Hindawi
Security and Communication Networks
Volume 2023, Article ID 1794053, 12 pages
https://doi.org/10.1155/2023/1794053

Research Article

A Postquantum Linkable Ring Signature Scheme from Coding Theory

Xindong Liu [1,2] **and Li-Ping Wang** [1,2]

[1]*State Key Laboratory of Information Security, Institute of Information Engineering, CAS, Beijing, China*
[2]*School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China*

Correspondence should be addressed to Li-Ping Wang; wangliping@iie.ac.cn

Received 12 February 2023; Revised 19 March 2023; Accepted 30 November 2023; Published 26 December 2023

Academic Editor: Shadab Alam

Linkable ring signatures (LRSs) are ring signatures with the extended property that a verifier can detect whether two messages were signed by the same ring member. LRSs play an important role in many application scenarios such as cryptocurrency and confidential transactions. The first code-based LRS scheme was put forward in 2018. However, this scheme was pointed out to be insecure. In this paper, we put forward a code-based LRS scheme by constructing a new Stern-like interactive protocol and prove that it meets the security requirements of LRSs. We also give the specific parameters and the performance on the platform of our scheme.

1. Introduction

Ring signatures [1], a significant cryptographic primitive, enable a group user to sign a message on behalf of the group (called a ring) while protecting their privacy. Anonymity means that a verifier only can verify the correctness of the signature but cannot identify who is the actual signer in the ring. As an efficient privacy protection technology, ring signatures have been widely used in various scenarios such as e-voting [2], e-cash [3], and e-lottery [4, 5]. However, in many real-world applications, it is significant to not only protect the privacy of the signer but also require each signer to sign no more than once. For example, in an e-voting system, each person signs their ballot on behalf of all those eligible to vote, and each person is required to vote only once.

For more application scenarios, Liu et al. came up with the first linkable ring signature (LRS) scheme in 2004 [6]. LRSs are ring signatures with an extended property, where one can publicly verify whether two signatures were produced by the identical signer without knowing the identity of the signer. Compared to a ring signature scheme, an LRS scheme contains a tag generated by an issue and the signing keys of the entire ring user, where an issue can represent a vote or a business event. If a ring member outputs two signatures with an identical tag, two signatures will be linked. In a more restricted version of LRSs, a signer will be linked as soon as he signs twice. We call this kind of LRSs one-time LRSs. This property plays an important role in building cryptocurrencies, such as keeping the spender's anonymity and avoiding double-spending attacks [7], since a sum of money can be used only once by a consumer no matter in any deal.

LRSs have been extensively researched based on the number theory problem [6, 8–10]. A general construction of LRSs was presented by Franklin and Zhang [11]. For the sake of linkability, they add a pseudorandom function (PRF) evaluation of the signer's private key to any ring signature scheme and combine it with a zero-knowledge proof of the correct evaluation. In 2019, Wang et al. [12] put forward a general construction of one-time LRSs that adds one-time signatures to any ring signature scheme to achieve linkability. With the arrival of high-performance quantum computers, most classical asymmetric cryptography schemes will be broken since Shor [13] came up with a quantum algorithm to break the discrete logarithm problem and the factoring problem. Therefore, a number of quantum-safe

LRS schemes have been put forward in the past few years, such as lattice-based LRS schemes [7, 14–16], code-based LRS schemes [17–19], and isogeny-based LRS schemes [15].

Code-based cryptography has flourished as one of the important fields of postquantum cryptography in recent years. The first signature scheme from coding theory is the Courtois–Finiasz–Sendrier (CFS) scheme [20]. After that, code-based signatures make a great development [21–23]. In 2007, Zheng et al. put forward the first ring signature scheme from coding theory [24]. Later, many other variants related to ring signatures appeared like threshold ring signatures [25, 26], traceable ring signatures [27], and group signatures [28–30]. In 2018, Branco and Mateus put forward the first code-based LRS scheme [18]. However, Feng et al. [31] point out that this scheme is not safe since the Cramer–Damgard–Schoenmakers (CDS) framework [32] was used to build the OR relationship. In 2020, Ren et al. proposed another code-based LRS scheme [19]. However, we found that once two signatures are linked, the information of the signer's private key will be leaked since the output contains the information of the private key.

1.1. Our Contributions. For the purpose of not using the CDS framework, we design a new Stern-like interactive zero-knowledge (ZK) protocol, which is inspired by Ezerman et al. [33], as the building block of our LRS scheme. A prover can use this interactive ZK protocol to prove that he holds a small-weight solution to two instances of the syndrome decoding problem, which means the prover is a certified ring member with a unique label vector. Therefore, we employ the interactive ZK protocol to build our LRS scheme. Then, we prove that our LRS scheme is not only correct but also achieves the LRS security requirements. Finally, we analyzed the efficiency of the scheme and gave the running time on the platform.

It should be pointed out that we do not use the techniques mentioned in the introduction to construct our LRS scheme. The first technique [11] needs secure PRFs with a succinct zero-knowledge system. The syndrome-based PRF [34] is the only proposed one with a zero-knowledge argument system. Nevertheless, such an argument system will lead to a very inefficient construction since given an input of length l, the PRF will implement l times of matrix multiplications. The second technique, to achieve one-time linkability through a one-time signature scheme, also increases the overhead of a concrete scheme. Our construction follows that of Baum et al. [14].

We assume a ring of size $N = 2^l$, each member of which is labeled $i \in \{1, \ldots, N\}$. Let $(\mathbf{H}, \mathbf{T}, \mathbf{s}_i)$ represent the public keys of the user U_i and $\mathbf{e}_i$ represent the secret key with Hamming weight t, where $\mathbf{H} \in \mathbb{F}_2^{k\times n}$, $\mathbf{T} \in \mathbb{F}_2^{k\times n}$ denote two matrices and $\mathbf{s}_i \in \mathbb{F}_2^k$, $\mathbf{e}_i \in \mathbb{F}_2^n$ denote two row vectors. Let $\mathbf{S} = [\mathbf{s}_1^\top, \ldots, \mathbf{s}_N^\top] \in \mathbb{F}_2^{k\times N}$ denote a matrix of size $k \times N$ and $\mathbf{x}_i$ denote a vector of length N such that the i-th position is 1 and the rest of the positions are 0. Therefore, there is $\mathbf{S} \cdot \mathbf{x}_i^\top = \mathbf{s}_i^\top$ and the equation $\mathbf{H} \cdot \mathbf{e}_i^\top = \mathbf{s}_i^\top$ can be reformulated as

$$\mathbf{H} \cdot \mathbf{e}_i^\top = \mathbf{S} \cdot \mathbf{x}_i^\top. \tag{1}$$

Then, U_i employs the secret key to get a vector

$$\mathbf{T} \cdot \mathbf{e}_i^\top = \mathbf{r}^\top. \tag{2}$$

Next, we construct a Stern-like ZK scheme where a prover is able to make the verifier believe he holds a pair $(\mathbf{e}_i, \mathbf{x}_i)$ satisfying equations (1) and (2) with hidden index i. By repeating this protocol a lot of times to make the soundness error negligible and employing the Fiat–Shamir transform [35], we get a transcript v of the NIZK argument. The final form of our proposed LRS is $(\mathbf{r}, v)$. The size of our scheme, including the public key and the signature, is linearly related to N. However, when setting the practical parameters, our scheme achieves better performance than that of the scheme put forward in [36] which is the best performance syndrome-based ring signature scheme with logarithmic signature size, as long as N does not exceed 2^{16} [28].

We note that Ren et al. [19] attempt to build another code-based LRS scheme. Unfortunately, there is a lot of weakness in this construction. In detail, they use an insecure signature scheme to build their LRS scheme, which would result in the disclosure of the signer's private key. We make an analysis in Section 6.

1.2. Roadmap. The remaining articles are structured as follows. In Section 2, we introduce many preliminaries needed in our paper. In Section 3, we first propose an interactive zero-knowledge protocol and then construct our LRS scheme. We give the security proof and some security parameters for our LRS scheme in Section 4. In Section 5, we present the implementation results of the proposed LRS scheme. In Section 6, we analyze Ren et al.'s scheme. Finally, in Section 7, we draw the conclusion.

2. Preliminaries

2.1. Notations. Let λ and $\mathrm{negl}(\lambda)$ represent a security parameter and a negligible function in λ, respectively. The set $\{1, 2, \ldots, z\}$ is abbreviated as $[z]$. We denote with $\oplus$ the addition modulo 2. If not specified explicitly, the bold lowercase and uppercase letters represent row vectors and matrices, respectively. The Hamming metric of a vector $\mathbf{y}$ is represented by $w(\mathbf{y})$. The transpose of $\mathbf{x}$ is represented by $\mathbf{x}^\top$. Let $B(N, t)$ be the set of vectors $\mathbf{v} \in \mathbb{F}_2^N$ such that $w(\mathbf{v}) = t$. Define a function I2B from a positive integer to its binary representation, so the inverse of I2B is written as B2I. For a distribution $\mathcal{F}$, the notation $b \leftarrow \mathcal{F}$ means that b is sampled from the distribution $\mathcal{F}$. If X is a set, then $x \xleftarrow{\$} X$ denotes that x is randomly picked from X.

2.2. Linkable Ring Signatures. We now introduce the definition of the LRSs. To keep things simple, $(\mathrm{pk}_1, \ldots, \mathrm{pk}_N)$ is abbreviated as $\overline{\mathrm{pk}}$.

