Journal of Computer Science and Artificial Intelligence
ISSN: 3078-8242 (Print), ISSN: 3078-2309 (Online) | Vol. 2, No. 3, 2025

Lattice-Based Traceable Ring Signatures with Range Proofs

Zixin Sang

College of Software, Henan Polytechnic University, Jiaozuo 454000, China

Abstract: The ring signature ensures the anonymity of the signer and the unforgeability of the message during communication. However, the presence of sensitive data in the message may expose the signer's identity to some extent. To address this issue, we propose an efficient lattice-based ring signature scheme with range proofs. The scheme uses a commitment mechanism to ensure the signer conceals sensitive data while allowing the verifier to effectively validate the data's range. Additionally, a tracking algorithm is introduced to mitigate abusive behavior by revealing the identity of the abuser. Furthermore, the accumulator technique is applied to reduce both the size of the signature and the number of ring members, achieving a logarithmic relationship. Security analysis demonstrates that the scheme satisfies anonymity, unforgeability, and excludability. Efficiency analysis shows that the signature size grows logarithmically with the number of ring members, and that the communication overhead for range proof validation is minimal, enabling this functionality with very low communication cost. This scheme is applicable in scenarios requiring privacy protection and regulatory compliance, such as electronic voting, anonymous financial transactions, and digital identity management.

Keywords: Lattice-Based Cryptography, Range proof, Ring signature, Zero-knowledge proof.

1. Introduction

Ring signatures (RS) are digital signatures that can conceal the identity of the signer within a set of ring members, thereby guaranteeing the signer's anonymity. This concept was first proposed by Rivest et al. [1] in 2001. The verifier is able to ascertain that the signer is affiliated with one of the rings members, but is unable to identify the specific ring member. However, complete anonymity may lead to abuse by malicious users without consequence. For example, malicious users may exploit anonymity by casting multiple votes in an e-voting system without detection. In order to address this issue, Liu et al. [2] proposed the Linkable Ring Signature (LRS). This approach maintains the anonymity of the signer while determining whether two signatures have been generated by the same individual. Nevertheless, the LRS ensures the anonymity of the abuser while guaranteeing an honest signer, and thus is not an effective means of preventing abusive behavior. In 2007, Fujisaki et al. [3] proposed the Traceable Ring Signature (TRS), which guarantees the anonymity of an honest signer while simultaneously enabling the identification of a signer who abuses the anonymity. The TRS plays a pivotal role in information protection and privacy security, including in blockchain, the Internet of Things (IoT), and anonymization systems.

The advent of the Shor algorithm rendered ring signature algorithms in 1994, which rely on traditional hard problems, insecure against attacks by quantum computers. Consequently, as a type of post-quantum cryptography, the field of lattice-based cryptography is gaining significant attention in the context of post-quantum cryptography. In 1996, Ajtai et al [4] demonstrated that the worst-case and averagecase of lattices were equivalent, thereby establishing a theoretical foundation for investigating lattice-based ring signatures. Gentry et al. [5] constructed the first lattice-based signature scheme based on the SIS assumption, which is a provably secure "hash-and-sign" signature scheme. Subsequently, scholars have proposed and refined a variety of different types of ring signature schemes [6-7]. Since most schemes [8-11] have a ring structure and are only capable of linear operations, the signature size and efficiency of these schemes are usually proportional to the number of ring members, making them well suited for cases with a small number of members. Nevertheless, this can result in exceedingly large signature sizes when the number of members is considerable, which in turn gives rise to a considerable increase in the communication overhead, concurrently, the efficacy of the signature diminishes in a linear manner. Consequently, researchers have commenced investigations into the construction of ring signature schemes utilising lattice-based accumulators. In 2016, Libert et al. [12] designed an RS scheme in which the signature length is logarithmically related to the number of ring members, it is based on the Stern protocol [13] and a lattice-based accumulator. Feng et al. [14] designed a TRS scheme based the lattice-base accumulators with a new traceable algorithm and proved its security in 2020. In 2022, Nguyen et al. [15] constructed an URS scheme by improving the lattice-based accumulator and proved that it is secure under the random oracle model. In these schemes, the efficiency and soundness error of the zero-knowledge proof protocol directly determines the signature size for the efficiency of ring signatures. It is important to note that the majority of ring signature schemes based on zero-knowledge proofs are constructed based on the Stern protocol.

Although the Stern protocol is efficient, it has a soundness error of 2/3 in a single execution, necessitating multiple repetitions to reduce the error to a negligible level. This limitation makes it challenging to further enhance the efficiency of ring signatures. In 2019, Yang et al. [16] presented an efficient lattice-based zero-knowledge proof protocol that reduces the soundness error to the inverse of a polynomial. The protocol can be employed to prove the relationship between the parameters on the lattice. In 2023, Liang et al. [17-18] introduced an identity-based TRS scheme and a Certificateless TRS scheme by incorporating an identity cryptosystem. Existing ring signature schemes only guarantee

the anonymity of the signer. However, certain sensitive information in the message, such as transaction amount or timestamp, may inadvertently reveal additional details about the signer's identity. Therefore, it is crucial to allow the signer to conceal such information while ensuring the authenticity of the message can still be verified. Fig. 1 illustrates our ring signature model.

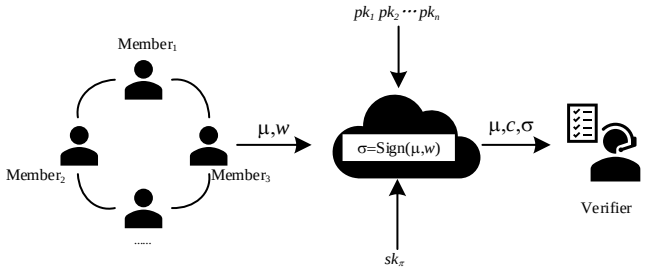

Figure 1. Our Ring Signature Model Diagram with Range Proofs

This paper presents a traceable ring signature scheme with range proofs. This scheme allows the signer to conceal sensitive data while enabling the verifier to validate the range of this data. It combines an efficient zero-knowledge proof protocol, as proposed in the literature, with a lattice-based accumulator. The scheme addresses the dual challenges of protecting sensitive information and preventing abuse of anonymity in signed messages. Our scheme incorporates a tracking algorithm to trace the identity of potential abusers while safeguarding sensitive data within a commitment value. Furthermore, the use of zero-knowledge proofs allows the verifier to validate the legitimacy of the data range without revealing the actual data, thus preserving the signer's privacy. Security analysis confirms that the scheme provides anonymity, traceability, and tamper resistance under the random oracle model. Efficiency analysis shows that the ring signature size and communication overhead scale logarithmically with the number of ring members. Therefore, our scheme is particularly suitable for scenarios with many ring members, as it efficiently handles the associated increase in signature size and communication overhead.

2. Preliminaries

2.1. Notations

The descriptions of the symbols in the scheme are given in Table 1.

