H = G$ and $s = r$ as inputs, the SD problem can be simplified to the GSD problem by Karp reduction. Therefore, SD and GSD problems are equivalent. Because the SD problem is NP-complete, the GSD problem is NP-complete.

2.4. Security Model. The security model we adopt is based on [9, 10], which enhances the security of the linkable ring signature scheme first proposed by Liu in 2004, capturing new and practical attacking scenarios and the properties more thoroughly. In this security model, the scheme must contain these properties: existential unforgeability, anonymity, nonslanderability, and linkability.

Suppose A is a PPT adversary and the security parameter is λ. Let N be the number of members in the ring; $\overline{pk} = (pk_1, \cdots, pk_N)$ is the set of public keys of all members in the ring and $L \subseteq \overline{pk}$. A signing oracle called $\mathrm{Sign}(\cdot, sk)$ takes queries of the form $(\overline{pk}, M)$ and returns $\sigma \longleftarrow \mathrm{Sign}(1^\lambda, \overline{pk}, M, sk_i)$, for every i between 1 and N. $\mathrm{Co}(\cdot)$ is a corruption oracle that receives queries of the public key and outputs the responding secret key.

Existential unforgeability: for each message M, if the adversary lacks a public key of this group, he is unable to forge a signature on behalf of this group. Consider the following game:

where (L, M) was not asked $\mathrm{Sign}(\cdot, sk)$ and A merely requested $pk \notin L$ from $\mathrm{Co}(\cdot)$ [21]. The advantage that A wins the game of existentially unforgeable is

$$\mathrm{Adv}_A^{\mathrm{unf}}(\lambda, N) \coloneqq \Pr[b = 1]. \tag{1}$$

$\mathrm{Game}_A^{\mathrm{unf}}(\lambda, N)$:
1. $(pk_i, sk_i) \longleftarrow \mathrm{KeyGen}(1^\lambda) \quad i = 1, \cdots, N.$
2. $(L, M, \sigma) \longleftarrow A^{\mathrm{Sign}(\cdot, sk_i), Co(\cdot)}(\overline{pk})$.
3. $b \longleftarrow \mathrm{Ver}(1^\lambda, L, M, \sigma)$.
4. **return** b

Algorithm 1: Game for existential unforgeability.

$\mathrm{Game}_A^{\mathrm{anon}}(\lambda, N)$:
1. $(pk_i, sk_i) \longleftarrow \mathrm{KeyGen}(1^\lambda) \quad i = 0, 1.$
2. $b \longleftarrow^{\$} \{0, 1\}$.
3. $b' \longleftarrow A^{\mathrm{Sign}(\cdot, sk_b), \mathrm{Sign}(\cdot, sk_0), \mathrm{Sign}(\cdot, sk_1)}(pk_0, pk_1)$.
4. **return** b'

Algorithm 2: Game for anonymity.

$\mathrm{Game}_A^{\mathrm{nsla}}(\lambda, N)$:
1. $(pk_i, sk_i) \longleftarrow \mathrm{KeyGen}(1^\lambda) \quad i = 1, \cdots, N.$
2. $(L, pk_1, M_1) \longleftarrow A^{\mathrm{Sign}(\cdot, sk_i)}(\overline{pk})$.
3. $\sigma_1 \longleftarrow \mathrm{Sign}(1^\lambda, L, M_1, sk_1)$.
4. $(pk_2, M_2, \sigma_2) \longleftarrow A^{\mathrm{Sign}(\cdot, sk_i)}(\overline{pk}, L, sk_2, pk_1, M_1, \sigma_1)$.
5. $b \longleftarrow \mathrm{Link}(1^\lambda, L, M_1, \sigma_1, M_2, \sigma_2)$.
6. **return** b

Algorithm 3: Game for nonslanderability.

If we can prove that $\mathrm{Adv}_A^{\mathrm{unf}}(\lambda, N)$ is negligible for all PPT adversaries, that is $\mathrm{Adv}_A^{\mathrm{unf}}(\lambda, N) \le negl(\lambda, N)$, then the linkable threshold ring signature scheme is existentially unforgeable.