Definition 1. An LRS scheme contains four polynomial-time algorithms (**KeyGen**, **Sign**, **Ver**, **Link**) in which

(i) $(\mathrm{pk}, \mathrm{sk}) \leftarrow \mathbf{KeyGen}(1^\lambda)$: taking λ as input, it publishes a pair of public and private keys $(\mathrm{pk}, \mathrm{sk})$

(ii) $\sigma \leftarrow \mathbf{Sign}(\overline{\mathrm{pk}}, M, \mathrm{sk})$: taking the public keys $\overline{\mathrm{pk}}$, a message M, and a private key sk as input, it generates a signature σ

(iii) $b \leftarrow \mathbf{Ver}(\overline{\mathrm{pk}}, M, \sigma)$: taking the public keys $\overline{\mathrm{pk}}$, a message M, and a signature σ as input, it outputs b either 1 (accept) or 0 (reject)

(iv) $b \leftarrow \mathbf{Link}(\overline{\mathrm{pk}}, M_1, \sigma_1, M_2, \sigma_2)$: taking the public keys $\overline{\mathrm{pk}}$, two messages M_1, M_2, and two signatures σ_1, σ_2 such that $\mathbf{Ver}(\overline{\mathrm{pk}}, M_1, \sigma_1) = 1$ and $\mathbf{Ver}(\overline{\mathrm{pk}}, M_2, \sigma_2) = 1$ as input, it outputs $b = 1$ or $b = 0$, where 1 means that σ_1 and σ_2 are issued by the same signer

2.2.1. Correctness. An LRS scheme achieves correctness if for any $\lambda \in \mathbb{N}$, and every messages M_1, M_2, $(\mathrm{pk}, \mathrm{sk}) \leftarrow \mathbf{KeyGen}(1^\lambda)$, the following holds:

$$\Pr\begin{bmatrix} \mathbf{Ver}\left(\overline{\mathrm{pk}}, M_1, \sigma_1\right) = 1 \\ \mathbf{Ver}\left(\overline{\mathrm{pk}}, M_2, \sigma_2\right) = 1 \\ \mathbf{Link}\left(\overline{\mathrm{pk}}, M_1, \sigma_1, M_2, \sigma_2\right) = 1 \end{bmatrix} = 1 - \mathrm{negl}(\lambda), \quad (3)$$

where $\sigma_1 = \mathbf{Sign}(\overline{\mathrm{pk}}, M_1, sk_j)$ and $\sigma_2 = \mathbf{Sign}(\overline{\mathrm{pk}}, M_2, sk_j)$, $j \in [N]$.

We employ the security model of [6, 14], which contains the following four aspects: existential unforgeability, anonymity, nonframeability, and linkability. In order to build the games used in these security models, we define the following two oracles:

(i) $\mathbf{Sign}(\cdot)$: taking a query of the form (M, i) as input, it generates a signature $\sigma \leftarrow \mathbf{Sign}(M, \mathrm{sk}_i)$

(ii) $\mathbf{Co}(\cdot)$: taking a $\mathrm{pk}_i, i \in [N]$ as input, it outputs the corresponding sk_i

Let $\mathscr{A}$ stand for a probabilistic polynomial-time (PPT) adversary and $\mathscr{A}^{\mathcal{O}}$ represent that $\mathscr{A}$ queries the random oracle $\mathcal{O}$.

Definition 2 (existential unforgeability). An LRS scheme $\mathscr{LRS} = (\mathbf{KeyGen}, \mathbf{Sign}, \mathbf{Ver}, \mathbf{Link})$ is existential unforgeable, if the advantage of $\mathscr{A}$ is negligible in the following game:

(1) $(\mathrm{pk}_i, \mathrm{sk}_i) \leftarrow \mathbf{KeyGen}(1^\lambda)\ i = [N]$

(2) $(M, \sigma) \leftarrow \mathscr{A}^{\mathbf{Sign}(\cdot)}(\overline{\mathrm{pk}})$

Here, $\mathscr{A}$ cannot use M to query the $\mathbf{Sign}(\cdot)$.

The advantage of breaking existential unforgeability is denoted by the following equation:

$$\mathrm{Adv}_{\mathscr{A}}^{Eu}(\lambda) = \Pr[\mathbf{Ver}(\overline{\mathrm{pk}}, M, \sigma) = 1]. \quad (4)$$

Definition 3 (anonymity). An LRS scheme $\mathscr{LRS} = (\mathbf{KeyGen}, \mathbf{Sign}, \mathbf{Ver}, \mathbf{Link})$ is anonymous, if the advantage of $\mathscr{A}$ is negligible in the following game:

(1) $(\mathrm{pk}_i, \mathrm{sk}_i) \leftarrow \mathbf{KeyGen}(1^\lambda)\ i = 1, 2$

(2) $p \leftarrow \{0, 1\}$

(3) $\sigma \leftarrow \mathbf{Sign}(M, \mathrm{sk}_p)$

(4) $p' \leftarrow \mathscr{A}^{\mathbf{Sign}(\cdot, \mathrm{sk}_p)}$

Here, $\mathscr{A}$ cannot query the $\mathbf{Sign}(\cdot, \mathrm{sk}_1)$ and the $\mathbf{Sign}(\cdot, \mathrm{sk}_2)$.

The advantage of breaking anonymity is denoted by the following equation:

$$\mathrm{Adv}_{\mathscr{A}}^{\mathrm{Anon}}(\lambda) = \left|\Pr[p = p'] - \frac{1}{2}\right|. \quad (5)$$

Definition 4 (nonframeability). An LRS scheme $\mathscr{LRS} = (\mathbf{KeyGen}, \mathbf{Sign}, \mathbf{Ver}, \mathbf{Link})$ is nonframeable, if the advantage of $\mathscr{A}$ is negligible in the following game:

(1) $(\mathrm{pk}_i, \mathrm{sk}_i) \leftarrow \mathbf{KeyGen}(1^\lambda)\ i = [N]$

(2) $(\mathrm{pk}_1, M_1) \leftarrow \mathscr{A}^{\mathbf{Sign}(\cdot)}(\overline{\mathrm{pk}})$

(3) $\sigma_1 \leftarrow \mathbf{Sign}(M_1, \mathrm{sk}_1)$

(4) $(M_2, \sigma_2) \leftarrow \mathscr{A}^{\mathbf{Sign}(\cdot)}(\overline{\mathrm{pk}}, \mathrm{pk}_1, M_1, \sigma_1)$

(5) $b \leftarrow \mathbf{Link}(\overline{\mathrm{pk}}, M_1, \sigma_1, M_2, \sigma_2)$

Here, $\mathscr{A}$ cannot use the M_2 to query the $\mathbf{Sign}(\cdot, \mathrm{sk}_1)$.

The advantage of breaking nonframeability is denoted by the following equation:

$$\mathrm{Adv}_{\mathscr{A}}^{\mathrm{Frame}}(\lambda) = \Pr[b = 1]. \quad (6)$$

Definition 5 (linkability). An LRS scheme $\mathscr{LRS} = (\mathbf{KeyGen}, \mathbf{Sign}, \mathbf{Ver}, \mathbf{Link})$ is linkable, if the advantage of $\mathscr{A}$ is negligible in the following game:

(1) $(\mathrm{pk}_i, \mathrm{sk}_i) \leftarrow \mathbf{KeyGen}(1^\lambda)\ i = [N]$

(2) $(M_i, \sigma_i) \leftarrow \mathscr{A}^{\mathbf{Sign}(\cdot), \mathbf{Co}(\cdot)}(\overline{\mathrm{pk}}) i = [N + 1]$

The advantage of breaking linkability is denoted by the following equation:

$$\mathrm{Adv}_{\mathscr{A}}^{\mathrm{Link}}(\lambda) = \Pr\begin{bmatrix} \forall i \in [N+1]: \mathbf{Ver}\left(\overline{\mathrm{pk}}, M_i, \sigma_i\right) = 1 \\ \forall i, k \in [N+1], i \neq k: \mathbf{Link}\left(\overline{\mathrm{pk}}, M_i, \sigma_i, M_k, \sigma_k\right) = 0 \end{bmatrix}. \quad (7)$$

Remark 6. In this work, the proof of existential unforgeability is not given. As shown in [37], it is easy to get existential unforgeability from linkability and nonframeability.

2.3. Cryptography in Coding Theory. We now introduce a few hard problems needed in our paper.

Problem 7 (syndrome decoding (SD) problem). Given a parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{k\times n}$, a syndrome $\mathbf{s} \in \mathbb{F}_2^k$, and a positive integer w, the SD (n, k, w) problem is to search for a solution $\mathbf{e}$ satisfying $\mathbf{H}\cdot\mathbf{e}^\top = \mathbf{s}^\top$ and $\mathrm{w}(\mathbf{e}) \le w$.

Problem 8 (general syndrome decoding (GSD) problem). Given two parity-check matrices $\mathbf{H}, \mathbf{G} \in \mathbb{F}_2^{k\times n}$, two syndromes $\mathbf{s}, \mathbf{r} \in \mathbb{F}_2^k$, and a positive integer w, the GSD (n, k, w) problem is to search for a solution $\mathbf{e}$ satisfying $\mathbf{H}\cdot\mathbf{e}^\top = \mathbf{s}^\top$, $\mathbf{G}\cdot\mathbf{e}^\top = \mathbf{r}^\top$, and $\mathrm{w}(\mathbf{e}) \le w$.

Considering the matrix $\mathbf{E}^\top = [\mathbf{H}^\top \| \mathbf{G}^\top]$ and the vector $\mathbf{p} = [\mathbf{s}\|\mathbf{r}]$, we have $\mathbf{E}\cdot\mathbf{e}^\top = \mathbf{p}^\top$. Therefore, the GSD (n, k, w) problem is equivalent to the SD $(n, 2k, w)$ problem.

Problem 9 (codeword finding (CF) problem). Given a parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{k\times n}$ and a positive integer w, the CF (n, k, w) problem is to search for a solution $\mathbf{e}$ satisfying $\mathbf{H}\cdot\mathbf{e}^\top = \mathbf{0}^\top$ and $\mathrm{w}(\mathbf{e}) \le w$.

Remark 10 (see [21]). For a binary $[n, k]$ linear code, the easy range for the weight w of the SD problem and CF problem is $[(n-k)/2, (n+k)/2]$.

Definition 11 (Gilbert–Varshamov (GV) bound). For a binary $[n, k]$ linear code, the GV bound d_{GV} is denoted by the following equation:

$$d_{\mathrm{GV}} := \max\left\{w: \sum_{i=0}^{w-1}\binom{n}{i} \le 2^{n-k}\right\}. \tag{8}$$

Remark 12. The SD problem has a unique solution with overwhelming probability if w is less than d_{GV}.

Problem 13 (decisional syndrome decoding (DSD) problem [38]). Given a parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{k\times n}$ and a syndrome $\mathbf{s} = \mathbf{H}\cdot\mathbf{e}^\top$, where the weight of the vector $\mathbf{e}$ is at most w, the DSD (n, k, w) problem is to distinguish between a random vector $\mathbf{r}$ and the syndrome $\mathbf{s}$.