Table 1. Symbols and definitions

Notations	Description
$\mathbb{Z}$	Set of integers
$\mathbb{Z}_q$	Set of integers modulo q
$\mathbb{Z}_q^m$	Set of matrices of m rows and m columns over $\mathbb{Z}_q$
$\mathbf{A}$	Matrix
$\mathfrak{R}$	Relationships between parameters in zero-knowledge proofs
$\mathcal{M}$	quadratic constraint of the witness
$\boldsymbol{R}$	Set of ring member public keys
$\boldsymbol{X}$	Privacy parameters in zero-knowledge proofs
$\boldsymbol{bin(b)}$	Decompose vector $\mathbf{b}$ into binary vectors
$\boldsymbol{r} \xleftarrow{\$} \mathbb{Z}_q^n$	Randomly select $\mathbf{r}$ from the distribution $\mathbb{Z}_q^n$
$\boldsymbol{A} \leftarrow \mathbb{Z}_q^{m\times n}$	select $\mathbf{A}$ from the distribution $\mathbb{Z}_q^{m\times n}$
$\{\mathbf{0},\mathbf{1}\}^* \rightarrow \mathbb{Z}_p^m$	Mapping strings $\{0,1\}^*$ into distribution $\mathbb{Z}_p^m$

2.2. Hardness Assumption

Definition 1 (SIS problem [19]). Given a random matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, find the vector $\boldsymbol{e} \in \mathbb{Z}^m$ such that $\boldsymbol{A}\cdot\boldsymbol{e} = 0\, mod\, q$ satisfying $0 < \| \boldsymbol{e} \|_\infty \leq \beta$, where $n, m, q \in \mathbb{N}$ and $\beta > 0$.

Definition 2 (Decision-LWE problem [20]). Randomly selecting the common matrix $\boldsymbol{B} \in \mathbb{Z}_q^{m\times n}$, given a vector $\boldsymbol{b} \in \mathbb{Z}_p^m$, the advantages of distinguishing which of the following two ways the vector might have been generated is negligible: $\boldsymbol{b} = \boldsymbol{B}\cdot\boldsymbol{s} + \boldsymbol{e}$ or $\boldsymbol{b} \xleftarrow{\$} \mathbb{Z}_q^m$, where $s \xleftarrow{\$} \{0,1\}^n$ and $\boldsymbol{e} \leftarrow \{0,1\}^m$.

Definition 3 (Decision-LWR problem [20]). Given a common matrix $\boldsymbol{T} \in \mathbb{Z}_q^{m\times n}$, and a vector $\boldsymbol{b} \in \mathbb{Z}_p^m$ generated in one of two ways: $\boldsymbol{b} \xleftarrow{\$} \lfloor \mathbb{Z}_q^m \rfloor_p$ or $\boldsymbol{b} = \lfloor \boldsymbol{T}\cdot\boldsymbol{s} \rfloor_p$, where $\boldsymbol{s} \xleftarrow{\$} \mathbb{Z}_q^n$, the probability of distinguishing which way it was generated is negligible. Here, for any $\boldsymbol{x} \in \mathbb{Z}_q^m$, we define $\lfloor \boldsymbol{x} \rfloor_p = \lfloor p/q \cdot \boldsymbol{x} \rfloor \in \mathbb{Z}_p^m$.

2.3. (Weak) Pseudorandom Function

Let n, p, q, m be postive integers that are polynomial in security parameter λ.

Definition 4 (PRF [21]). If the function $F : \mathcal{G} \times \mathcal{U} \rightarrow \mathcal{V}$ satisfies pseudo-randomness, then for any quantum adversary $\mathcal{A}$ in a polynomially bounded query, we have:

$$Pr[\mathcal{A}^{F_g}(1^\lambda) = 1, g \leftarrow Gen(1^\lambda)] - Pr[\mathcal{A}^{O}(1^\lambda) = 1, O \leftarrow \mathcal{F}[\mathcal{U}:\mathcal{V}]] \in negl(\lambda) \quad (1)$$

Where $\mathcal{F}[\mathcal{U}:\mathcal{V}]$ is the set of all functions with the domain $\mathcal{U}$ and range $\mathcal{V}$.

Lemma 1 (PRF in QROM [21]). Under the parameter settings of the lattice difficulty assumption, F^H is in QROM as a random function, satisfy $m > (n+1)\log q$ and $q \geq p\cdot\sqrt{m}\cdot U\cdot n^{\omega(1)}$, where $n, m, q, p \in \mathbb{N}^+$, χ denote a uniform bounded error distribution.

In our scheme, there are two properties:

(1) Uniqueness. For a uniformly selected message $u \leftarrow \mathcal{U}$, we have that:

$$\Pr\left[\exists g_1, g_2 \in \mathcal{G}, g_1 \neq g_2 \wedge F_{g_1}(u) = F_{g_2}(u)\right] \in \mathrm{negl}(\lambda) \quad (2)$$

(2) Intersection-Free Range. For any two different elements $v_1, v_2 \in \mathcal{V}$ and any polynomial $N(\cdot)$, we have that:

$$Pr[\exists c_1, c_2, d_1, d_2 \leq N(\lambda), c_1 v_1 + d_1 z_1 = c_2 v_2 + d_2 z_2 : v_1, v_2 \leftarrow \mathcal{V}] \leq negl(\lambda) \quad (3)$$

Where the range $\mathcal{V}$ is a vector space of the rational number field $\mathbb{Q}$.

Lemma 2: (Uniqueness [21]). Under the LWR assumption, a pseudo-random function F^H is uniqueness if $m \geq n\cdot(\log q + 1)/(\log p - 1)$.

2.4. Lattice-Base Accumulator

Let λ denote the security parameter, $N = 2^\ell$, $\ell \in \mathbb{N}^+$, denotes the number of leaf nodes in the accumulator, which are satisfied by all nodes: $\boldsymbol{t}_{i,j} \in \{0,1\}^m$, $k = \lceil log_2\, q \rceil$, the specific algorithm is as follows:

(1) ACC. Setup: Randomly selects a matrix $\boldsymbol{B} = (\boldsymbol{B}_1 | \boldsymbol{B}_2) \xleftarrow{\$} \mathbb{Z}_q^{n\times 2m}$, where $m = nk$, return the common parameter $para = \boldsymbol{B}$;

(2) ACC. Acc: For a given $R = \{d_i\}$, let $\boldsymbol{t}_{\ell,i} = d_i$, where $j \in [0, \ell-1]$, $i \in [0, 2^j - 1]$, compute $\boldsymbol{t}_{j,i} = bin(\boldsymbol{B}_1 \cdot$

$\boldsymbol{t}_{j+1,2i} + \boldsymbol{B}_2 \cdot \boldsymbol{t}_{j+1,2i+1}$) and return the root node value $\boldsymbol{t}_{0,0}$;

(3) ACC. Witness: Let $g(j) = 4\lfloor i^*/2^{\ell-(j-1)} \rfloor - \lfloor i^*/(2^{\ell-j}) \rfloor + 1$, $f(j) = \lfloor i^*/(2^{\ell-j}) \rfloor$, for a given $R = \{d_i\}$, node value $\boldsymbol{t}_{j,i}$, and element $\boldsymbol{d} = \boldsymbol{d}_{i^*}$, return $(bin(i^*), \{\boldsymbol{t}_{j,f(j)}\}, \{\boldsymbol{t}_{j,g(j)}\})$;

(4) ACC. Verify: Given a root node value $\boldsymbol{t}_{0,0}$, a leaf node $\boldsymbol{d}_i$ and the witness $(\iota, \{\boldsymbol{v}_i\}_{i\in[1,\ell]}, \{\boldsymbol{w}_i\}_{i\in[1,\ell]})$, return 1 if:

$$\begin{cases} \forall i \in [2,\ell], bin\left(\boldsymbol{B}_{1+\iota[i]} \cdot \boldsymbol{v}_i + \boldsymbol{B}_{2-\iota[i]} \cdot \boldsymbol{w}_i\right) = \boldsymbol{v}_{i-1}, \\ bin\left(\boldsymbol{B}_{1+\iota[1]} \cdot \boldsymbol{v}_1 + \boldsymbol{B}_{2-\iota[1]} \cdot \boldsymbol{w}_1\right) = \boldsymbol{t}. \end{cases} \quad (4)$$

2.5. Commitment

In our paper, secret values are hidden in promises so that ring members can self-certify by opening the commitment in case of signature ambiguity. We adopt the commitment form of Kawachi et al [22]. For the commitment values $m \in \{0,1\}^L$, randomly select the matrices $\boldsymbol{C}_1 \in \mathbb{Z}_q^{n\times m}$ and $\boldsymbol{C}_2 \in \mathbb{Z}_q^{n\times L}$, vectors $r \in \{0,1\}^m$, compute the commitments $c = \boldsymbol{C}_1 \cdot r + \boldsymbol{C}_2 \cdot w$, satisfying the following two properties:

(1) Hiddenness: The verifier cannot get any information about the secret value w through the promise c;

(2) Binding: The promisor cannot change the value w after it is committed.