Anonymity: an adversary cannot determine which member of the group has signed a specified message thanks to anonymity. Consider the following game:

where the adversary is not permitted to ask $\mathrm{Sign}(\cdot, sk_b)$ questions with different L, nor to ask both $\mathrm{Sign}(\cdot, sk_b)$ and $\mathrm{Sign}(\cdot, sk_0)$ or to both $\mathrm{Sign}(\cdot, sk_b)$ and $\mathrm{Sign}(\cdot, sk_1)$ with the same L. The advantage that A wins the game of anonymity is

$$\mathrm{Adv}_A^{\mathrm{anon}}(\lambda, N) \coloneqq \Pr\left[b = b'\right] - \frac{1}{2}. \tag{2}$$

If we prove that $\mathrm{Adv}_A^{\mathrm{anon}}(\lambda, N) \le negl(\lambda, N)$, the scheme is anonymous.

Nonslanderability: an adversary cannot generate a signature that is connected to another user's signature thanks to nonslanderability, which was initially mentioned in [11] and standardized in [8]. Consider the following game:

where $pk_1, pk_2 \in L$, $pk_1 \ne pk_2$ and neither (L, M_1), σ_1 nor (L, M_2), σ_2 was questioned nor responded to by $\mathrm{Sign}(\cdot, sk_i)$. The advantage that A wins the game of nonslanderability is

$$\mathrm{Adv}_A^{\mathrm{nsl}}(\lambda, N) \coloneqq \Pr[b = 1]. \tag{3}$$

$\text{Game}_A^{\text{link}}(\lambda, N)$:
1. $(pk_i, sk_i) \longleftarrow \text{KeyGen}(1^\lambda) \quad i = 1, \cdots, N.$
2. $pk \longleftarrow A^{\text{Sign}(\cdot, sk_i)}(\overline{pk})$.
3. $(L, M_1, \sigma_1, M_2, \sigma_2) \longleftarrow A^{\text{Sign}(\cdot, sk_i)}(\overline{pk}, pk)$.
4. $b \longleftarrow 1 - \text{Link}(1^\lambda, L, M_1, \sigma_1, M_2, \sigma_2)$.
5. **return** b

Algorithm 4: Game for linkability.

If we prove that $\text{Adv}_A^{\text{link}}(\lambda, L) \leq negl(\lambda, N)$, the scheme has nonslanderability.

Linkability: this property ensures that it is impossible for a user to provide two valid signatures without linking. Consider the following game:

where $\text{Ver}(1^\lambda, L, M_1, \sigma_1) = 1, \text{Ver}(1^\lambda, L, M_2, \sigma_2) = 1$ and neither (L, M_1), σ_1 nor (L, M_2), σ_2 were questioned or responded to by $\text{Sign}(\cdot, sk_i)$. The advantage that A wins the game of linkability is

$$\text{Adv}_A^{\text{link}}(\lambda, L) \coloneqq \Pr[b = 1]. \tag{4}$$

If we prove that $\text{Adv}_A^{\text{link}}(\lambda, L) \leq negl(\lambda, N)$, the scheme has linkability.

3. Proposed Scheme

3.1. Proposed Threshold GStern's Protocol. In this section, we explain the main algorithm that composes our proposed threshold ring signature as given in Section 2.1.

As the go-between of both the t signers and the verifier V, the leader L is also one of the t signers. Let the i-th signer's first, second, and third commitments be denoted as co $m_{1,i}$, $com_{2,i}$, and $com_{3,i}$, respectively. The first, second, and third commitments produced by the leader L will be denoted as COM_1, COM_2, and COM_3, accordingly.

(1) *Commitment phase*: for all $1 \leq i \leq t$, commitments $com_{1,i}$, $com_{2,i}$, and $com_{3,i}$ are computed by the leader L and other $t - 1$ signers. As one of the t signers, L also calculates his commitments. Then, on the basis of the commitments of each signer, L utilizing them constructs the master commitments COM_1, COM_2, and COM_3. After that, L sends COM_1, COM_2, and COM_3 to the verifier V

(2) *Challenge phase*: the verifier V sends a random challenge 0, 1, or 2 to L who broadcasts it to other $t - 1$ signers

(3) *Response phase*: the signers construct their responses based on the value of the given challenge and then transmit them to the leader L. After that, L delivers these responses, together with the data needed for verification, to V

(4) *Verify phase*: according to the challenge's value, the verifier V validates COM_1 and COM_2, COM_1 and COM_3, or COM_2 and COM_3

3.2. Proposed Linkable Ring Signature Scheme. In this section, we give the description of our new linkable threshold ring signature scheme in Algorithm 6. Our scheme is improved based on the linkable ring signature scheme [17]. More precisely, our scheme is constructed by using the noninteractive protocol obtained by applying the Fiat-Shamir transform [22] to threshold GStern's protocol.