Lemma 14 (leftover hash lemma [39]). *Given a distribution $\mathcal{F}$ over $\mathbb{F}_2^n$ with min-entropy s, a matrix $\mathbf{G} \xleftarrow{\$} \mathbb{F}_2^{k\times n}$, and a vector $\mathbf{r} \leftarrow \mathcal{F}$, the statistical distance between the distribution of $(\mathbf{G}, \mathbf{G}\cdot\mathbf{r}^\top)$ and the corresponding uniform distribution is less than μ, where $\mu > 0$ and $k \le s - 2\log(1/\mu) - \mathcal{O}(1)$.*

2.4. Zero-Knowledge Proof and Stern Protocol. We will introduce the definition of the zero-knowledge argument systems in this section. We use the set of statements-witnesses $\mathcal{C} = \{(a, b) \in \mathbb{F}_2^* \times \mathbb{F}_2^*\}$ to denote an NP-relation. We first introduce the definition of an interactive zero-knowledge argument system as follows.

Definition 15 (zero-knowledge argument systems [40]). Let $(\mathcal{P}, \mathcal{V})$ represent an interactive algorithm between a prover and a verifier, and $\mathcal{C}$ denote an NP-relation. We say the $(\mathcal{P}, \mathcal{V})$ is a ZK argument for a relation $\mathcal{C}$, if the following three conditions hold:

(i) Completeness: if $(a, b) \in \mathcal{C}$, there is

$$\Pr[\langle \mathcal{P}(a, b), \mathcal{V}(a)\rangle = 1] = 1. \tag{9}$$

(ii) ϵ-Soundness: if $(a, b) \notin \mathcal{C}$, then for any PPT $\widehat{\mathcal{P}}$,

$$\Pr[\langle \widehat{\mathcal{P}}(a, b), \mathcal{V}(a)\rangle = 1] \le \eta, \tag{10}$$

where η is negligible.

(iii) Statistical zero-knowledge: if there exists a PPT simulator $\mathcal{S}(a)$ which can interact with any $\mathcal{V}(a)$ and produce a simulated transcript I_{sim}, we have the following equation:

$$I_{\mathrm{sim}} \approx I_{\mathrm{real}}, \tag{11}$$

where I_{real} denotes the transcript of a real interaction's transcript.

Let $\mathbf{Setup}(\lambda)_\Pi$ be a setup algorithm about the protocol Π with an input λ, then return the parameters pp. The zero-knowledge property and simulation-extractability of noninteractive protocol are presented as follows.

Definition 16 (noninteractive zero-knowledge). Let $\Pi = (\mathbf{Setup}_\Pi, \mathcal{P}, \mathcal{V})$ denote a noninteractive protocol. The protocol $\Pi = (\mathbf{Setup}_\Pi, \mathcal{P}, \mathcal{V})$ is zero-knowledge for a relation $\mathcal{C}$, if a pair of PPT simulators (S_1, S_2) are presented such that for any $\mathcal{A}$, there is

$$\left|\Pr\left[\mathcal{A}^{\mathcal{O}_1(\cdot)}(\mathrm{pp}) \longrightarrow 1: \mathrm{pp} \leftarrow \mathbf{Setup}(\lambda)\right] - \Pr\left[\mathcal{A}^{\mathcal{O}_2(\cdot)}(\mathrm{pp}) \longrightarrow 1: (\mathrm{pp}, \tau) \leftarrow S_1(\lambda)\right]\right| \le \mathrm{negl}(\lambda), \tag{12}$$

where $\mathcal{O}_1$ and $\mathcal{O}_2$ first check that if the input (a, b) is contained in $\mathscr{C}$, if so, $\mathcal{O}_1$ outputs $\pi \leftarrow \mathscr{P}(\mathrm{pp}, a, b)$, and $\mathcal{O}_2$ outputs $\pi \leftarrow S_2(\mathrm{pp}, a, \tau)$; otherwise, return $\perp$.

Definition 17 (simulation-extractability). We say that $\Pi = (\mathbf{Setup}_\Pi, \mathscr{P}, \mathscr{V})$ achieves simulation-extractable with regard to a pair of PPT simulators (S_1, S_2), if there exists a PPT extractor $\mathscr{E}$ such that for $\mathscr{A}$, there is

$$\left|\Pr\left[\mathscr{V}(\mathrm{pp}, a, \vartheta^*) \longrightarrow 1 : (\mathrm{pp}, \tau) \leftarrow S_1(\lambda), (a, \vartheta^*) \leftarrow \mathscr{A}^{\mathcal{O}(\mathrm{pp},\tau,\cdot)}(\mathrm{pp}), b \leftarrow \mathscr{E}^{\mathscr{A}}(a, \vartheta^*)\right]\right| \in \mathrm{negl}(\lambda), \tag{13}$$

where $\mathscr{C}(a, b) \neq 1$, $\mathbb{V}$ represents all proofs produced by S_2, ϑ^* is not contained in $\mathbb{V}$, and $\mathcal{O}(\mathrm{pp}, \tau, \cdot)$ takes the input a and outputs $S_2(\mathrm{pp}, \tau, a)$.

2.4.1. Stern Protocol. In 1996, Stern introduced a three-round zero-knowledge argument of knowledge (ZKAoK) for the SD problem in coding theory [41]. The system parameters are a public matrix $\mathbf{H} \in \mathbb{F}_2^{k\times n}$, a syndrome $\mathbf{s} \in \mathbb{F}_2^k$, and a weight w. According to the Stern protocol, anyone can verify whether the prover holds a solution $\mathbf{e} \in \mathbb{F}_2^n$ with Hamming metric w such that $\mathbf{H} \cdot \mathbf{e}^\top = \mathbf{s}^\top$. If we embed the statistically hiding commitment scheme into it, then we can get a soundness error of 2/3. We describe the Stern protocol in Algorithm 1.

3. Our Code-Based Linkable Ring Signature

We first put forward an interactive zero-knowledge protocol, in which a prover can prove that he is a ring member with a label vector $\mathbf{r}$ that is generated by his own private key. Based on this interactive zero-knowledge protocol, we construct our LRS scheme by Fiat–Shamir transform.

3.1. The Underlying Zero-Knowledge Protocol. In this section, our main result is to show a zero-knowledge argument system as the underlying protocol of our LRS scheme. First, we need to introduce a set of vectors and an important permutation, which are introduced in [28]. Let l denote an integer and $N = 2^l$. Then, there are the following:

(1) For $\mathbf{y} = (y_1, \ldots, y_N) \in \mathbb{F}_2^N$ and $k \in [N]$, let δ_k^N denote a vector of length N such that the k-th position is 1 and the rest of the positions are 0

(2) Given a vector $\mathbf{a} = (a_1, \ldots, a_l) \in \mathbb{F}_2^l$, we introduce the permutation $\phi_{\mathbf{a}}$: $\mathbb{F}_2^N \longrightarrow \mathbb{F}_2^N$: $\mathbf{y} = (y_1, \ldots, y_N) \mapsto \mathbf{y}' = (y_1', \ldots, y_N')$, where $y_j = y_\pi'$, $\pi = B2I(I2B(j) \oplus \mathbf{a})$ for each $j \in [N]$

For any $j \in [N]$ and any $\mathbf{a} \in \mathbb{F}_2^l$, there is

$$\mathbf{y} = \delta_j^N \Leftrightarrow \phi_{\mathbf{a}}(\mathbf{y}) = \delta_{B2I(I2B(j)\oplus\mathbf{a})}^N. \tag{14}$$

(3) For any vector $\mathbf{x}, \mathbf{y} \in \mathbb{F}_2^N$ and $\mathbf{a} \in \mathbb{F}_2^l$, we have $\phi_{\mathbf{a}}(\mathbf{x} + \mathbf{y}) = \phi_{\mathbf{a}}(\mathbf{x}) + \phi_{\mathbf{a}}(\mathbf{y})$

We build our interactive ZKAoK following the Stern framework and make a summarization as follows:

(i) The public keys include two matrices $\mathbf{H}$ and $\mathbf{T}$ and N syndromes $\{\mathbf{s}_1, \ldots, \mathbf{s}_N\}$

(ii) The secret is a vector $\mathbf{e}_j \in B(n, t)$ satisfying $\mathbf{H} \cdot \mathbf{e}_j^\top = \mathbf{s}_j^\top$ where the $\mathbf{s}_j \in \{\mathbf{s}_1, \ldots, \mathbf{s}_N\}$ with hidden index j

(iii) It is the prover's target to make others convince of the following relations:

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}_j^\top = \mathbf{s}_j^\top \wedge \mathbf{e}_j \in B(n, t), \\ \mathbf{T} \cdot \mathbf{e}_j^\top = \mathbf{r}^\top. \end{cases} \tag{15}$$

Let $\mathbf{S} = [\mathbf{s}_1^\top, \ldots, \mathbf{s}_N^\top]$. The relation $\mathbf{H} \cdot \mathbf{e}_j^\top = \mathbf{s}_j^\top$ is equivalent to $\mathbf{H} \cdot \mathbf{e}_j^\top \oplus \mathbf{S} \cdot \mathbf{x}^\top = \mathbf{0}^\top$, where $\mathbf{x}$ denotes δ_j^N. Then, the relation can be rewritten as follows:

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}_j^\top \oplus \mathbf{S} \cdot \mathbf{x}^\top = \mathbf{0}^\top \wedge \mathbf{e}_j \in B(n, t), \\ \mathbf{T} \cdot \mathbf{e}_j^\top = \mathbf{r}^\top. \end{cases} \tag{16}$$

We use COM to denote a collision-resistant hash function. Next, we showed the details of the underlying interactive protocol in Algorithm 2.

Lemmas 14 and 18 point out that Algorithm 2 has the statistically zero-knowledge property and the special soundness property.