3. Traceable Ring Signature with Range Proof

3.1. Definition

The following describes the definition of a traceable ring signature with range proof, in this case, the message μ consists of two parts: the range (T_1, T_2) of secret value w and the non-sensitive message:

(1) $pp \leftarrow \boldsymbol{TRS.Setup}\ (\lambda)$: Input security parameters λ, Output public system parameters pp;

(2) $sk_i \leftarrow \boldsymbol{TRS.Extract}\ (pp)$: . Input public system parameters pp, Output private and private key pairs (pk_i, sk_i);

(3) $\sigma \leftarrow \boldsymbol{TRS.Sign}\ (pp, I, R, \mu, w, sk_\pi)$: Input public system parameters pp, event I, a ring R, message μ, private key sk_π, and secret value w, output a ring signature σ;

(4) $\frac{0}{1} \leftarrow \boldsymbol{TRS.Verify}(pp, I, R, \mu, \sigma)$: Input system parameters pp, event I, a ring R, message μ, and a ring signature σ. If accepting the signature, return 1, else return 0.

(5) $\eta \leftarrow \boldsymbol{TRS.Trace}\ (pp, I, R, \mu, \sigma, \mu', \sigma')$: Input public system parameters pp, event I, a ring R, two valid tuples (μ, σ) and (μ', σ'), output $\eta \in \{accept, linked, pk\}$.

Correctness. A correct traceable ring signature scheme shall fulfill the following two conditions. The scheme has completeness if:

$$\Pr\left[\begin{matrix} 0 \leftarrow \boldsymbol{TRS.Verify}(pp, I, R, \mu, c, \sigma): \\ pp \leftarrow \boldsymbol{TRS.Setup}(\lambda), \\ (sk_i, pk_i) \leftarrow \boldsymbol{TRS.Extract}(pp), \\ \sigma \leftarrow \boldsymbol{TRS.Sign}(pp, I, R, \mu, w, sk_i) \end{matrix}\right] \leq negl(n) \quad (5)$$

A TRS scheme satisfies publicly traceable. If $pp \leftarrow \boldsymbol{TRS.Setup}(1^\lambda)$, $(sk_i, pk_i, sk_i', pk_i') \leftarrow \boldsymbol{TRS.Extract}(pp)$, $\sigma \leftarrow \boldsymbol{TRS.Sign}(pp, I, R, \mu, w, sk_i)$, $\sigma' \leftarrow \boldsymbol{TRS.Sign}(pp, I, R, \mu', w', sk_i')$, there is an overwhelming probability of satisfaction:

$$\eta \leftarrow \boldsymbol{TRS.Trace}(pp, I, R, \mu, \sigma, \mu', \sigma') = \begin{cases} accept, sk_i \neq sk_i' \\ linked, \mu = \mu' \ and \ sk_i = sk_i' \\ pk, \mu \neq \mu' \ and \ sk_i = sk_i' \end{cases} \quad (6)$$

3.2. Security Models

This section defines the security concepts and models that are employed in our scheme. These security properties are defined as follows:

Tag linkability ensures that under the same event, even if a PPT adversary $\mathcal{A}$ has access to the private keys of N ring members, $\mathcal{A}$ is not able to produce $N+1$ valid signatures that satisfy the unlinkability of any two signatures.

Definition 6 (Tag linkability). Define the probability that an adversary $\mathcal{A}$ wins the following games as an advantage for $\mathcal{A}$, i.e. $Adv_{\mathcal{A}}^{TaL} = \Pr[\mathcal{A}wins]$. If $Adv_{\mathcal{A}}^{TaL}(\lambda) \leq negl(\lambda)$, then our scheme satisfies tag linkability.

Anonymity requires that for an honest ring member, it is difficult to distinguish the signer in the ring in their generated signatures. In the anonymous game, the adversary $\mathcal{A}$ generates the public keys at random, except for the challenger who generates the public keys pk_0 and pk_1. Additionally, the adversary $\mathcal{A}$ has access to random oracle $\boldsymbol{Sign}_{\boldsymbol{sk_0}}$, $\boldsymbol{Sign}_{\boldsymbol{sk_1}}$ and $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$, where the sk_0 and sk_1 correspond to the public keys pk_0 and pk_1, $b \in \{0,1\}$, respectively, and the adversary's objective is to ascertain b.

Definition 7 (Anonymity). The advantage of the adversary $\mathcal{A}$ is defined as the probability of $\mathcal{A}$ winning the game, i.e. $Adv_{\mathcal{A}}^{anon} = \Pr''[b' = b] - 1/2$ .A TRS scheme satisfies anonymity if $Adv_{\mathcal{A}}^{anon}(\lambda) \leq negl(\lambda)$.

To prevent trivial attacks, restrictions are placed on the ability of adversary $\mathcal{A}$ to query the signature oracle:

(1) If $\mathcal{A}$ query (Γ, μ) in $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$, then Γ must contain the public key pk_0 and pk_1;

(2) If two tuples (Γ, μ) and $(\Gamma, \bar{\mu})$ are queried in $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$, then $\mu = \bar{\mu}$;

(3) If the tuple (Γ, μ) is queried for $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$ and $(\tilde{\Gamma}, \tilde{\mu})$ is queried by the adversary $\mathcal{A}$ in $\boldsymbol{Sign}_{\boldsymbol{sk_0}}$ or $\boldsymbol{Sign}_{\boldsymbol{sk_1}}$, then $\Gamma \neq \tilde{\Gamma}$.

Under the same tag and ring R, exculpability requires the same signer to sign different messages for anonymity to be lost. Addition an honest signer must ensure that an adversary $\mathcal{A}$ is unable to output two message signature pairs even after learning the signer's public key. This causes the tracking algorithm to output the signer's public key pk.

Definition 8 (Exculpability). The advantage of the adversary $\mathcal{A}$ is defined as the probability of $\mathcal{A}$ winning the game, i.e. $Adv_{\mathcal{A}}^{Excul} = Pr[\mathcal{A}wins]$, the TRS scheme satisfies Exculpability if $Adv_{\mathcal{A}}^{Excul}(\lambda) \leq negl(\lambda)$.

Adversary $\mathcal{A}$ must specify the object of the attack $pk*$, before the start of the game, They should then use that public key to generate two sets (T, μ, σ) and $(T, \mu*, \sigma*)$ of signatures, each consisting of at least one tuple N that cannot be linked to a tuple that the adversary $\mathcal{A}$ has queried in the oracle $\boldsymbol{Sign}_{\boldsymbol{sk}}$.

(1) Initialization: Executed before the game starts, the challenger $\mathcal{C}$ enters the security parameters λ to generate the public parameters pp and returns them to the adversary $\mathcal{A}$.