First, considering a ring with N users, use the public information $\overline{pk} = (pk_1, \cdots, pk_N)$ and a public cryptographic hash function g to construct the matrix G. Then, generate a set of random syndromes $r = (r_1, \cdots, r_N)$, where some r_i are the vectors that have a linear relationship with the actual signer's private information e_i.

Then, apply the Fiat-Shamir transform to the threshold GStern's protocol on input (H, G, s, r), where $s = (s_1, \cdots, s_N)$ are the public information of all the ring members. From one point of view, the SD hard problem prevents the verifier from knowing which users computed r. From another point of view, with regard to the same ring, r will be a component of each signature that these users issue. If for the same issue, the intersection of the set r in two valid message signatures is not empty, the verifier links the two signatures together and outputs r. The number of signers involved in multiple signatures can be known by checking the number of identical r_i.

4. Security Analysis

4.1. Security Analysis of Threshold Ring Signature. We prove that the protocol presented in Algorithm 5 satisfies the three properties: completeness, soundness, and HVZK.

Completeness: it is obvious that honest provers with access to a valid secret key can respond to any of the honest leader's queries with the proper information, enabling him to calculate the master commitments. Meanwhile, to enable the honest verifier to confirm that these commitments are accurate, the leader is then allowed to provide the information needed. This means that the verifier will always accept the signature signed by the honest prover according to the above process. For example, for $b = 0$, the verifier can reconstruct the master commitment COM_1 and COM_2 using the knowledge of com_i by utilizing the knowledge of y_i and δ_i to recover all the com_i. Cases $b = 1$ and $b = 2$ behave similarly. Thus, the protocol has perfect completeness.

Soundness: the proof of soundness for our protocol is identical to the proofs for Stern's protocol since it is based on the generic construction in [23]. The soundness error is confined by 2/3 since the GSD problem is difficult. Meanwhile, our protocol may be thought of as a collection of t parallel GSD scheme executions. This means that the soundness error for our protocol in a single round cannot exceed 2/3. More specifically, we consider the following cases according to the various values of challenges b and b'.

(1) When $b = 0$ and $b' = 1$, e_i can be extracted from y_i and $y_i + e_i$ by the simulator

(2) When $b = 0$ and $b' = 2$, e_i can be extracted from δ_i and $\delta_i(e_i)$ by the simulator

(1) Parameters: $n, k, t \in Z$
(2) Private information: $e_i \longleftarrow^{\$} \{0,1\}^n$ satisfying $w(e_i) = t$
(3) Public information: H_1, s, H_2, r, where $s = (s_1, \cdots, s_N)$, $r = (r_1, \cdots, r_N)$ satisfying $H_1 e_i^T = s_i^T$ and $H_2 e_i^T = r_i^T$
(4) The prover P: chooses $y_i \longleftarrow^{\$}$, permutation δ_i, randomness a, b, c for commitments.
  (i) let $\text{com}_{1,i} = h(\delta_i, Hy_i^T, Gy_i^T)$, $\text{com}_{2,i} = h(\delta_i(y_i))$, and $\text{com}_{3,i} = h(\delta_i(y_i + e_i))$.
  (ii) let $\text{COM}_1 = h(\text{com}_{1,1} \| \cdots \| \text{com}_{1,t} \| a)$, $\text{COM}_2 = h(\text{com}_{2,1} \| \cdots \| \text{com}_{2,t} \| b)$, and $\text{COM}_3 = h(\text{com}_{3,1} \| \cdots \| \text{com}_{3,t} \| c)$.
  (iii) L sends COM_1, COM_2, and COM_3 to the verifier V
(5) The verifier V sends $b \longleftarrow^{\$} \{0,1,2\}$ to L, who transfers it to all the $t-1$ signers
(6) The t signers
  (i) $b = 0$: each signer sends $\text{RSP}_{0,i} = (y_i, \delta_i)$ to L, then L sends $\text{RSP}_{0,1}, \cdots, \text{RSP}_{0,t}$ a, b to V.
  (ii) $b = 1$: each signer sends $\text{RSP}_{1,i} = (y_{i+} e_i, \delta_i)$ to L, then L sends $\text{RSP}_{1,1}, \cdots, \text{RSP}_{1,t}$ a, c to V.
  (iii) $b = 2$: each signer sends $\text{RSP}_{2,i} = (\delta_i(y_i), \delta_i(e_i))$ to L, then L sends $\text{RSP}_{2,1}, \cdots, \text{RSP}_{2,t}$, b, c to V.
(7) The verifier V
  (i) $b = 0$: the verifier uses $\text{RSP}_{0,i}$ to construct $\text{com}_{1,i}$, $\text{com}_{2,i}$, verifying that COM_1 is equal to $h(com_{1,1} \| \cdots \| com_{1,t} \| a)$ and that COM_2 is equal to $h(com_{2,1} \| \cdots \| com_{2,t} \| b)$.
  (ii) $b = 1$: the verifier uses $\text{RSP}_{1,i}$ to construct $\text{com}_{1,i}$, $\text{com}_{3,i}$, verifying that COM_1 is equal to $h(com_{1,1} \| \cdots \| com_{1,t} \| a)$ and that COM_3 is equal to $h(com_{3,1} \| \cdots \| com_{3,t} \| c)$.
  (iii) $b = 2$: the verifier uses $\text{RSP}_{2,i}$ to construct $\text{com}_{2,i}$, $\text{com}_{3,i}$, verifying that COM_2 is equal to $h(com_{2,1} \| \cdots \| com_{2,t} \| b)$, COM_3 is equal to $h(com_{3,1} \| \cdots \| com_{3,t} \| c)$ and $w(\delta_i(e_i)) = t$.