Lemma 18. *The interactive protocol shown in Algorithm 2 is an argument with the statistical zero-knowledge property if the COM is a statistically hiding string commitment scheme.*

Proof. We employ a simulator $\mathcal{S}$ which can interact with the verifier $\widehat{\mathscr{V}}$ after giving the public input $(\mathbf{H}, \mathbf{S}, \mathbf{T}, \mathbf{r})$. First, the simulator $\mathcal{S}$ *picks a* $\widehat{\mathrm{Ch}} \in \{0, 1, 2\}$, and then depending on the value ch chosen by $\widehat{\mathscr{V}}$,$\mathcal{S}$ proceeds as follows. □

Case 19. $\widehat{\mathrm{Ch}} = 0$: $\mathcal{S}$ randomly selects the following objects:

$$\begin{cases} \mathbf{r}_1 \xleftarrow{\$} \mathbb{F}_2^n, \mathbf{r}_2 \xleftarrow{\$} \mathbb{F}_2^N, \delta \xleftarrow{\$} \mathbf{S}_n, \mathbf{a} \leftarrow \mathbb{F}_2^l, \mathbf{e}' \xleftarrow{\$} B(n, t), \\ \mathbf{x}' \xleftarrow{\$} B(N, 1), \epsilon_1, \epsilon_2, \epsilon_3 \xleftarrow{\$} \{0, 1\}^\lambda. \end{cases} \tag{17}$$

Then, $\mathcal{S}$ sets the CMT as (c_1', c_2', c_3'), in which

$$\begin{cases} c_1' = \mathrm{COM}(\delta, \mathbf{a}, \mathbf{H} \cdot \mathbf{r}_1^\top \oplus \mathbf{S} \cdot \mathbf{r}_2^\top, \mathbf{T} \cdot \mathbf{r}_1^\top, \epsilon_1), \\ c_2' = \mathrm{COM}(\delta(\mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{r}_2), \epsilon_2), \\ c_3' = \mathrm{COM}(\delta(\mathbf{e}' \oplus \mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{x}' \oplus \mathbf{r}_2), \epsilon_3). \end{cases} \tag{18}$$

When the challenge *ch* is received, $\mathcal{S}$ performs as follows:

(1) **Public parameters:** n, k, w.
(2) **Private key:** Samples $\mathbf{e} \in \mathbb{F}_2^n$ such that $w(\mathbf{e}) = w$.
(3) **Public key:** Samples $\mathbf{H} \in \mathbb{F}_2^{k\times n}$ and calculates $\mathbf{H} \cdot \mathbf{e}^\top = \mathbf{s}^\top$.
(4) **Prover** $\mathscr{P}$:
  (i) Samples $\mathbf{r} \xleftarrow{\$} \mathbb{F}_2^n$ and $\varphi \xleftarrow{\$} \mathbf{S}_n$.
  (ii) Calculates $c_1 = \mathrm{COM}(\varphi, \mathbf{H} \cdot \mathbf{r}^\top)$, $c_2 = \mathrm{COM}(\varphi(\mathbf{r}))$, $c_3 = \mathrm{COM}(\varphi(\mathbf{r} + \mathbf{e}))$ and sets $\mathrm{CMT} = (c_1, c_2, c_3)$.
  (iii) Sends CMT to $\mathscr{V}$.
(5) **Verifier** $\mathscr{V}$:
  (i) Samples the challenge $\mathrm{ch} \xleftarrow{\$} \{0, 1, 2\}$.
(6) **Prover** $\mathscr{P}$:
  (i) Case ch = 0: Sets $\mathrm{resp} \coloneqq \{\mathbf{r}, \varphi\}$.
  (ii) Case ch = 1: Sets $\mathrm{resp} \coloneqq \{\mathbf{r} + \mathbf{e}, \varphi\}$.
  (iii) Case ch = 2: Sets $\mathrm{resp} \coloneqq \{\varphi(\mathbf{r}), \varphi(\mathbf{e})\}$.
  (iv) Sends the response resp to $\mathscr{V}$.
(7) **Verifier** $\mathscr{V}$:
  (i) If ch = 0, checks if $\mathrm{COM}(\varphi, \mathbf{H} \cdot \mathbf{r}^\top) = c_1$ and $\mathrm{COM}(\varphi(\mathbf{r})) = c_2$ are true.
  (ii) If ch = 1, checks if $\mathrm{COM}(\varphi, \mathbf{H} \cdot (\mathbf{r} + \mathbf{e})^\top + \mathbf{s}^\top) = c_1$ and $\mathrm{COM}(\varphi(\mathbf{r} + \mathbf{e})) = c_3$ are true.
  (iii) If ch = 2, checks if $\mathrm{COM}(\varphi(\mathbf{r})) = c_2, \mathrm{COM}(\varphi(\mathbf{r}) + \varphi(\mathbf{e})) = c_3$ and $w(\varphi(\mathbf{e})) = w$ are true.

Algorithm 1: Stern protocol.

(1) **Public parameters:** $n, k, w, N (\log N = l)$.
(2) **Private key:** Samples $\mathbf{e} \in B(n, t)$ and $\mathbf{x} = \delta_j^N$.
(3) **Public key:** $\mathbf{H}, \mathbf{T}, \mathbf{S}, \mathbf{r}$, where $\mathbf{H} \cdot \mathbf{e}^\top \oplus \mathbf{S} \cdot \mathbf{x}^\top = \mathbf{0}, \mathbf{T} \cdot \mathbf{e}^\top = \mathbf{r}^\top$.
(4) **Prover** $\mathscr{P}$:
  (i) Samples the following uniformly random objects:
  $\mathbf{r}_1 \xleftarrow{\$} \mathbb{F}_2^n, \mathbf{r}_2 \xleftarrow{\$} \mathbb{F}_2^N, \delta \xleftarrow{\$} \mathbf{S}_n, \mathbf{a} \leftarrow \{0, 1\}^l, \epsilon_1, \epsilon_2, \epsilon_3 \xleftarrow{\$} \{0, 1\}^\lambda$.
  (ii) Sets $c_1 = \mathrm{COM}(\delta, \mathbf{a}, \mathbf{H} \cdot \mathbf{r}_1^\top \oplus \mathbf{S} \cdot \mathbf{r}_2^\top, \mathbf{T} \cdot \mathbf{r}_1^\top, \epsilon_1)$,
  $c_2 = \mathrm{COM}(\delta(\mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{r}_2), \epsilon_2)$,
  $c_3 = \mathrm{COM}(\delta(\mathbf{e} \oplus \mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{x} \oplus \mathbf{r}_2), \epsilon_3)$.
  (iii) Sets the CMT as (c_1, c_2, c_3).
(5) **Verifier** $\mathscr{V}$:
  (i) Samples the $\mathrm{ch} \xleftarrow{\$} \{0, 1, 2\}$.
(6) **Prover** $\mathscr{P}$:
  (i) If ch = 0, sets $\mathrm{resp} \coloneqq \{\mathbf{r}_1, \mathbf{r}_2, \delta, \mathbf{a}, \epsilon_1, \epsilon_2\}$.
  (ii) If ch = 1, sets $\mathrm{resp} \coloneqq \{\mathbf{e} \oplus \mathbf{r}_1, \mathbf{x} \oplus \mathbf{r}_2, \delta, \mathbf{a}, \epsilon_1, \epsilon_3\}$.
  (iii) If ch = 2, sets $\mathrm{resp} \coloneqq \{\delta(\mathbf{r}_1), \delta(\mathbf{e}), \phi_{\mathbf{a}}(\mathbf{r}_2), \phi_{\mathbf{a}}(\mathbf{x}), \epsilon_2, \epsilon_3\}$.
  (iv) Sends resp.
(7) **Verifier** $\mathscr{V}$:
  (i) If ch = 0, checks if $\mathrm{COM}(\delta, \mathbf{a}, \mathbf{H} \cdot \mathbf{r}_1^\top \oplus \mathbf{S} \cdot \mathbf{r}_2^\top, \mathbf{T} \cdot \mathbf{r}_1^\top, \epsilon_1) = c_1$ and $\mathrm{COM}(\delta(\mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{r}_2), \epsilon_2) = c_2$ are true.
  (ii) If ch = 1, checks if $\mathrm{COM}(\delta, \mathbf{a}, \mathbf{H} \cdot (\mathbf{r}_1 \oplus \mathbf{e})^\top \oplus \mathbf{S} \cdot (\mathbf{r}_2 \oplus \mathbf{x})^\top, \mathbf{T} \cdot (\mathbf{r}_1 \oplus \mathbf{e})^\top \oplus \mathbf{r}, \epsilon_1) = c_1$ and $\mathrm{COM}(\delta(\mathbf{e} \oplus \mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{x} \oplus \mathbf{r}_2), \epsilon_3) = c_3$ are true.
  (iii) If ch = 2, checks if $\mathrm{COM}(\delta(\mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{r}_2), \epsilon_2) = c_2, \mathrm{COM}(\delta(\mathbf{e}) \oplus \delta(\mathbf{r}_1), \phi_{\mathbf{a}}(\mathbf{x}) \oplus \phi_{\mathbf{a}}(\mathbf{r}_2)\epsilon_3) = c_3$ and $w(\delta(\mathbf{e})) = w, w(\phi_{\mathbf{a}}(\mathbf{x})) = 1$ are true.

Algorithm 2: The proposed underlying ZK protocol.

(i) If ch = 0, lets $\mathrm{resp} = (\mathbf{r}_1, \mathbf{r}_2, \delta, \mathbf{a}, \epsilon_1, \epsilon_2)$ and sends it to $\widehat{\mathscr{V}}$
(ii) If ch = 1, terminates the process and outputs $\perp$
(iii) If ch = 2, lets $resp = (\delta(\mathbf{r}_1), \delta(\mathbf{e}), \phi_{\mathbf{a}}(\mathbf{r}_2), \phi_{\mathbf{a}}(\mathbf{x}), \epsilon_2, \epsilon_3)$ and sends it to $\widehat{\mathscr{V}}$

Case 20. $\widehat{\mathrm{Ch}} = 1$: computes $\mathbf{e}' \in \mathbb{F}_2^n$ and $\mathbf{x}' \in \mathbb{F}_2^N$ such that

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}'^\top = \mathbf{S} \cdot \mathbf{x}'^\top, \\ \mathbf{T} \cdot \mathbf{e}'^\top = \mathbf{r}^\top, \end{cases} \tag{19}$$

and samples random objects as follows:

$$\begin{cases} \mathbf{r}_1 \xleftarrow{\$} \mathbb{F}_2^n, \mathbf{r}_2 \xleftarrow{\$} \mathbb{F}_2^N, \delta \xleftarrow{\$} \mathbf{S}_n, \\ \mathbf{a} \leftarrow \{0,1\}^l, \epsilon_1, \epsilon_2, \epsilon_3 \xleftarrow{\$} \{0,1\}^\lambda, \end{cases} \tag{20}$$

where $\mathbf{S}_n$ denotes the symmetric group of all permutations of n elements.

Then, $\mathcal{S}$ sets $\mathrm{CMT} = (c_1', c_2', c_3')$ as the same as equation (18).