(2) Query: Except that $\mathcal{A}$ cannot query the private key corresponding to $pk*$, it can adaptively query the public-private key pair (sk, pk).

(3) Forgery: $\mathcal{A}$ forges two tuples $(\Gamma^*, \bar{\mu}, \bar{\sigma})$ and $(\Gamma^*, \mu', \sigma')$. $\mathcal{A}$ wins if the following conditions are satisfied: $1 \leftarrow \boldsymbol{TRS.Verify}(pp, \Gamma^*, \bar{\mu}, \bar{\sigma})$ and $1 \leftarrow \boldsymbol{TRS.Verify}(pp, \Gamma^*, \mu', \sigma')$, $\mathcal{A}$ forges at least one of $SQ(pp, pk^*, \Gamma^*, \bar{\mu})$ and $SQ(pp, pk^*, \Gamma^*, \mu')$; $pk^* \leftarrow \boldsymbol{TRS.Trace}(pp, \Gamma^*, \bar{\mu}, \bar{\sigma}, \mu', \sigma')$, where the advantage of $\mathcal{A}$ is expressed as $Adv_{\mathcal{A}}^{Excul} = Pr[\mathcal{A}wins]$.

Definition 9 (unforgeability [3]): A TRS scheme satisfies unforgeability if it satisfies tag linkability and exculpability.

4. Scheme Construction

4.1. Construction

In the scheme, $N = 2^{\ell}$ denotes the number of ring members, the ring R is the set of ring members' public keys, the tag of the event is denoted as I, $\Gamma = (I, R)$, and the scheme is constructed as follows.

(1) $pp \leftarrow Setup(\lambda)$

a. Choose the parameters $n = \mathcal{O}(\lambda)$, the distribution χ, the prime number q and a prime number p on the lattice to satisfy: $q \geq p \cdot B \cdot n^{\omega(1)}$, and the parameters β, $m > (n+1) \cdot (log\ q)$, $q_1 = \tilde{\mathcal{O}}(\lambda)$, $m'' = m \cdot \lceil log\ p \rceil$, $m' = n \cdot \lceil log\ q_1 \rceil$, $k' = \lceil log\ q \rceil - 1$, $l = \lceil L/k' \rceil$ under the difficult assumption $\mathrm{LWE}_{n,q,\chi}$, and set the common parameter $params = (m, m', m'', n, p, q, q_1)$;

b. Choose hash functions $H_1: \{0,1\}^* \rightarrow \mathbb{Z}_q^{m \times n}$, $H_2: \{0,1\}^* \rightarrow \mathbb{Z}_p^m$, where H_1 is modeled as a random oracle;

c. Choose the random matrix $\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times n}$, $\boldsymbol{D}' = [\boldsymbol{D}_1' | \boldsymbol{D}_2'] \leftarrow \mathbb{Z}_{q_1}^{n \times 2m'}$, $\boldsymbol{D} \leftarrow \mathbb{Z}_{q_1}^{n \times 2m''}$, $B_1 \leftarrow \mathbb{Z}_q^{m \times n}$, and output the common parameters: $pp = (params, H_1, H_2, H_3, \boldsymbol{A}, \boldsymbol{D}', \boldsymbol{D})$.

(2) $(sk, pk) \leftarrow Extract(pp)$

a. Input public parameters pp, randomly selected $\boldsymbol{s} \leftarrow \mathbb{Z}_q^n$, compute $\boldsymbol{y} = \boldsymbol{A} \cdot \boldsymbol{s}\ mod\ q$,

b. Output private key $sk = \boldsymbol{s}$, public key $pk = \boldsymbol{y}$.

(3) $\sigma \leftarrow Sign(pp, sk_{\pi}, \Gamma, \mu, w)$

a. Compute $\boldsymbol{B}_{\Gamma} = H_1(\Gamma) \in \mathbb{Z}_q^{m \times n}$;

b. Compute $\mathbf{b}_{\pi} = \lceil \boldsymbol{B}_{\Gamma} \cdot \boldsymbol{s}_{\pi} \rceil_p$, $\mathbf{b}_0 = H_2(\Gamma, \mu)$;

c. Compute $\alpha = \frac{\mathbf{b}_{\pi} - \mathbf{b}_0}{\pi} mod\ p$, for all $i \in [L]$, $i \neq \pi$, $\mathbf{b}_i = (\mathbf{b}_0 + \alpha \cdot i)\ mod\ p$;

d. For all $i \in [N]$, compute $\boldsymbol{d}_i = \mathrm{bin}(\mathbf{D}_1' \cdot bin(pk_i) + \boldsymbol{D}_2' \cdot bin(\mathbf{b}_i))$, define $R' = (\boldsymbol{d}_i)_{[N]}$;

e. For the input values w, randomly selected $\mathbf{r} \xleftarrow{\$} \mathbb{Z}_q^n$, $\mathbf{B}' \leftarrow \mathbb{Z}_q^{m \times L}$, where L is the number of bits after w conversion to binary, compute $\boldsymbol{c} = \boldsymbol{B}_{\Gamma} \cdot r + \mathbf{B}' \cdot (bin(w))^T$, for the ranges (T_1, T_2), compute $w - T_1 = U$, $T_2 - w = V$, compute $\bar{\boldsymbol{A}}_3 \cdot \bar{\boldsymbol{x}}_3 = \bar{y}_3$, where $\bar{y}_3 = \left(bin(T_1, T_2)\right)^T$, $\bar{\boldsymbol{A}}_3$ and $\bar{\boldsymbol{x}}_3$ are constructed in detail in Section 5;

f. Calculate $\mathbf{t} = \mathrm{ACC.Acc}(\mathbf{D}, R')$ on the accumulator and generate witness$w_A = \mathrm{ACC.Witness}(\mathbf{D}, R', t, \boldsymbol{d}_{\pi})$;

g. Set $X = (\boldsymbol{A}, \boldsymbol{B}_{\Gamma}, \boldsymbol{D}, \boldsymbol{D}', \boldsymbol{B}, \boldsymbol{t}, \boldsymbol{c}, T_1, T_2)$ to be the public parameter in the zero-knowledge proof and $W = (\boldsymbol{s}_{\pi}, pk_{\pi}, \boldsymbol{b}_{\pi}, \boldsymbol{d}_{\pi}, w_A, r, w)$ to be the privacy parameter in the zero-knowledge proof, and run the zero-knowledge proof to generate the corresponding proof δ.

h. Output Signature $\sigma = (\delta, \alpha)$.

(4) $0/1 \leftarrow \boldsymbol{Verify}(pp, I, R, \sigma, \mu)$

a. Compute $\boldsymbol{B}_{\Gamma} = H_1(\Gamma) \in \mathbb{Z}_q^{m \times n}$, $\mathbf{b}_0 = H_2(\Gamma, \mu)$;

b. Compute $\boldsymbol{b}_i = \boldsymbol{b}_0 + \alpha \cdot i$ where $i \in [N]$;

c. Compute $\boldsymbol{d}_i = \mathrm{bin}(\mathbf{D}_1' \cdot bin(pk_i) + \boldsymbol{D}_2' \cdot bin(b_i))$, $i \in [N]$;

d. Compute $\mathbf{t} = \mathrm{ACC.Acc}(\mathbf{D}, R')$;

e. For the public parameter $X = (\boldsymbol{A}, \boldsymbol{B}_{\Gamma}, \boldsymbol{D}, \boldsymbol{D}', \boldsymbol{B}, \boldsymbol{t}, \boldsymbol{c}, T_1, T_2)$, run the zero-knowledge proof protocol that $v \leftarrow Verify(X, \delta)$;

f. Returns True if $v = 1$, else returns false.