Algorithm 5: Proposed threshold GStern's protocol.

(3) When $b = 1$ and $b' = 2$, e_i can be extracted from δ_i and $\delta_i(e_i)$ by the simulator

HVZK: the proof is analogous to the proof of HVZK for Stern's protocol in [23]. When $b = 0$, the simulator easily reveals y_i and $y_i + e_i$. When $b = 1$, the simulator gets x_i, where $Hx_i^T = s_i^T$. When $b = 2$, the simulator obtains a vector with weight t.

4.2. Security Analysis of Linkable Threshold Ring Signature. In this section, we present the analysis concerning the two aspects: correctness analysis and security analysis of the proposed scheme. We prove that the scheme has the usual properties for a linkable ring signature scheme: existential unforgeability, anonymity, nonslanderability, and linkability.

4.2.1. Completeness. The completeness of our linkable threshold ring signature is easily verified. When taking an honest signature as input, the verification algorithm's output is always correct since the underlying protocol is complete.

4.2.2. Existential Unforgeability. Ref. [17] presents that existential unforgeability in the classical setting is a direct consequence of the fact that the signature is obtained by applying the Fiat-Shamir transform to a sigma protocol that is complete, special sound, and HVZK. The subprotocol called in the linkable threshold ring signature proposed in this paper, namely, Algorithm 5, has been proven to satisfy the above properties in Section 4.1, so our scheme satisfies the unforgeability.

Theorem 1. *Our scheme is existentially unforgeable due to the hardness of the GSD problem.*

4.2.3. Anonymity. An adversary will be unable to determine which subset of t signers cooperates to sign a message due to anonymity. Supposing that we are capable of overcoming the anonymity and create an algorithm that handles the GSD problem, in order to demonstrate the anonymity of our scheme, Ref. [17] presents a method that the error vector's $w/2$ nonnull coordinates must be known in advance, and the details are as follows. We are incapable of determining if the provided tuple (H, s, G, r, w) is a legitimate public key or is just random. With some of the secrets known, we may create another tuple (H, s', G, r', w). There are two situations here. When (H, s, G, r, w) is a GSD tuple, (H, s', G, r', w) is also a GSD tuple, but when (H, s, G, r, w) is a random tuple, it is likewise a random tuple. Because the GSD problem is computationally difficult, knowing $w/2$ locations has no effect on the security proof. To keep the same level of security, we just need to raise the settings.

Due to the fact that every signer only provides commitments and responses to L as in typical zero-knowledge Stern's scheme, it is not feasible for information to be leaked