When *ch* is received, $\mathcal{S}$ performs as follows:

(i) If ch = 0, lets $\mathrm{resp} = (\mathbf{r}_1, \mathbf{r}_2, \delta, \mathbf{a}, \epsilon_1, \epsilon_2)$ and sends it to $\widehat{\mathcal{V}}$

(ii) If ch = 1, lets $\mathrm{resp} = (\mathbf{r}_1 \oplus \mathbf{e}', \mathbf{r}_2 \oplus \mathbf{x}', \delta, \mathbf{a}, \epsilon_1, \epsilon_3)$ and sends it to $\widehat{\mathcal{V}}$

(iii) If ch = 2, terminates the process and outputs $\perp$

Case 21. $\widehat{\mathrm{Ch}} = 2$: samples random objects:

$$\begin{cases} \delta \xleftarrow{\$} \mathbf{S}_n, \mathbf{a} \leftarrow \mathbb{F}_2^l, \mathbf{e}' \xleftarrow{\$} B(n,t), \\ \mathbf{x}' \xleftarrow{\$} B(N,1), \epsilon_1, \epsilon_2, \epsilon_3 \xleftarrow{\$} \{0,1\}^\lambda, \end{cases} \tag{21}$$

computes $\mathbf{r}_1 \in \mathbb{F}_2^n$ and $\mathbf{r}_2 \in \mathbb{F}_2^N$ such that

$$\begin{cases} \mathbf{H} \cdot \mathbf{r}_1^\top \oplus \mathbf{S} \cdot \mathbf{r}_2^\top = \mathbf{H} \cdot \mathbf{e}'^\top \oplus^\top \mathbf{S} \cdot \mathbf{x}'^\top, \\ \mathbf{T} \cdot \mathbf{r}_1^\top = \mathbf{r}^\top \oplus \mathbf{T} \cdot \mathbf{e}'^\top. \end{cases} \tag{22}$$

Then, $\mathcal{S}$ sets the CMT as the same as equation (18).

When the challenge ch is received, $\mathcal{S}$ performs as follows:

(i) If ch = 0, terminates the process and outputs $\perp$

(ii) If ch = 1, lets $\mathrm{resp} = (\mathbf{r}_1 \oplus \mathbf{e}', \mathbf{r}_2 \oplus \mathbf{x}', \delta, \mathbf{a}, \epsilon_1, \epsilon_3)$ and sends it to $\widehat{\mathcal{V}}$

(iii) If ch = 2, lets $\mathrm{resp} = (\delta(\mathbf{r}_1), \delta(\mathbf{e}'), \phi_{\mathbf{a}}(\mathbf{r}_2), \phi_{\mathbf{a}}(\mathbf{x}'), \epsilon_2, \epsilon_3)$ and sends it to $\widehat{\mathcal{V}}$

Because ch is sampled from $\{0,1,2\}$, the probability of $\mathcal{S}$ terminating is 1/3. As the simulator $\mathcal{S}$ outputs a successful transcript, the distribution of its outputs is indistinguishable from the real interaction.

Lemma 22. *On the input of* $(\mathbf{H}, \mathbf{S}, \mathbf{T}, \mathbf{r})$*, there is an extractor* $\mathcal{E}$ *that can obtain a pair* $(\mathbf{e}, \mathbf{x})$ *from a CMT* $= (c_1, c_2, c_3)$ *and 3 valid* $(resp_1, resp_2, resp_3)$ *to all 3 possible pairs* (ch_1, ch_2, ch_3)*. The witness* $(\mathbf{e}, \mathbf{x})$ *satisfies the following equations:*

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}^\top \oplus \mathbf{S} \cdot \mathbf{x}^\top = \mathbf{0}^\top \wedge \mathbf{e} \in B(n,t), \\ \mathbf{T} \cdot \mathbf{e}^\top = \mathbf{r}^\top. \end{cases} \tag{23}$$

Proof. We can construct an efficient knowledge extractor $\mathcal{E}$. Suppose that we have three valid transcripts $(\mathrm{CMT}, \mathrm{ch}_1, \mathrm{resp}_1), (\mathrm{CMT}, \mathrm{ch}_2, \mathrm{resp}_2)$, and $(\mathrm{CMT}, \mathrm{ch}_3, \mathrm{resp}_3)$ of the proposed protocol, where $\mathrm{ch}_1 \neq \mathrm{ch}_2 \neq \mathrm{ch}_3$, where

$$\begin{cases} \mathrm{resp}_1 \coloneqq \{\mathbf{r}_1, \mathbf{r}_2, \delta^1, \mathbf{a}^1, \epsilon_1^1, \epsilon_2^1\}, \\ \mathrm{resp}_2 \coloneqq \{\mathbf{e} \oplus \mathbf{r}_1, \mathbf{x} \oplus \mathbf{r}_2, \delta^2, \mathbf{a}^2, \epsilon_1^2, \epsilon_3^2\}, \\ \mathrm{resp}_3 \coloneqq \{\delta(\mathbf{r}_1), \delta(\mathbf{e}), \phi_{\mathbf{a}}(\mathbf{r}_2), \phi_{\mathbf{a}}(\mathbf{x}), \epsilon_2^3, \epsilon_3^3\}. \end{cases} \tag{24}$$

Because of the collision-resistance property of COM, we have

$$\begin{cases} \mathbf{a}^1 = \mathbf{a}^2, \delta^1 = \delta^2, \epsilon_1^1 = \epsilon_1^2, \epsilon_2^1 = \epsilon_2^3, \epsilon_3^2 = \epsilon_3^3, \\ \delta(\mathbf{r}_1) = \delta^1(\mathbf{r}_1), \delta^2(\mathbf{e} \oplus \mathbf{r}_1) = \delta(\mathbf{r}_1) + \delta(\mathbf{e}), \\ \phi_{\mathbf{a}}(\mathbf{r}_2) = \phi_{\mathbf{a}^1}(\mathbf{r}_2), \phi_{\mathbf{a}^2}(\mathbf{r}_2 \oplus \mathbf{x}) = \phi_{\mathbf{a}}(\mathbf{r}_2) + \phi_{\mathbf{a}}(\mathbf{x}), \\ \mathbf{H}\mathbf{r}_1 + \mathbf{T}\mathbf{r}_2 = \mathbf{H}(\mathbf{r}_1 \oplus \mathbf{e}) + \mathbf{T}(\mathbf{r}_2 \oplus \mathbf{x}), \\ w(\mathbf{e}) = t, w(\phi_{\mathbf{a}}(\mathbf{x})) = 1. \end{cases} \tag{25}$$

Therefore, the knowledge extractor $\mathcal{E}$ can extract the witness $(\mathbf{e}, \mathbf{x})$ from $\mathbf{e} = \mathbf{r}_1 \oplus \mathbf{e} \oplus \mathbf{r}_1 =$, $\mathbf{x} = \mathbf{r}_2 \oplus \mathbf{x} \oplus \mathbf{r}_2$. □

3.2. Our Code-Based Linkable Ring Signature Protocol. Our LRS scheme is put forward in Algorithm 3. Roughly speaking, we construct an LRS by repeating the underlying ZKAoK protocol enough times so that the soundness error is negligible and applying the Fiat–Shamir transform.

Given two positive integers N and l such that $N = 2^l$, we choose a random matrix $\mathbf{H}$ and two collision-resistant hash functions as follows:

(i) h_1: $\mathbb{F}_2^* \longrightarrow \mathbb{F}_2^p$

(ii) h_2: $\mathbb{F}_2^* \longrightarrow \mathbb{F}_2^{2\lambda}$

Here, p stands for the repetition number of the underlying ZK protocol.

The function h_1 is used to generate the Fiat–Shamir transform, and h_2 is the one used in the underlying ZKAoK protocol.

During the key generation algorithm, the user $\mathcal{U}_i$ randomly chooses a vector $\mathbf{e}_i$ from the set $B(n,t)$ and sets $\mathbf{s}_i^\top = \mathbf{H} \cdot \mathbf{e}_i^\top$.

In the signing algorithm, the user $\mathcal{U}_i$ gets the underlying protocol with $(\mathbf{H}, \mathbf{S}, \mathbf{T}, \mathbf{r})$ in which $\mathbf{S} = [\mathbf{s}_1^\top, \ldots, \mathbf{s}_N^\top]$ and $\mathbf{r}^\top = \mathbf{T} \cdot \mathbf{e}_i^\top$. By repeating the protocol many times and the Fiat–Shamir transform, he gets a NIZKAoK Π. The form of the proposed signature is $(\mathbf{r}, v)$, where $v \leftarrow P(\mathrm{pp}, \mathrm{pk}, \mathrm{sk})$.

Due to the validity of the NIZKAoK, the algorithm Ver always outputs 1. If two signatures $(\mathbf{r}, v, M)$ and $(\mathbf{r}', v', M')$ were generated by the same user $\mathcal{U}_i$, then by $\mathbf{r}^\top = \mathbf{T} \cdot \mathbf{e}_i^\top = \mathbf{r}'^\top$, the algorithm Link always outputs 1.

4. Analysis of the Proposed Protocol

We analyze the proposed LRS scheme from the aspects of correctness and security.

Theorem 23. *Our LRS scheme achieves correctness with an overwhelming probability.*

Proof. To explain that the given LRS scheme is correct, we first prove that the algorithm Ver $(\overline{\mathrm{pk}}, M, \sigma)$ always returns

(1) **Public parameters:** $n, k, w \in \mathbb{N}, N(\log N = l)$.
$\mathbf{H} \xleftarrow{\$} \{0,1\}^{k\times n}, \mathbf{T} \xleftarrow{\$} \{0,1\}^{k\times n}$
(2) **KeyGen:** User $\mathcal{U}_i, i \in [N]$:
(i) Samples $\mathbf{e}_i \in B(n, w)$ as private key.
(ii) Calculates the public key $\mathbf{s}_i^\top = \mathbf{H} \cdot \mathbf{e}_i^\top$ and sets $\mathbf{S}$ as $[\mathbf{s}_1^\top, \ldots, \mathbf{s}_N^\top]$.
(3) **Sign:** $\mathcal{U}_i$ performs the following steps:
(i) Calculates $\mathbf{T} \cdot \mathbf{e}_i^\top = \mathbf{r}^\top$.
(ii) Repeats **Algorithm** 2 with the input $(\mathbf{H}, \mathbf{T}, \mathbf{S}, \mathbf{r})$ p times such that the soundness error is negligible and then obtains $(\mathrm{CMT}_1, \ldots, \mathrm{CMT}_p)$.
(iii) Sets $(\mathrm{ch}_1, \ldots, \mathrm{ch}_p)$ as $h_1(\mathrm{CMT}_1, \ldots, \mathrm{CMT}_p, M)$.
(iv) Sets the corresponding responses $(\mathrm{resp}_1, \ldots, \mathrm{resp}_p)$ according to the step 6 of **Algorithm** 2.
(v) Sets the transcript $v = (\mathrm{CMT}_1, \ldots, \mathrm{CMT}_p, \mathrm{resp}_1, \ldots, \mathrm{resp}_p)$.
(vi) Outputs the linkable ring signature $\sigma = (\mathbf{r}, v)$.
(4) **Ver:** Given a signature (M, σ), the verifier
(i) Calculates $(\mathrm{ch}_1', \ldots, \mathrm{ch}_p') = h_1(\mathrm{CMT}_1, \ldots, \mathrm{CMT}_p, M)$;
(ii) According to **Algorithm** 2 with the input $\mathbf{r}$ to check if $v = (\mathrm{CMT}_1, \ldots, \mathrm{CMT}_p, \mathrm{resp}_1, \ldots, \mathrm{resp}_p)$ is a valid transcript.
(iii) If the above condition holds, it returns 1; otherwise, it returns 0.
(5) **Link:** When two signatures (σ, σ') are received, the verifier parses $\sigma = (\mathbf{r}, v)$ and $\sigma' = (\mathbf{r}', v')$
(i) Checks if σ and σ' are correct.
(ii) Checks if $\mathbf{r} = \mathbf{r}'$.
(iii) If the above two conditions hold, it outputs 1; otherwise, it outputs 0.