(5) $(linked/unlinked/pk_{\pi}) \leftarrow Trace(pp, \Gamma, \mu, \sigma, \mu', \sigma')$

a. Compute $\mathbf{b}_0 = H_2(\Gamma, \mu)$, $\boldsymbol{b}_i = \boldsymbol{b}_0 + \alpha \cdot i$, where $i \in [N]$;

b. Compute $\mathbf{b}_0' = H_2(\Gamma, \mu')$, $\boldsymbol{b}_i' = \boldsymbol{b}_0' + \alpha' \cdot i$, where $i \in [N]$;

c. Returns linked if for all $i \in [N]$, there is $\mathbf{b_i} = \mathbf{b_i'}$;

d. If there is only one $i \in [N]$, with $\mathbf{b_i} = \mathbf{b_i'}$, then return pk_i;

f. All other cases return unlinked.

4.2. Completeness

Correctness: In this TRS, α is calculated by computing the difference of two vectors and dividing by π. Since in the scheme, p is a large prime number and much larger than N, the elements in the resulting vector are always integers. The verifier can always compute the index value of the signer in the ring using the message, label and α. Given the completeness of NIZKAoK, the verifier is guaranteed to output a value of 1 for signatures generated by honest signers, i.e. $1 \leftarrow \boldsymbol{TRS.Verify}(pp, \Gamma, \sigma, \mu)$.

Public Traceability: In accordance with the established definition of public traceability, given two tuples (σ, μ) and (σ', μ'), the following three cases exist:

(1) TRS. Trace will output linked when $\mu \neq \mu'$ and $\pi = \pi'$, it means that the same signer signed the same message twice, $\boldsymbol{b}_0 = \boldsymbol{b}_0'$ and $\boldsymbol{b}_{\pi} = \boldsymbol{b}_{\pi'}'$;

(2) TRS. Trace will be outputpk_{π}, when $\mu \neq \mu'$and $\pi = \pi'$, it means that the same signer has signed different messages twice, we can get $\boldsymbol{b}_i \neq \boldsymbol{b}_i'$, $(i \neq \pi)$ and $\boldsymbol{b}_{\pi} = \boldsymbol{b}_{\pi'}'$, so we can get the index of the signer's public key, ;

(3) When $\pi \neq \pi'$, it means that a different signer has signed the message, TRS.Trace will output unlinked with overwhelming probability.

5. Scheme Construction

Our scheme employs the efficient zero-knowledge proof protocol proposed by Yang et al. [16] In this section, the relationships between the parameters in the scheme are constructed as those in the zero-knowledge proof through equation construction.

Given positive m, n, l, and a sufficiently large prime number q, an efficient protocol can demonstrate the relationship:

$$\Re = \{(\boldsymbol{A}, p, \mathcal{M}), s) \in (\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m \times ([1,n]^3)^l) \times (\mathbb{Z}_q^n): \boldsymbol{A} \cdot \boldsymbol{s} = p \wedge \forall (i,i,i) \in \mathcal{M}, s[i] = s[i] \cdot s[i]\} \quad (7)$$

Where the set $\mathcal{M}$ denotes the quadratic constraints over Witness $\boldsymbol{s}$.

The TRS scheme will be constructed using the ZKAoK described above. In this protocol, the proving party $\mathcal{P}$ sends a proof δ to the verifier $\mathcal{V}$. $X = (\boldsymbol{A}, \boldsymbol{B}_{\Gamma}, \boldsymbol{B}', \boldsymbol{D}_1, \boldsymbol{D}_2, \boldsymbol{B}, \boldsymbol{t}, \boldsymbol{c})$ is a set of publicly available parameters. The verifier $\mathcal{V}$ believes that the sender $\mathcal{P}$ possesses the proof $W = (\boldsymbol{s}_{\pi}, pk_{\pi}, \boldsymbol{b}_{\pi}, d_{\pi}, w_A, r, w)$ and satisfies the following relation after successfully checking the proof δ:

$$\Re_{TRS} = \{(\boldsymbol{A}, \boldsymbol{B}_\Gamma, \boldsymbol{B}', \boldsymbol{D}_1, \boldsymbol{D}_2, \boldsymbol{B}, \boldsymbol{t}, \boldsymbol{c}, T_1, T_2) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^{m\times L} \times \mathbb{Z}_q^{n\times m'''} \times \mathbb{Z}_q^{n\times m'} \times \mathbb{Z}_q^{n\times 2m''} \times \mathbb{Z}_q^{n} \times \mathbb{Z}_{\square}^{L} \times \mathbb{Z} \times \mathbb{Z}; (\boldsymbol{s}_\pi, pk_{id_\pi}, \boldsymbol{b}_\pi, r, w, d_\pi, w_A = (\boldsymbol{wit}, \{\boldsymbol{v}_i\}_{i\in[1,l]}, \{\boldsymbol{w}_i\}_{i\in[1,l]})) \in \mathbb{Z}_q^n \times \mathbb{Z}_q^m \times \mathbb{Z}_p^m \times \mathbb{Z}_q^n \times \{0,1\}^L \times \{0,1\}^{m''} \times (((0,1)^l) \times ((0,1)^{m''})^l \times ((0,1)^{m''})^l:$$

$$pk_\pi = \boldsymbol{A} \cdot \boldsymbol{s}_\pi mod q \tag{8}$$

$$\wedge\, \boldsymbol{b}_\pi = B_\Gamma \cdot \boldsymbol{s}_\pi \bmod p \tag{9}$$

$$\wedge\, d_\pi = bin(\boldsymbol{D}_1 \cdot pk_\pi + \boldsymbol{D}_2 \cdot \boldsymbol{b}_\pi) \tag{10}$$

$$\wedge\, \boldsymbol{c} = \boldsymbol{B}_\Gamma \cdot r + B' \cdot bin(w) \tag{11}$$

$$\wedge\, T_1 \le w \le T_2 \tag{12}$$

$$\wedge\, AVerify(\boldsymbol{t}, d_i, (\boldsymbol{wit}, \{\boldsymbol{v}_i\}_{i\in[1,l]}, \{\boldsymbol{w}_i\}_{i\in[1,l]})) \tag{13}$$

The equation (12) instantiation can be found in the literature [16] and will not be repeated here. Below is an example of how the relations can be translated into a zero-knowledge proof:

For $pk_\pi = \boldsymbol{A} \cdot \boldsymbol{s}_\pi$ in the first equation, since pk_π and $\boldsymbol{s}_\pi$ need to be hidden, the equation can be transformed into the following form:

$$\bar{\boldsymbol{A}}_1 \cdot \bar{\boldsymbol{x}}_1 = \bar{\boldsymbol{y}}_1 \bmod q \tag{14}$$

Where $\bar{\boldsymbol{A}}_1 = (\boldsymbol{A}| - \boldsymbol{I}_m)$, $\bar{\boldsymbol{x}}_1 = (\boldsymbol{s}_\pi^T \quad pk_\pi^T)^T$, $\bar{\boldsymbol{y}}_1 = \boldsymbol{0}$.