(1) Parameters: $n, k, t \in N$ satisfying $n > k$, $H \longleftarrow^{\$} \{0,1\}^{(n-k)\times n}$
(2) KeyGen: for each prover P_i, where i denotes one of the all members in the ring, and there are N members in ring R.
  (i) randomly chooses $e_i \longleftarrow^{\$} \{0,1\}^n$ satisfying $w(e) = w$
  (ii) computes $s_i^T = He_i^T$
  (iii) Public information of P_i: H, s_i, w
  (iv) Private information of P_i: e_i satisfying $w(e) = w$ and $He_i^T = s_i^T$
(3) Sign: t signers in ring R sign the message M
  (i) Compute the matrix $G = g(I)$ and $Ge_i^T = r_i^T$, where $I = (\overline{pk}, \text{issue})$ $\overline{pk} = (pk_1, \cdots, pk_N)$ and g is a public hash function which maps $\overline{pk}$ to $G \in F_2^{(n-k)\times n}$.
  (ii) Apply the Fiat-Shamir transform to Algorithm 5 on input (H, G, s, r), where $s = (s_1, \cdots, s_N)$, $r = (r_1, \cdots, r_N)$.
    (a) Commitment Com are calculated according to Algorithm 5
    (b) Simulate the verifier's challenge b as $f(\text{Com}, M)$
    (c) Calculate the corresponding responses RSP according to Algorithm 5
    (d) Output the transcript $T = (\text{Com}, b, \text{RSP})$
  (iii) Output the pair of message and signature (M, σ) where $\sigma = (r, \text{Com}, \text{RSP})$, and $r = (r_{i1}, \cdots, r_{it})$, where $i1, \cdots, it$ denote the t signers.
(4) Verify: the verifier V
  (i) Computes $b = f(\text{Com}, M)$
  (ii) Computes $G = g(I)$
  (iii) According to Algorithm 5 and input (H, G, s, r) verifies whether $T = (\text{Com}, b, \text{RSP})$ is a valid transcript. If the signature is valid, this stage outputs "1," otherwise, outputs "0."
(5) Link: given two signatures (M, σ) and (M', σ') where $\sigma = (r, \text{Com}, RSP)$, $\sigma' = (r', \text{Com}', \text{RSP}')$, $r = (r_{i1}, \cdots, r_{it})$, $r' = (r_{j1}, \cdots, r_{it})$, where $i1, \cdots, it$ denote the t signers, $j1, \cdots, jt$ denote other t signers, and satisfying $\text{Ver}(M, \sigma) = 1$ and $\text{Ver}(M', \sigma') = 1$, the verifier:
  (i) Check if the intersection of the set r and r' is empty
  (ii) If the intersection is not empty, outputs "linked" and the same element from two sets. Thus, it shows how many dishonest provers there are. Otherwise, accepts it and outputs "nonlinked."

Algorithm 6: Linkable threshold ring signature scheme.

between signers throughout the protocol. The outputs of all the games do not include any information on b because of the underlying protocol's zero-knowledge feature as well as the GSD problem being challenging to solve. This means that the adversary's likelihood of accurately estimating b is 1/2; hence, there is little chance of adversary A winning the game.

Theorem 2. *Our threshold ring signature scheme is anonymous due to the hardness of the GSD problem.*

4.2.4. Linkability. We give the proof of public traceability from two cases:

(i) *Case 1.* Suppose that there is at least one $r_{\text{in}} = r_{jm}$ in the set of two distinct signers, where $n \in \{1, \cdots, t\}$, $m \in \{1, \cdots, t\}$. Therefore, we have $r_{\text{in}}^T = g(I)e_{\text{in}}^T = g(I)$ $e_{jm}^T = r_{jm}^T$. In this case, the output of the proposed scheme is linked

(ii) *Case 2.* Suppose that $r_{\text{in}} \neq r_{jm}$, for all $n \in \{1, \cdots, t\}$, $m \in \{1, \cdots, t\}$. Thus, we have $r_{\text{in}}^T = g(I)e_{\text{in}}^T$, $r_{jm}^T = g(I)$ e_{jm}^T, and $r_{\text{in}} \neq r_{jm}$. Therefore, the intersection of r and r' is empty. That means the output of this scheme is accepted and nonlinked

It is worth noting that the same signers can sign different issues without being linked together, due to the collision resistance of hash function g. Because the issue in I is different, so $G = g(I)$ is different. Hence, even though the same signers use the same private key e_i, they will have different r_i, so the intersection of the two signers remains empty in the link phase.

Theorem 3. *Our scheme is linkable due to the hardness of the GSD problem.*

4.2.5. Nonslanderability. Theorems 1 and 2 have been used by the authors in [16] to demonstrate that any linkable ring signature schemes that fulfill unforgeability and linkability also satisfy the nonslanderability condition.

Theorem 4. *Our scheme is nonslanderable under the SD and GSD assumptions.*

5. Experimental Result Analysis

In this section, we consider the efficiency of our scheme in four aspects: public key size, secret key size, signature length, and implementation efficiency. At last, we give the comparison of safety properties.