Algorithm 3: Our LRS scheme.

1, if the signature σ was generated by a ring user $\mathcal{U}_i$ honestly running algorithm $\mathbf{Sign}(M, \mathrm{sk}_i)$, for any $i \in [N]$.

We observe that the algorithm **Sign** is equivalent to the signer repeating the underlying protocol enough times and employing the Fiat–Shamir heuristic to it. Because of the perfect completeness and validness of the underlying protocol, we have $\mathrm{Ver}(\overline{\mathrm{pk}}, M, \sigma) = 1$ for any pair of (M, σ) generated by an honest signer.

As for the validity of the algorithm Link, it is straightforward to observe that if σ and σ' were generated by the identical user $\mathcal{U}_i$, for any $i \in [N]$, where $\sigma = (\mathbf{r}, v)$ and $\sigma' = (\mathbf{r}', v')$, then by $\mathbf{r}^\top = \mathbf{T} \cdot \mathbf{e}_i^\top$ and $\mathbf{r}'^\top = \mathbf{T} \cdot \mathbf{e}_i^\top$, the algorithm Link $(\overline{\mathrm{pk}}, M, M', \sigma, \sigma')$ always outputs 1. □

Theorem 24. *Our LRS scheme satisfies anonymity in the random oracle model because of the ZK property of Algorithm 2 and the intractability of the DSD problem.*

Proof. We set the five hybrid games $\mathbf{G}_0, \mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3$, and $\mathbf{G}_4$ so as to explain that the proposed scheme satisfies anonymity. $\mathbf{G}_0$ and $\mathbf{G}_4$ represent the real anonymous game, in which $p = 0$ and $p = 1$, respectively. We will prove that the sequence of games is indistinguishable from any adversary $\mathscr{A}$. The advantage of $\mathscr{A}$ in $\mathbf{G}_i$ is represented by $\mathrm{Adv}^{\mathrm{anon}}_{\mathscr{A},\mathbf{G}_i}(\lambda)$.

$\mathbf{G}_0$: This is a real anonymous game where $p = 0$. In this game, the challenger $\mathscr{C}$ performs **KeyGen** to generate $\mathrm{pk} = (\mathbf{H}, \mathbf{T}, \mathbf{s}_1, \ldots, \mathbf{s}_N)$ and $\mathrm{sk} = (\mathbf{e}_1, \ldots, \mathbf{e}_N)$. The adversary $\mathscr{A}$ can ask queries to three oracles, which are $\mathbf{Sign}(\cdot, \mathrm{sk}_0)$, $\mathbf{Sign}(\cdot, \mathrm{sk}_1)$, and $\mathbf{Sign}(\cdot, \mathrm{sk}_p)$. $\mathscr{C}$ honestly responds to the inquiries from the above three oracles by employing the private key to execute the **Sign** algorithm.

$\mathbf{G}_1$: Compared with the previous game, we set the following changes: $\mathscr{C}$ runs the simulator (S_1, S_2) instead of faithfully executing the underlying protocol and does the following steps:

(1) Outputs the parameters by running S_1 instead of executing the setup algorithm of the NIZKAoK
(2) Uses the simulation proof to reply to the queries of $\mathscr{A}$ by running S_2, when the adversary $\mathscr{A}$ accesses $\mathbf{Sign}_{sk_p}$

Since the underlying protocol is zero-knowledge, there is

$$\mathrm{Adv}^{\mathrm{anon}}_{\mathscr{A},\mathbf{G}_0}(\lambda) \approx \mathrm{Adv}^{\mathrm{anon}}_{\mathscr{A},\mathbf{G}_1}(\lambda). \tag{26}$$

$\mathbf{G}_2$: Compared with $\mathbf{G}_1$, $\mathbf{G}_2$ has the following modifications: The challenger $\mathscr{C}$ establishes a table with an empty initial state. When $\mathscr{A}$ makes a query M_i to $\mathbf{Sign}_{sk_p}$, $\mathscr{C}$ first checks if there is a tuple $(M_i, \mathbf{r}_p^i)$ in the table, where $\mathbf{r}_p^i$ is included in $\sigma = (\mathbf{r}, Com, Resp)$. If it exists, then the challenger $\mathscr{C}$ produces the signature with the $\mathbf{r}_p^i$ by running the simulator S_2. If not, the $\mathbf{r}_p^i$ is set as $\mathbf{r}_p^i = \mathbf{r} \xleftarrow{\$} \mathbb{F}_2^k$. Then, $\mathscr{C}$ records the tuple $(M_i, \mathbf{r}_p^i)$ in the table and produces the signature with $\mathbf{r}_p^i$ by running the simulator S_2.

In $\mathbf{G}_2$, the vector $\mathbf{r}_p^i$ is randomly sampled from $\mathbb{F}_2^k$ instead of being evaluated by $\mathbf{T} \cdot \mathbf{e}_p^\top$. Due to the intractability of the DSD problem, we have

$$(\mathbf{r}, \mathbf{T}) \approx \left(\mathbf{T} \cdot \mathbf{e}_p^\top, \mathbf{T}\right). \tag{27}$$

Therefore, there is

$$\mathrm{Adv}^{\mathrm{anon}}_{\mathscr{A},\mathbf{G}_2}(\lambda) \approx \mathrm{Adv}^{\mathrm{anon}}_{\mathscr{A},\mathbf{G}_1}(\lambda). \tag{28}$$

$\mathbf{G}_3$: The modification between $\mathbf{G}_3$ and $\mathbf{G}_2$ is that the $\mathbf{r}^i_p$ is evaluated by sk_1 instead of being randomly sampled in $\mathbf{G}_2$. In addition to the above modification, the other steps of $\mathbf{G}_3$ are the same as those of $\mathbf{G}_2$

$\mathbf{G}_4$: This is a real anonymity game where $p = 1$.

The above proof shows that $\mathscr{A}$ cannot distinguish $\mathbf{G}_0$ from $\mathbf{G}_4$ with nonnegligible probability. Therefore, it is impossible for $\mathscr{A}$ to distinguish whether p is 1 or 0. □

Theorem 25. *Our LRS scheme satisfies linkability in the random oracle model based on the CF problem and the simulation-extractability of the underlying interactive protocol.*

Proof. Let $\mathscr{A}$ denote an adversary who breaks the linkability of the Algorithm 3. It implies that $\mathscr{A}$ is able to output $N + 1$ pairs of tuples (M_i, σ_i) satisfying the following:

(1) $\mathbf{Ver}(\overline{\mathrm{pk}}, M_i, \sigma_i) = 1\ i \in [N+1]$

(2) $\mathbf{Link}(\overline{\mathrm{pk}}, M_i, M_k, \sigma_i, \sigma_k) = 0\ i, k \in [N+1]$

We use I_{real} to denote the real interface, in which all the parameters are honestly produced, and I_{simo} to denote the simulator's interface, in which all parameters are produced by the simulator S_1. Because the underlying protocol is zero-knowledge (Lemma 18), $\mathscr{A}$ cannot tell the difference between the two interfaces. For I_{simo}, there is an extractor $\mathscr{E}$ which can obtain a pair $(\mathbf{e}_i, \mathbf{x}_i)$, where $\mathbf{x}_i = \boldsymbol{\delta}^N_i$, for every message-signature pair (M_i, σ_i) satisfying the following equation:

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}_i^\top \oplus \mathbf{S} \cdot \mathbf{x}_i^\top = \mathbf{0}^\top, \\ \mathbf{T} \cdot \mathbf{e}_i^\top = \mathbf{r}_i^\top. \end{cases} \tag{29}$$

Since there are only N public keys contained in the ring, there must exist $\boldsymbol{\delta}^N_i = \boldsymbol{\delta}^N_k, i \neq k$, so the following hold:

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}_i^\top = \mathbf{S} \cdot \boldsymbol{\delta}^N_i = \mathbf{s}_i^\top, \\ \mathbf{H} \cdot \mathbf{e}_k^\top = \mathbf{S} \cdot \boldsymbol{\delta}^N_k = \mathbf{s}_i^\top, \end{cases} \tag{30}$$

where $\mathbf{s}_i$ stands for the i-th column of $\mathbf{S}$.

Since we suppose $\mathscr{A}$ breaks the linkability of the scheme, there is $\mathbf{T} \cdot \mathbf{e}_i^\top \neq \mathbf{T} \cdot \mathbf{e}_k^\top$. Thus, we have $\mathbf{e}_i \neq \mathbf{e}_k$. Otherwise, it contradicts the previous assumption. Therefore, by equation (30), we have $\mathbf{H} \cdot (\mathbf{e}_i^\top \oplus \mathbf{e}_k^\top) = \mathbf{0}^\top$. It implies that a PPT $\mathscr{A}$ is able to break the CF problem which contradicts the intractability of the CF problem. □

Theorem 26. *Our LRS scheme satisfies nonframeability in the random oracle model based on the GSD problem and the simulation extractability of Algorithm 2.*

Proof. We set the following two hybrid games to explain that breaking the nonframeability property is as intractable as solving the GSD problem.