For the third equation $\boldsymbol{c}_\pi = bin(\boldsymbol{D}_1 \cdot pk_\pi + \boldsymbol{D}_2 \cdot \boldsymbol{b}_\pi)$, it is necessary to hide pk_π, $\boldsymbol{c}_\pi$ and $\boldsymbol{b}_\pi$, so the equation is transformed into the following form:

$$\boldsymbol{D}_1 \cdot pk_\pi + \boldsymbol{D}_2 \cdot \boldsymbol{b}_\pi - \boldsymbol{H}_n \cdot d_\pi = \boldsymbol{0} mod q \tag{15}$$

Let $\bar{\boldsymbol{A}}_2 = (\boldsymbol{D}_1|\boldsymbol{D}_2| - \boldsymbol{H}_n)$, $\bar{\boldsymbol{x}}_2 = \left(pk_\pi^T \; \mathrm{b}_\pi^T \; \boldsymbol{d}_\pi^T\right)^T$, $\bar{\boldsymbol{y}}_2 = \boldsymbol{0}$, it can be obtained:

$$\bar{\boldsymbol{A}}_2 \cdot \bar{\boldsymbol{x}}_2 = \bar{\boldsymbol{y}}_2 \bmod q \tag{16}$$

The equations (10) and (11), which are presented in Appendix G of the literature [16] and used here, are described objectively.

Let $\bar{\boldsymbol{A}}_3 = \begin{pmatrix} G & -G & \square & M_1 & \square \\ G & \square & G & \square & M_2 \end{pmatrix}$, $\bar{\boldsymbol{x}}_3 = \left(w^T \; u^T \; v^T \; {c_1'}^T \; {c_2'}^T\right)^T$, $\bar{y}_3 = \begin{pmatrix} bin(T_1) \\ bin(T_2) \end{pmatrix}$, where T_1 and T_2 are the upper and lower bounds of the range of secret values w and the vector $\bar{\boldsymbol{x}}_3$ is in binary form.

Next, the witness is decomposed into binary form as $\bar{\boldsymbol{x}}_b = bin(\bar{\boldsymbol{x}}_1)$, and $\bar{\boldsymbol{A}}_b = \bar{\boldsymbol{A}}_1 \cdot \boldsymbol{H}_{n+m}$, where $\boldsymbol{H}_{n+m} = \boldsymbol{I}_{n+m} \otimes (1\, 2 \cdots 2^{k_q-1})$.Then we have:

$$\boldsymbol{A}_{com} \cdot \boldsymbol{x}_{com} = \boldsymbol{y}_{com} \tag{17}$$

Where $\boldsymbol{A}_{com} = \begin{pmatrix} \bar{\boldsymbol{A}}_b & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \bar{\boldsymbol{A}}_2 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \bar{\boldsymbol{A}}_3 \end{pmatrix}$, $\boldsymbol{x}_{com} = (\bar{\boldsymbol{x}}_b^T \quad \bar{\boldsymbol{x}}_2^T \quad \bar{\boldsymbol{x}}_3^T)^T$, $\boldsymbol{y}_{com} = (\bar{\boldsymbol{y}}_1^T \quad \bar{\boldsymbol{y}}_2^T \quad \bar{\boldsymbol{y}}_3^T)^T$. Let $\mathcal{M} = \{(i,i,i)\}_{i\in[1,(m+n)k_q+mk_p+3L+2l]}, k_p = \lceil log\, p \rceil$.

6. Proof of Security

Lemma 4 (Tag Linkability). A trace ring signature scheme is tag linkable if the underlying NIZKoK has completeness, the SIS and LWR assumption are hard, and in the security parameter, satisfy $m \ge n \cdot (log\, q + 1)/(log\, p - 1)$.

Proof of Lemma 4: By proving indistinguishability between the following games, the advantage of a successful attack by PPT adversary $\mathcal{A}$ is negligible.

Initialization: Enter the security parameters1^λ, $\mathcal{C}$ gets the public parameters, and then hands over the public parameters pp to the adversary $\mathcal{A}$.

Query: Adversary $\mathcal{A}$ can execute the following polynomial time query for random oracle:

EQ (pp): Input public parameter pp, random oracle outputs public-private key pairs (sk_i, pk_i).Since adversary $\mathcal{A}$ can make unlimited queries, it can obtain all public-private key pairs. At the same time, $\mathcal{A}$ can generate valid signatures in the current environment.

Forgery: Adversary $\mathcal{A}$ forges $N+1$ valid signatures $\sigma_i = (\alpha_i, \delta_i)$ and $N+1$ tuples (μ_i, I, R), where $i \in [N+1]$, by calling $\boldsymbol{Sign}_{\boldsymbol{sk}}$ i.e.:

(1) $\boldsymbol{TRS.Verify}(pp, \Gamma, \mu_i, \sigma_i) = 1,\ \forall i \in [N+1]$

(2) $\boldsymbol{TRS.Trace}(pp, \Gamma, \mu_i, \sigma_i, \mu_j, \sigma_j) = accept\ ,\ \forall i, j \in [N+1],\ i \ne j$.

Analysis: The adversary $\mathcal{A}$ interacts with the challenger $\mathcal{C}$ by interacting with the challenger $\mathcal{C}$ who honestly generates the public parameters according to the scheme, and then uses the simulator's algorithm $\mathcal{S}_k$ is empioyed to replace the signature algorithm in the scheme. By virtue of the zero-knowledge nature of ZKAoK, the adversary $\mathcal{A}$ cannot distinguish this substitution. The corresponding proof is then generated $(\mu_i, \sigma_i = (\alpha_i, \delta_i))$ by $\boldsymbol{Sign}_{\boldsymbol{sk}}$, which can compute the sequence $(\boldsymbol{b}_1^{(i)}, \cdots, \boldsymbol{b}_N^{(i)})$ from the signature, where the witness is $w = (\pi_i, sk_{\pi_i})$, satisfying the following conditions: $pk_i = \boldsymbol{A} \cdot sk_{\pi_i} \wedge \boldsymbol{b}_{\pi_i}^{(i)} = \left\lfloor \boldsymbol{A}_\Gamma \cdot sk_{\pi_i} \right\rfloor_p, \pi_i \in [N]$.

Observing the above equation, under the SIS assumption, pk_i uniquely corresponds with overwhelming probability to sk_i. Under the same tag Γ, F uniquely corresponds to the key sk_i due to the uniqueness of $\boldsymbol{b}_{\pi_i}^{(i)}$. Therefore, pk_i uniquely corresponds with overwhelming probability to $\boldsymbol{b}_{\pi_i}^{(i)}$.

Since condition b) holds, it holds for $\boldsymbol{b}_{\pi_i}^{(i)} \ne \boldsymbol{b}_{\pi_j}^{(j)}$ corresponding to any index i and j in as $N+1$ sequence $\boldsymbol{b}_i^{(i)}$. Since PRFs are uniquely correspondent, the public and private keys are also uniquely correspondent, it is required that there are $N+1$ public keys satisfying $pk_i \ne pk_j,\ i, j \in [N+1]$. But there are only N public keys in Γ, which contradicts the condition, so the probability that the generated $N+1$ signatures satisfy conditions 1 and 2 is negligible, i.e., $Adv_{\mathcal{A}}^{TaL}(\lambda) \le negl(\lambda)$.

Lemma 5 (Anonymity). If the LWR assumption is hard, and the underlying zero-knowledge proof is zero-knowledge, the parameters satisfy $m \ge n\, log\, q + log\, q$, $q \ge p \cdot \sqrt{m} \cdot U \cdot n^{\omega(1)}$, TRS scheme satisfies anonymity.

Proof of Lemma 5. By proving indistinguishability between the following games, the advantage of a successful attack by PPT adversary $\mathcal{A}$ is negligible. Define the adversary $\mathcal{A}$ in the Game i is $Adv_{\mathcal{A}Gamei}^{Anon}(1^\lambda)$.