5.1. Key Cost Analysis

(1) *Public key size*: the public key in our scheme consists of $H \in \{0,1\}^{(n-k)\times n}$ and $s_i \in \{0,1\}^{(n-k)}$, where i denotes one of all the members in the ring. The public key size of our scheme is $n^2 - nk + N(n-k)$ bits. Hence, we achieve a size of public key in $O(N)$ complexity

(2) *Secret key size*: the secret key of each signer is a vector $e_i \in \{0,1\}^n$, and the bit length of e_i is n bits. The secret key size of our scheme is Nn bits

According to decoding attacks in [24], we set the parameters of our scheme under 80-bit, 128-bit, and 256-bit security as follows. For 80-bit security, we denote the scheme proposed in this paper with parameters $n = 1268$, $k = 951$, $w = 25$, and $p = 140$ as Scheme 1. For 128-bit security, we denote the scheme proposed in this paper with parameters $n = 2400$, $k = 2006$, $w = 58$, and $p = 220$ as Scheme 2. For 256-bit security, we denote the scheme proposed in this paper with parameters $n = 4150$, $k = 3307$, $w = 132$, and $p = 440$ as Scheme 3.

It can be seen from Table 1 that under the 80-bit security level, if there are 100 ring members, the length of the public key is 0.051 Mbytes. For fixed $N = 100$, with 80-bit security, the public key size in [4] is 4.23 kbytes, that in [5] is 400 kbytes, and that in [6] is 8.56 kbytes. The length of the public key in this paper is shorter than that of [5] and longer than that of the scheme in [4, 6]. However, their public key lengths all have the complexity of $O(N)$.

5.2. Signature Length Analysis. In our scheme, the proof σ determines the signature length, and its complexity is $O(t)$ in our scheme, where t is the threshold value. The signature size of our scheme specifically consists of the following three aspects: (i) the size of three hash values of commitments is 3λ bits, depending on the security parameter λ. (ii) The size of response is $2nt + 2\lambda$ bits in each situation when $b = 0$, $b = 1$, or $b = 2$. (iii) The size of the vector set $r = (r_{i1}, \cdots, r_{it})$ is $t(n-k)$ bits. The threshold GStern's protocol needs to be repeated p times due to the cheating probability (for instance, to achieve security level with 2^{128}, p needs to be equal to 220). In conclusion, the signature size is $3\lambda + t(n-k) + 2ntp + 2\lambda p$ bits in our scheme. Hence, we achieve a size of signature in $O(t)$ complexity.

From the above analysis, it can be found that the signature length of the proposed scheme in this paper is only related to the threshold value t and not to the number of ring members N. Therefore, compared with other variable-length signature schemes, our scheme can achieve fixed-length signatures.

Table 1: Public key size (Mbytes).

Signature scheme	N			
	100	1000	10000	100000
Scheme 1	0.051	0.085	0.425	3.826
Scheme 2	0.117	0.159	0.582	4.809
Scheme 3	0.427	0.517	1.421	10.466

Table 2: Signature length (Mbytes).

Signature scheme	t			
	2	10	50	100
Scheme 1	0.087	0.423	2.119	4.239
Scheme 2	0.258	1.259	6.303	12.598
Scheme 3	0.897	4.380	21.799	43.555

Table 3: Execution efficiency.

Scheme	Keygen time (s)	Sign time (s)	Verify time (s)	Security level
Scheme 1	0.031	28.421	0.425	2^{80}
Scheme 2	0.249	164.862	1.322	2^{128}
Scheme 3	0.265	996.611	6.842	2^{256}

The practical results presented in Table 2 shows clearly the signature length of our scheme. Considering a particular example with $N = 100$ and $t = 50$ under the 80-bit security level, the threshold ring signature length of the scheme in [4] is 2 Mbytes, the signature length of the scheme in [5] is 2.3 Mbytes, and the signature length of our scheme is 2.1 Mbytes. These three values are in close proximity to each other. However, the signature lengths in both [4, 5] grow linearly as the number of ring members increases, while the signature length in our scheme is fixed. For example, when $N = 2000$, the signature length in [4, 5] is obviously longer than our scheme, which is several times that of our scheme.

For fixed $N = 1000$ and $t = 100$ with 80-bit security, compared with the current best performing threshold ring signature scheme, the signature length of the scheme in [6] is 4.1 Mbytes and the signature length of this paper is 4.2 Mbytes, which are very close to each other, but for larger systems, i.e., with more members in the ring, the advantage of this scheme can be shown. As long as the threshold value is determined, the length of signature is fixed for any number of ring members.

Compared with the lattice-based linkable threshold ring signature [19], the signature length reaches 0.28 Mbytes at the 111-bit security level with $N = 800$ and $t = 2$, which is longer than the signature length of our scheme using the same parameters at a higher security level. As the ring membership N increases, the signature length of [19] reaches 1.75 Mbytes when N reaches 5000, which is much larger than our scheme. It can be seen that our scheme is especially

Table 4: Comparison of safety features.