$\mathbf{G}_0$: This is a real nonframeability game regarding the challenge public key pk_π. In this game, $\mathscr{C}$ runs **KeyGen** to get $\mathrm{pk} = (\mathbf{H}, \mathbf{T}, \mathbf{s}_1, \ldots, \mathbf{s}_N)$ and $\mathrm{sk} = (\mathbf{e}_1, \ldots, \mathbf{e}_N)$. The adversary $\mathscr{A}$ submits queries to the given oracles $\mathbf{Sign}(\cdot)$ and $\mathbf{Co}(\cdot)$ with the limitation that he cannot query the corruption oracle $\mathbf{Co}(\cdot)$ with the public key pk_π.

$\mathbf{G}_1$: $\mathscr{C}$ employs the simulator of the NIZKAoK to output the parameters and selects $\mathbf{y}_i \xleftarrow{\$} \mathbb{Z}_2^k, i = [N]$. Then, $\mathscr{C}$ establishes an empty table. When receiving a query (j, M), $\mathscr{C}$ first checks if a pair $(j, M, \mathbf{r}_j)$ exists in the table. If it does, $\mathscr{C}$ employs the simulator S_2 to produce the signature with the vector $\mathbf{r}_j$. If not, $\mathbf{r}_j$ is set as $\mathbf{r}_j = \mathbf{r} \xleftarrow{\$} \mathbb{Z}_2^k$. $\mathscr{C}$ records the pair $(j, M, \mathbf{r}_j)$ in the table and produces the signature combining $\mathbf{r}_j$ by employing the simulator S_2. Since the underlying protocol has the zero-knowledge property, it is impossible for a PPT adversary to distinguish Game 1 from Game 0

Then, we show that it is impossible to successfully attack in $\mathbf{G}_1$. Assume $\mathscr{A}$ can forge a pair $(M, \sigma = (\mathbf{r}, v))$ such that

(1) $\mathbf{Ver}(\overline{\mathrm{pk}}, M, \sigma) = 1$

(2) $\mathbf{Link}(\overline{\mathrm{pk}}, M, \sigma, M^*, \sigma^*) = 1$, the signature σ^* returned by $\mathscr{C}$ on a signing query (π, M^*)

Since $(\sigma = (\mathbf{r}, v), \sigma' = (\mathbf{r}^*, v^*))$ can be linked, we have $\mathbf{r} = \mathbf{r}^*$. Since the underlying noninteractive protocol achieves the simulation-extractability, there is an extractor that puts v as input and returns the witness $w = (\mathbf{e}, \boldsymbol{\delta}^N_\pi)$ satisfying the following equation:

$$\begin{cases} \mathbf{H} \cdot \mathbf{e}^\top = \mathbf{y}_\pi^\top, \\ \mathbf{T} \cdot \mathbf{e}^\top = \mathbf{r}^{*\top}. \end{cases} \tag{31}$$

However, since the vector $\mathbf{y}_\pi$ is output by the simulator and the vector $\mathbf{r}^*$ is sampled from $\mathbb{Z}_2^k$ in $\mathbf{G}_1$, it means the adversary $\mathscr{A}$ can break the GSD problem. Therefore, this is a contradiction, which means that there is no way of breaking the nonframeability of our LRS scheme. □

5. Parameters and Key Size

We make an analysis of the key and signature size and give the parameters for our scheme under 80-bit security.

(1) Public key size: the public key is composed of $(\mathbf{H}, \mathbf{T}, \mathbf{S}) \in \mathbb{Z}_2^{k \times n} \times \mathbb{Z}_2^{k \times n} \times \mathbb{Z}_2^{k \times N}$ and its size is $k(2n + N)$ bits.

(2) Private key size: the private key $\mathbf{e}_i \in \mathbb{Z}_2^n$ is n bit-size.

(3) Signature size: the signature σ is composed of a transcript v and a vector $\mathbf{r}$. In detail, the signature size contains the following parts:

(i) The commitment CMT is 6λ bit-size.

(ii) If the challenge ch = 0 and ch = 1, the response resp is $n + N + \mathrm{nlog}n + l + 2\lambda$ bit-size; otherwise, the response $resp$ is $2(n + N) + 2\lambda$ bit-size. Thus, the size of response resp is no more than $n(\log n + 1) + N + \log N + 2\lambda$ bit-size.

Table 1: Performance of our LRS scheme.

Log N	PK size	Average signature size	Sign (s)	Ver (s)
4	411 KB	66 KB	0.040	0.024
8	428 KB	71 KB	0.040	0.024
12	710 KB	160 KB	0.053	0.031
16	5.20 MB	1576 KB	0.240	0.132
20	77.20 MB	24.2 MB	3.428	1.764

(1) **Public parameters:** $n, k, w \in \mathbb{N}$.
$\mathbf{A} \xleftarrow{\$} F_q^{k\times n}, \mathbf{B} \xleftarrow{\$} F_q^{k\times n}$
(2) **KeyGen:** User $\mathcal{U}_i, i \in [N]$:
(i) Samples $\mathbf{e}_i \in F_q^n$ and $w(\mathbf{e}_i) \le w$.
(ii) Calculates $\mathbf{s}_i^\top = \mathbf{A} \cdot \mathbf{e}_i^\top$.
(3) **Sign:** In order to sign the message m, $\mathcal{U}_\pi$:
(i) Calculates $\mathbf{r}_\pi^\top = \mathbf{B} \cdot \mathbf{e}_\pi^\top$.
(ii) Samples $\mathbf{u} \leftarrow F_q^n$ with $wt(\mathbf{u}) \le w$.
(iii) Sets $d_{\pi+1} = H(L, \mathbf{r}_\pi, m, \mathbf{A} \cdot \mathbf{u}, \mathbf{B} \cdot \mathbf{u})$
For $i = \pi + 1, \ldots, 1, \ldots, \pi - 1$.
(iv) Samples $\mathbf{z}_i \leftarrow F_q^n$ with $wt(\mathbf{z}_i) \le 2w$.
(v) Sets $d_{i+1} = H(L, \mathbf{r}_\pi, \mathbf{y}_{i,1}, \mathbf{y}_{i,2}, m,)$, in which:
$\mathbf{y}_{i,1}^\top = \mathbf{A} \cdot \mathbf{z}_i^\top - d_i \mathbf{s}_i^\top$,
$\mathbf{y}_{i,2}^\top = \mathbf{B} \cdot \mathbf{z}_i^\top - d_i \mathbf{r}_\pi^\top$.
(vi) Sets $\mathbf{z}_\pi = d_\pi \mathbf{e}_\pi + \mathbf{u}$.
(vii) Outs the signature $\sigma = (\mathbf{r}_\pi, d_1, \mathbf{z}_1, \ldots, \mathbf{z}_N)$.
(4) **Ver**: Given the signature σ, the verifier:
(1) For $i = [N]$, calculates
$\mathbf{y}_{i,1}^{'\top} = \mathbf{A} \cdot \mathbf{z}_i^\top - d_i \mathbf{s}_i^\top$,
$\mathbf{y}_{i,2}^{'\top} = \mathbf{B} \cdot \mathbf{z}_i^\top - d_i \mathbf{r}_\pi^\top$,
$d_{i+1}' = H(L, \mathbf{r}_\pi, \mathbf{y}_{i,1}', \mathbf{y}_{i,2}', m)$.
(2) If $d_1 = H(L, \mathbf{r}_\pi, \mathbf{y}_{N,1}', \mathbf{y}_{N,2}', m) = d_{N+1}$ and $\mathrm{wt}(\mathbf{z}_i) \le 2w$ for $i = [N]$, σ is accepted, otherwise rejected.

Algorithm 4: Ren et al.'s LRS scheme.

(iii) The vector $\mathbf{r}$ is k bit-size and the underlying protocol is repeated p times. Therefore, the signature size is no more than by $p(n(\log n + 1) + N + \log N + 2 + 8\lambda) + k$ bit-size.

We use seeds to replace permutations in the underlying protocol to reduce the signature size.

The parameters should satisfy not only that the SD problem and the GSD problem are hard but also that the distributions of $\mathbf{s}_i$ are indistinguishable from the uniform distribution, for any $i \in [N]$.

According to Lemma 14 and the information set decoding algorithm [42], the parameters (n, k, w) were chosen as (2800, 600, 132) so that the CF problem and the GSD problem are intractable under the 80-bit security, and the statistical distance of the distribution $\mathbf{s}_i$ between the uniform distribution is 2^{-80}, for any $i \in [N]$. The positive integer w is moderately larger than the GV bound but belongs to the difficult range of the CF problem and the GSD problem.

Remark 27. For the intractability of the GSD problem, k does not exceed $n/4$.

We perform our scheme on an Intel Core i7-8750H CPU@2.20 GHz and 4 GB of memory. For each set of parameters, the performance of our LRS scheme is presented in Table 1.

6. Analysis of Ren et al.'s Scheme

We first describe Ren et al.'s scheme [19] in Algorithm 4 and then analyze a serious weakness in their scheme. Let $\mathbf{A}, \mathbf{B} \leftarrow F_q^{(n-k)\times n}$ be two matrices and H: $H(\cdot) \longrightarrow F_q$ be a hash function. Their scheme is built as follows.

The above scheme is not safe because the attacker is able to obtain the private key from a large number of signatures.

Observed that the signature $\sigma = (d_1, \mathbf{z}_1, \ldots, \mathbf{z}_N, \mathbf{t}_\pi)$ signed by $\mathcal{U}_\pi$, except $\mathbf{z}_\pi$ is generated by the private key $\mathbf{e}_\pi$, other $\mathbf{z}_i$ are randomly generated during the signing process. So the distribution of $\mathbf{z}_\pi$, determined by $\mathbf{e}_\pi$, is different from the other distributions of $\mathbf{z}_i$, $i = [N], i \neq \pi$. If an adversary $\mathcal{A}$ can obtain multiple signatures that are linked to each other, he can perform statistical analysis on these signatures to obtain the identity of the signer. Therefore, $\mathcal{A}$ can obtain

$\sigma' = (d_\pi, \mathbf{z}_\pi, \mathbf{t}_\pi)$ by removing those $\mathbf{z}_i$ not generated by $\mathbf{e}_\pi$. In other words, $\mathcal{A}$ gets a lot of signatures generated by the same signer.