Game0: Same as Definition 7, where $b = 0$.The adversary $\mathcal{A}$ can invoke the key and the signature query random oracle. The challenger $\mathcal{C}$ takes the honest query public-private key pair $sk_i, pk_i \leftarrow \boldsymbol{TRS.Extract}(pp)$ and signature $\sigma \leftarrow \boldsymbol{TRS.Sign}(pp, sk_i, \Gamma, \mu)$ and returns it to $\mathcal{A}$.

Game1: A simulator is used $\mathcal{S} = (\mathcal{S}_k, \mathcal{S}_\pi)$ instead of a real

zero-knowledge proof protocol, $\mathcal{S}_1$ replaces $\mathcal{C}$, and the adversary $\mathcal{A}$ interacts with $\mathcal{S}_1$, where:

(1) $\mathcal{S}_k$ replaces the true NIZK generation public parameter;

(2) During the challenge phase, in which $\mathcal{S}_1$ generates a signature for the challenger, $\mathcal{S}_1$ is not executing the genuine NIZK algorithm. but simulates the algorithm $\mathcal{S}_\pi$ by calling the simulator $\mathcal{S}$ to generate a proof $\delta*$ and replaces the real signature δ_{real}.

Since the underlying zero-knowledge proofs have zero-knowledge properties, there: $Adv^{Anon}_{\mathcal{A},Game1}(\lambda) \approx Adv^{Anon}_{\mathcal{A},Game0}(\lambda)$.

Game2: $\mathcal{S}_2$ replaces $\mathcal{S}_1$, the adversary $\mathcal{A}$ interacts with $\mathcal{S}_2$, where:

(1) In the challenge phase, upon generation of a signature for a message μ by $\mathcal{S}_2$, $\mathcal{S}_2$ creates a new table S, and then instead of using $F_{sk_b}(\Gamma^*)$ to generate $\boldsymbol{b}_\pi^{(*)}$, $\mathcal{S}_2$ randomly selected a vector $\boldsymbol{b}_r^{(*)}$ from $\mathbb{Z}_p^m$ and returned it to the opponent $\mathcal{A}$.

(2) In signature query, when $\mathcal{A}$ query a signature from $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$, $\mathcal{S}_2$ receives the tuple $(\Gamma_i, \mu_i, \boldsymbol{b}_r^{(i)})$ and first checks if the same tuple exists in table S. If it is, it generates a signature by replacing $\boldsymbol{b}_r^{(i)}$ with $\boldsymbol{b}_\pi^{(i)}$ and runs a simulated proof algorithm with $\boldsymbol{b}_r^{(i)}$. Else proceeds in the same way as the above query phase, where $\mathcal{S}_2$ randomly samples elements $\boldsymbol{b}_r^{(i)}$ from $\mathbb{Z}_p^m$. Note that due to restriction 1, if Γ_i is an element in S, the tuple $(\Gamma_i, \mu_i, \boldsymbol{b}_r^{(i)})$ must exist in S.

In Game2, the pseudo-random function F_{sk_b} is executed only during the challenge phase or when the adversary $\mathcal{A}$ makes a query signature $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$ to $\mathcal{S}_2$. Thus, there is a set S for executing F_{sk_b}. Therefore, there is a set to execute based on pseudo-randomness, this set is as follows: $(\Gamma, \boldsymbol{b}: \boldsymbol{b} \leftarrow F_{sk_b}(\Gamma)) \approx_c (\Gamma, \boldsymbol{b}: \boldsymbol{b} \leftarrow \mathbb{Z}_p^m)$. Therefore, $Adv^{Anon}_{\mathcal{A},Game2}(\lambda) \approx Adv^{Anon}_{\mathcal{A},Game1}(\lambda)$

Game3: $\mathcal{S}_3$ replaces $\mathcal{S}_2$ and $\mathcal{A}$ interacts with $\mathcal{S}_3$ where:

If $\mathcal{A}$ submits a signed query for $\boldsymbol{Sign}_{\boldsymbol{sk_b}}$, the system will not randomly select $\boldsymbol{b}_r$ from $\mathbb{Z}_p^m$, but rather utilize sk_1 to generate $\boldsymbol{b}_\pi$. Furthermore, in the challenge phase, $\mathcal{S}_3$ employs sk_1 to generate $\boldsymbol{b}_\pi^{(*)}$.

It is difficult for the adversary $\mathcal{A}$ to distinguish whether the vector $\boldsymbol{b}_r$ is computed by running F_{sk_b} or is randomly chosen from a uniform distribution $\mathbb{Z}_p^m$ because of the pseudo-randomness of the pseudo-random function F_{sk_b}. Therefore there is: $Adv^{Anon}_{\mathcal{A},Game3}(\lambda) \approx Adv^{Anon}_{\mathcal{A},Game2}(\lambda)$.

Game4: $\mathcal{S}_4$ replaces $\mathcal{S}_3$ and $\mathcal{A}$ interacts with $\mathcal{S}_4$ where:

In the challenge phase, $\mathcal{S}_4$ runs the real NIZK algorithm to obtain the public parameters and the proof δ. Since zero-knowledge proofs of zero-knowledgeability, we obtain $Adv^{Anon}_{\mathcal{A},Game4}(\lambda) \approx Adv^{Anon}_{\mathcal{A},Game3}(\lambda)$.

Game4 is equivalent to Definition 7, which $b = 1$.

To conclude, the likelihood that $\mathcal{A}$ can distinguish the distinction between the two games is negligible, so $Adv^{Anon}_{\mathcal{A},Game4}(\lambda) \approx Adv^{Anon}_{\mathcal{A},Game0}(\lambda)$. i.e., $Adv^{anon}_{\mathcal{A}}(\lambda) \leq negl(\lambda)$.

Lemma 6 (Exculpability). A TRS is exempt if the LWR assumption holds and NIZKoK is zero-knowledge, the family of hash functions H is collision-resistant, the parameters satisfy $m \geq n \cdot (log\, q + 1)/(log\, p - 1)$, and the intersection-free range of the functions F^H.

Proof of Lemma 6. Owing to pseudo-randomness, Game0 and Game1 are indistinguishable. In Game1, the probability that the adversary $\mathcal{A}$ succeeds in distinguishing between the output of the pseudo-random function F and the random selection is negligible.

Game0: This game is designed as described in Definition 8. Challenger $\mathcal{C}$ obtains the system parameters pp and the public-private key pair (sk, pk) through the initialization and key generation algorithms, and obtains the corresponding signatures through $\boldsymbol{Sign}_{\boldsymbol{sk}}$, which are honestly returned to adversary $\mathcal{A}$.

Game1: $\mathcal{S}_1$ replaces $\mathcal{C}$ where:

The public parameters are generated by the NIZK's simulator, and $\mathcal{S}_1$ randomly selected private key sk_i from $\mathbb{Z}_q^n$ is transmitted to the adversary $\mathcal{A}$, who then computes the public key corresponding to sk_i. $\mathcal{S}_1$ initialize a table $S = \emptyset$. When $\mathcal{A}$ queries $\boldsymbol{Sign}_{\boldsymbol{sk_i}}$, $\mathcal{S}_1$ first check whether the set S contains tuple $(pk^*, \Gamma_i, \mu_i, \boldsymbol{b}_r^{(i)})$. If it does, generates a signature by replacing $\boldsymbol{b}_\pi^{(i)}$ with $\boldsymbol{b}_r^{(i)}$ and runs $\mathcal{S}_\pi$; else, randomly selects vectors $\boldsymbol{b}_r^{(i)} \leftarrow \mathbb{Z}_p^m$ and saves the tuple $(pk^*, \Gamma_i, \mu_i, \boldsymbol{b}_r^{(i)})$ in table S. At the same time, the signature is generated by $\boldsymbol{b}_r^{(i)}$.