Scheme	Hard problem	Resistant to quantum attacks	Linkable	Threshold
[4]	Minimum distance	√	×	√
[5]	q-ary SD problem	√	×	√
[6]	General decoding problem	√	×	√
[17]	GSD problem	√	√	×
[19]	Ideal lattice	√	√	√
Our scheme	GSD problem	√	√	√

suitable for large systems, and the larger the number of ring members, the more the advantages of this scheme can be demonstrated.

5.3. Implementation Efficiency. The following tables show the timings we have obtained for a Python implementation. We give the implementation of our scheme on an Intel(R) Xeon(R) Gold 6148 CPU @ 2.40 GHz, running Windows 10.

In the following case, the parity check matrices we used are random. The following table shows the keygen time, sign time, and verify time when the ring members are set to 100 and the number of actual provers is 2. The keygen time is consumed for the generation of the necessary public and private keys.

It is vital to note that the implementation given in this paper is a proof of concept. Additionally, the experimental results in this paper are supported by the fact that the exchange of information between the leader and the provers occurs within the on the same computer or even in the same executable file. In contrast, in real life, different provers will be located on different computers, transmitting information with the leader through the network before each other, and the performance of the computers varies; in such a heterogeneous scenario, there are problems such as communication delays, so the interaction process will be dominated by the slowest provers. In order to reduce the network delay, we can make our scheme more efficient through edge computing in the wireless network [25–28].

In Table 3, we give some timing for the proposed linkable threshold ring signature. The saving using other matrix types such as systematic form matrices is negligible compared to the gained signature.

5.4. Comparison of Safety Features. In this section, we compare our scheme with five schemes from [4–6, 17, 19] in terms of security properties. As can be seen from Table 4, all six schemes are code-based signature schemes that are resistant to quantum attacks. In terms of linkability, only the scheme in [17, 19] and our scheme have linkability properties. In terms of threshold, all schemes except [17] have threshold properties.

Both [19] and our scheme, as shown in Table 4, have both linkable and threshold features, as well as being resistant to quantum attacks. The three features listed above provide the variety of possibilities for signature application scenarios.

6. Conclusions

Linkable threshold ring signature schemes have a wide range of applications, such as free group voting. In this paper, we propose a new code-based linkable threshold ring signature scheme, which is well suited for large voting systems. Our scheme is constructed by using the noninteractive protocol obtained by applying the Fiat-Shamir transform to a variety of GStern's protocol. The signature size of our scheme is in $O(t)$, and the size of public key is $O(N)$. We also provide the existential unforgeability, anonymity, nonslanderability, and linkability of our scheme, and so, our scheme is secure in ROM due to the difficulty of the SD problem and GSD problem.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (Program No. 61902315), the Natural Science Basic Research Plan of Shaanxi Province of China (Program Nos. 2021JM-463 and 2022JM-353), the Scientific Research Program funded by the Education Department of Shaanxi Province (No. 22JK0560), and the Graduate Innovation Fund of Xi'an University of Posts and Telecommunications (CXJJLY202021).

References

[1] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *International conference on the theory and application of cryptology and information security*, pp. 552–565, Springer, Berlin, Heidelberg, 2001.

[2] D. Chaum, E. van Heyst, and Group Signatures, "Advances in cryptology —EUROCRYPT 1991," in *Lecture Notes in Computer Science*, D. W. Davies, Ed., vol. 547, pp. 257–265, Springer, Berlin, Heidelberg, 1991.

[3] E. Bresson, J. Stern, and M. Szydlo, "Threshold ring signatures and applications to ad-hoc groups," in *Annual International*

Cryptology Conference, pp. 465–480, Springer, Berlin, Heidelberg, 2002.

[4] C. A. Melchor, P. L. Cayrel, and P. Gaborit, "A new efficient threshold ring signature scheme based on coding theory," *IEEE Transactions on Information Theory*, vol. 57, no. 7, pp. 4833–4842, 2011.

[5] P. L. Cayrel, S. M. E. Yousfi Alaoui, and G. Hoffmann, "An improved threshold ring signature scheme based on error correcting code," in *International Workshop on the Arithmetic of Finite Fields*, pp. 45–63, Springer, Berlin, Heidelberg, 2012.

[6] H. Assidi, E. B. Ayebie, and E. M. Souidi, "An efficient code-based threshold ring signature scheme," *Journal of information security and applications*, vol. 45, pp. 52–60, 2019.