$\mathcal{A}$ can compute $\mathbf{z}'_\pi = d_\pi^{-1}\mathbf{u} + \mathbf{e}_\pi$. And the random vector $\mathbf{u}' = d_\pi^{-1}\mathbf{u}$ is of low Hamming weight, and hence, the support set of $\mathbf{u}'$ is small. Therefore, Persichetti [23] points out that a simple statistical analysis will recover the private key $\mathbf{e}_\pi$. The problem stems from the fact that the small weight vector $\mathbf{u}$ that is added to $d_\pi \mathbf{e}_\pi$ is not sufficient to cover up the support of the private key.

Our analysis suggests that this scheme not only fails to satisfy anonymity but also leaks the secret of the signer.

7. Conclusion

In this paper, we put forward a new LRS scheme, which achieves existential unforgeability, anonymity, nonframeability, and linkability since the intractability of the CF problem, the DSD problem, and the GSD problem. The key point of this work is to construct a Stern-like ZK protocol in which a prover can prove that he is a member of this ring with a certain tag generated by his private key. The sizes of public keys and signatures of our scheme are linearly related to the size of the ring N. We ran this scheme with different numbers of users on a computer and gave the corresponding results. Finally, we point out that Ren et al.'s scheme is insecure because an adversary can get the private key of the signer by doing a simple statistical analysis.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Natural Science Foundation of China under Grant no. 62372446 and the National Key Research and Development Program of China under Grant no. 2018YFA0704703.

References

[1] R. L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," in *Advances in Cryptology-ASIACRYPT 2001*, C. Boyd, Ed., vol. 2248, pp. 552–565, Springer, Berlin, Germany, 2001.

[2] F. Rodríguez-Henríquez, D. Ortiz-Arroyo, and C. García-Zamora, "Yet another improvement over the mu-varadharajan e-voting protocol," *Computer Standards & Interfaces*, vol. 29, no. 4, pp. 471–480, 2007.

[3] B. Libert, S. Ling, K. Nguyen, and H. Wang, *Zero-Knowledge Arguments for Lattice-Based Prfs and Applications to e-Cash*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2017.

[4] D. M. Goldschlag and S. G. Stubblebine, "Publicly verifiable lotteries: applications of delaying functions," in *Financial Cryptography, Second International Conference, FC'98*, R. Hirschfeld, Ed., vol. 1465, pp. 214–226, Springer, Berlin, Germany, 1998.

[5] E. Kushilevitz and T. Rabin, "Fair e-lotteries and e-casinos," in *Topics in Cryptology-CT-RSA 2001, the Cryptographer's Track at RSA 2001*, D. Naccache, Ed., vol. 2020, pp. 100–109, Springer, Berlin, Germany, 2001.

[6] J. K. Liu, V. K. Wei, and D. S. Wong, *Linkable Spontaneous Anonymous Group Signature For Ad Hoc Groups*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2004.

[7] W. A. A. Torres, R. Steinfeld, A. Sakzad et al., *Post-Quantum One-Time Linkable Ring Signature And Application To Ring Confidential Transactions In Blockchain (Lattice Ringct V1.0)*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2018.

[8] J. K. Liu and D. S. Wong, "Linkable ring signatures: security models and new schemes," in *Computational Science and its Applications - ICCSA 2005*, O. Gervasi, M. L. Gavrilova, V. Kumar et al., Eds., vol. 3481, pp. 614–623, Springer, Berlin, Germany, 2005.

[9] M. H. Au, S. S. M. Chow, W. Susilo, and P. P. Tsang, "Short linkable ring signatures revisited," in *Public Key Infrastructure, Third European PKI Workshop: Theory and Practice, EuroPKI 2006*, A. S. Atzeni and A. Lioy, Eds., vol. 4043, pp. 101–115, Springer, Berlin, Germany, 2006.

[10] J. K. Liu, M. H. Au, W. Susilo, and J. Zhou, "Linkable ring signature with unconditional anonymity," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 1, pp. 157–165, 2014.

[11] M. K. Franklin and H. Zhang, *A Framework for Unique Ring Signatures*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2012.

[12] X. Wang, Y. Chen, and X. Ma, "Adding linkability to ring signatures with one-time signatures," in *Information Security-22nd International Conference, ISC 2019*, Z. Lin, C. Papamanthou, and M. Polychronakis, Eds., vol. 11723, pp. 445–464, Springer, Berlin, Germany, 2019.

[13] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, 1997.

[14] C. Baum, H. Lin, and S. Oechsner, "Towards practical lattice-based one-time linkable ring signatures," in *Information and Communications Security-20th International Conference, ICICS 2018*, D. Naccache, S. Xu, S. Qing et al., Eds., vol. 11149, pp. 303–322, Springer, Berlin, Germany, 2018.

[15] W. Beullens, S. Katsumata, and F. Pintore, *Calamari and Falafl: Logarithmic (Linkable) Ring Signatures From Isogenies and Lattices*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2020.

[16] X. Lu, M. H. Au, and Z. Zhang, *Raptor: A Practical Lattice-Based (Linkable) Ring Signature*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2018.

[17] A. Barenghi, J. Biasse, T. Ngo, E. Persichetti, and P. Santini, "Advanced signature functionalities from the code equivalence problem," *International Journal of Computer Mathematics: Computer Systems Theory*, vol. 7, no. 2, pp. 112–128, 2022.

[18] P. Branco and P. Mateus, "A code-based linkable ring signature scheme," in *Provable Security-12th International Conference, ProvSec 2018*, J. Baek, W. Susilo, and J. Kim, Eds., vol. 11192, pp. 203–219, Springer, Berlin, Germany, 2018.

[19] Y. Ren, Q. Zhao, H. Guan, and Z. Lin, "On design of single-layer and multilayer code-based linkable ring signatures," *IEEE Access*, vol. 8, pp. 17854–17862, 2020.

[20] N. T. Courtois, M. Finiasz, and N. Sendrier, *How to Achieve A Mceliece-Based Digital Signature Scheme*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2001.

[21] N. Aragon, O. Blazy, P. Gaborit, A. Hauteville, and G. Zémor, *Durandal: A Rank Metric Based Signature Scheme*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2018.

[22] T. Debris-Alazard, N. Sendrier, and J. Tillich, "Wave: a new family of trapdoor one-way preimage sampleable functions based on codes," in *Advances in Cryptology-ASIACRYPT 2019*, S. D. Galbraith and S. Moriai, Eds., vol. 11921, pp. 21–51, Springer, Berlin, Germany, 2019.

[23] E. Persichetti, "Efficient one-time signatures from quasi-cyclic codes: a full treatment," *Cryptography*, vol. 2, no. 4, p. 30, 2018.

[24] D. Zheng, X. Li, and K. Chen, "Code-based ring signature scheme," *International Journal on Network Security*, vol. 5, no. 2, pp. 154–157, 2007.

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

[26] L. Dallot and D. Vergnaud, "Provably secure code-based threshold ring signatures," in *Cryptography and Coding, 12th IMA International Conference, Cryptography and Coding 2009*, M. G. Parker, Ed., vol. 5921, pp. 222–235, Springer, Berlin, Germany, 2009.

[27] P. Branco and P. Mateus, *A Traceable Ring Signature Scheme Based On Coding Theory*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2019.

[28] M. F. Ezerman, H. T. Lee, S. Ling, K. Nguyen, and H. Wang, "Provably secure group signature schemes from code-based assumptions," *IEEE Transactions on Information Theory*, vol. 66, no. 9, pp. 5754–5773, 2020.

[29] Q. Alamélou, O. Blazy, S. Cauchie, and P. Gaborit, "A code-based group signature scheme," *Designs, Codes and Cryptography*, vol. 82, no. 1-2, pp. 469–493, 2017.

[30] O. Blazy, P. Gaborit, and D. T. Mac, "A rank metric code-based group signature scheme," in *Code-Based Cryptography-9th International Workshop, CBCrypto 2021*, A. Wachter-Zeh, H. Bartz, and G. Liva, Eds., vol. 13150, pp. 1–21, Springer, Berlin, Germany, 2021.

[31] H. Feng, J. Liu, D. Li, Y. Li, and Q. Wu, "Traceable ring signatures: general framework and post-quantum security," *Designs, Codes and Cryptography*, vol. 89, no. 6, pp. 1111–1145, 2021.

[32] R. Cramer, I. Damgård, and B. Schoenmakers, "Proofs of partial knowledge and simplified design of witness hiding protocols," in *Advances in Cryptology - CRYPTO '94*, Y. Desmedt, Ed., vol. 839, pp. 174–187, Springer, Berlin, Germany, 1994.

[33] M. F. Ezerman, H. T. Lee, S. Ling, K. Nguyen, and H. Wang, *A Provably Secure Group Signature Scheme From Code-Based Assumptions*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2015.

[34] C. Brunetta, B. Liang, and A. Mitrokotsa, "Code-based zero knowledge PRF arguments," in *Information Security-22nd International Conference, ISC 2019*, Z. Lin, C. Papamanthou, and M. Polychronakis, Eds., vol. 11723, pp. 171–189, Springer, Berlin, Germany, 2019.

[35] A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems," in *Advances in Cryptology-CRYPTO '86*, A. M. Odlyzko, Ed., vol. 263, pp. 186–194, Springer, Berlin, Germany, 1986.

[36] K. Nguyen, H. Tang, H. Wang, and N. Zeng, *New Code-Based Privacy-Preserving Cryptographic Constructions*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2019.

[37] E. Fujisaki and K. Suzuki, *Traceable Ring Signature*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 2006.

[38] B. Applebaum, Y. Ishai, and E. Kushilevitz, "Cryptography with constant input locality," *Journal of Cryptology*, vol. 22, no. 4, pp. 429–469, 2009.

[39] S. Goldwasser, Y. T. Kalai, C. Peikert, and V. Vaikuntanathan, "Robustness of the learning with errors assumption," in *Innovations in Computer Science-ICS 2010*, A. C. Yao, Ed., pp. 230–240, Tsinghua University Press, Beijing, China, 2010.

[40] M. Bellare, M. Jakobsson, and M. Yung, *Round-Optimal Zero-Knowledge Arguments Based On Any One-Way Function*, IACR Cryptology ePrint Archive, Bellevue, WA, USA, 1997.

[41] J. Stern, "A new paradigm for public key identification," *IEEE Transactions on Information Theory*, vol. 42, no. 6, pp. 1757–1768, 1996.

[42] R. C. Torres and N. Sendrier, "Analysis of information set decoding for a sub-linear error weight," in *Post-quantum Cryptography-7th International Workshop, PQCrypto 2016*, T. Takagi, Ed., vol. 9606, pp. 144–161, Springer, Berlin, Germany, 2016.