[7] L. Dallot and D. Vergnaud, "Provably secure code-based threshold ring signatures," in *IMA International Conference on Cryptography and Coding*, pp. 222–235, Springer, Berlin, Heidelberg, 2009.

[8] J. K. Liu, V. K. Wei, and D. S. Wong, "Linkable spontaneous anonymous group signature for ad hoc groups," in *Australasian Conference on Information Security and Privacy*, pp. 325–335, Springer, Berlin, Heidelberg, 2004.

[9] M. H. Au, S. S. M. Chow, and W. Susilo, *Short Linkable Ring Signatures Revisited*, European Public Key Infrastructure Workshop, Springer, Berlin, Heidelberg, 2006.

[10] J. K. Liu and D. S. Wong, "Linkable ring signatures: Security models and new schemes," in *International Conference on Computational Science and Its Applications*, pp. 614–623, Springer, Berlin, Heidelberg, 2005.

[11] J. K. Liu, M. H. Au, and W. Susilo, "Linkable ring signature with unconditional anonymity," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 1, pp. 157–165, 2013.

[12] P. P. Tsang and V. K. Wei, "Short linkable ring signatures for e-voting, e-cash and attestation," in *International Conference on Information Security Practice and Experience*, pp. 48–60, Springer, Berlin, Heidelberg, 2005.

[13] P. P. Tsang, V. K. Wei, and T. K. Chan, "Separable linkable threshold ring signatures," in *International Conference on Cryptology in India*, pp. 384–398, Springer, Berlin, Heidelberg, 2004.

[14] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Review*, vol. 41, no. 2, pp. 303–332, 1999.

[15] C. Baum, H. Lin, and S. Oechsner, "Towards practical lattice-based one-time linkable ring signatures," in *International Conference on Information and Communications Security*, pp. 303–322, Springer, Cham, 2018.

[16] W. A. Alberto Torres, R. Steinfeld, and A. Sakzad, "Post-quantum one-time linkable ring signature and application to ring confidential transactions in blockchain (Lattice RingCT v1.0)," in *Australasian Conference on Information Security and Privacy*, pp. 558–576, Springer, Cham, 2018.

[17] P. Branco and P. Mateus, "A code-based linkable ring signature scheme," in *International Conference on Provable Security*, pp. 203–219, Springer, Cham, 2018.

[18] T. H. Yuen, J. K. Liu, M. H. Au, W. Susilo, and J. Zhou, "Efficient linkable and/or threshold ring signature without random oracles," *The Computer Journal*, vol. 56, no. 4, pp. 407–421, 2013.

[19] L. S. Zhuang, J. Chen, and Q. Y. Wang, "Lattice-based linkable threshold ring signature in E-voting," *Journal of Cryptologic Research*, vol. 8, no. 3, pp. 402–416, 2021.

[20] P. Branco and P. Mateus, "A traceable ring signature scheme based on coding theory," in *International Conference on Post-Quantum Cryptography*, pp. 387–403, Springer, Cham, 2019.

[21] A. Bender, J. Katz, and R. Morselli, "Ring signatures: stronger definitions, and constructions without random oracles," in *Theory of Cryptography Conference*, pp. 60–79, Springer, Berlin, Heidelberg, 2006.

[22] A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems," in *Conference on the theory and application of cryptographic techniques*, pp. 186–194, Springer, Berlin, Heidelberg, 2006.

[23] J. Stern, "A new identification scheme based on syndrome decoding," in *Annual International Cryptology Conference*, pp. 3–21, Springer, Berlin, Heidelberg, 1993.

[24] A. Canteaut and F. Chabaud, "A new algorithm for finding minimum-weight words in a linear code: application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 367–378, 1998.

[25] J. Feng, W. Zhang, Q. Pei, J. Wu, and X. Lin, "Heterogeneous computation and resource allocation for wireless powered federated edge learning systems," *IEEE Transactions on Communications*, vol. 70, no. 5, pp. 3220–3233, 2022.

[26] J. Feng, L. Liu, Q. Pei, and K. Li, "Min-max cost optimization for efficient hierarchical federated learning in wireless edge networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 11, pp. 2687–2700, 2022.

[27] L. Liu, M. Zhao, M. Yu, M. A. Jan, D. Lan, and A. Taherkordi, "Mobility-aware multi-hop task offloading for autonomous driving in vehicular edge computing and networks," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–14, 2022.

[28] H. Han, L. Fang, W. Lu, W. Zhai, Y. Li, and J. Zhao, "A GCICA grant-free random access scheme for M2M communications in crowded massive MIMO systems," *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 6032–6046, 2022.