Due to the pseudo-random nature of the random function F, there:

$(\Gamma, \boldsymbol{b}: \boldsymbol{b} \leftarrow F_{sk}(\Gamma)) \approx_c (\Gamma, \boldsymbol{b}: \boldsymbol{b} \xleftarrow{\$} \mathbb{Z}_p^m)$. So, $Adv^{Excul}_{\mathcal{A},Game1}(\lambda) \approx Adv^{Excul}_{\mathcal{A},Game0}(\lambda)$.

Suppose $\mathcal{A}$ outputs two tuples $(\Gamma^*, \bar{\mu}, \bar{\sigma})$ and $(\Gamma^*, \mu', \sigma')$, satisfy: $\boldsymbol{TRS.Verify}(pp, \Gamma^*, \bar{\mu}, \bar{\sigma}) = 1$, $\boldsymbol{TRS.Verify}(pp, \Gamma^*, \mu', \sigma') = 1$ and $\boldsymbol{TRS.Trace}(pp, \Gamma^*, \bar{\mu}, \bar{\sigma}, \mu', \sigma') = pk^*$, where π^* is the index of pk^* in the ring R^*.

Consider two scenarios:

(1) There is only one tuple queried in signature query $\boldsymbol{Sign}_{\boldsymbol{sk}*}$. Suppose (Γ^*, μ') is a tuple that has been queried and $(\Gamma^*, \bar{\mu})$ has not. Suppose that there exists an extractor ε capable of extracting the witness $w = (\pi', sk_{\pi'})$ from the underlying zero-knowledge proofs. Since tuple (Γ^*, μ') has been queried, the returned signature is supposed to be $\sigma^* = (\alpha^*, \delta^*)$. $w = (\pi', sk_{\pi'})$ is still the witness of tuple $(\Gamma^*, \mu', \sigma^*)$, then compute $\boldsymbol{b}^*_{\pi'} = F_{sk_{\pi'}}(\Gamma^*)$, where $\boldsymbol{b}^*_{\pi'} = H_2(\Gamma^*, \mu') + \pi' \cdot \alpha^*$, In Game1, $\boldsymbol{b}^*_{\pi'}$ is indistinguishable from uniformly distributed random vectors, and $\boldsymbol{b}^*_{\pi'} = F_{sk_{\pi'}}(\Gamma^*)$ is found to satisfy the π' corresponding to the public key $pk_{\pi'}$ in the ring. If the corresponding public key $pk_{\pi'}$ is found, it indicates that the adversary $\mathcal{A}$ is capable of distinguishing between elements generated by a pseudo-random function F and those generated by a random selection. The results reveal a contradiction between the adversary $\mathcal{A}$ being able to successfully attack and the pseudo-randomness of F. The probability of the adversary $\mathcal{A}$ succeeding in an attack is negligible.

(2) Neither tuple $(\Gamma^*, \bar{\mu})$ nor (Γ^*, μ') is queried by $\boldsymbol{Sign}_{\boldsymbol{sk}*}$. As the third condition holds, the vector $\boldsymbol{b}_i$ is equivalent to the vector indexed by π^* in $\boldsymbol{b}'_i$, i.e., $H_2(\Gamma^*, \bar{\mu}) + \pi^* \bar{\alpha} = H_2(\Gamma^*, \mu') + \pi^* \alpha'$, by the definition of traceability, which is the same as in case 1. To prove the protocol property based on the underlying zero-knowledge suppose that extractor ε is able to extract the witness $w' = (\pi', sk_{\pi'})$ and $w = (\bar{\pi}, sk_{\bar{\pi}})$ from $\bar{\delta}$ and δ' where, $pk_{\bar{\pi}} = \boldsymbol{A} \cdot sk_{\bar{\pi}}$, $pk_{\pi'} = \boldsymbol{A} \cdot sk_{\pi'}$, $F_{sk_{\bar{\pi}}}(\Gamma^*) = H_2(\Gamma^*, \bar{\mu}) + \bar{\pi} \cdot \bar{\alpha}$, $F_{sk_{\sigma'}}(\Gamma) = H_2(\Gamma, \mu') + \pi' \cdot \alpha'$, and by extension:

$$H_2(\Gamma,\bar{\mu}) + \pi^* \frac{F_{sk_{\bar{\pi}}}(\Gamma^*) - H_2(\Gamma^*,\mu)}{\bar{\pi}} = H_2(\Gamma^*, M') + \pi^* \frac{F_{sk_{\pi'}}(\Gamma^*) - H_2(\Gamma^*,\mu')}{\pi'} \quad (18)$$

Consider two cases. The first case is when $\bar{\pi} = \pi'$, there is $\pi' \cdot \bar{\boldsymbol{b}}_0 + \pi^*(\boldsymbol{b}_{\pi'} - \bar{\boldsymbol{b}}_0) = \pi' \cdot \boldsymbol{b}'_0 + \pi^*(\boldsymbol{b}_{\pi'} - \boldsymbol{b}'_0)$. Since $\bar{\boldsymbol{b}}_0 \neq \boldsymbol{b}'_0$ can be simply computed to get $\bar{\pi} = \pi' = \pi^*$. But in Game1, pk_i is selected randomly and the SIS problem is hard, it is highly probable that the corresponding private key $sk_{\bar{\pi}}$ will remain unidentified unless the SIS difficulty can be resolved. The other case is when $\bar{\pi} \neq \pi'$, i.e., $((\Gamma,\bar{\mu}),(\Gamma,\mu'),H_2(\Gamma,\bar{\mu}),H_2(\Gamma,\mu')) \in R_a$, contradicts the definition of R_a as a sparse relation proposed by Banerjee et al [21]. If $\mathcal{A}$ successfully launches a strike, then this will contradict the multiple input correlation of H_{hk}.

To summarize, the probability that the adversary $\mathcal{A}$ attack succeeds is negligible, i.e., $Adv_{\mathcal{A}}^{Excul}(\lambda) \leq negl(\lambda)$, so the scheme satisfies exculpability.

7. Efficiency Analysis

In our scheme, the signature is composed of two elements: a sequence of zero-knowledge proofs δ and a vector α. The size of the vector is primarily determined by the length of the sequence of zero-knowledge proofs δ, as the length of the vector α is fixed. The size of the proof sequence is: $\| \delta \| = (log(2p+1) + \kappa + (3l_1 + 2l_2 + 2\mathfrak{n} + 2\mathfrak{l}) \cdot log\, q) \cdot N + (l_1 + n) \cdot log\, q$ bit, where $\mathfrak{n}$ is the size of the witness and $\mathfrak{l}$ is the size of $\mathcal{M}$.

In the relational formulas in Section 5, each of which can be transformed into an instance of the relation. Table 2 illustrates the size of each component in each relationship equation, where $\mathfrak{R}_{(1)}$ to $\mathfrak{R}_{(6)}$ represent the instances of the relationships corresponding to Equations (8) to (13), respectively.

Table 2. Theoretical size of each relationship.

Relation	Size of witness	Size of $\boldsymbol{\mathcal{M}}$
$\boldsymbol{\mathfrak{R}_{(1)}}$	$(m+n)k_q$	$(m+n)k_q$
$\boldsymbol{\mathfrak{R}_{(2)}}$	$(n+2m)k_q + mk$	$(n+2m)k_q + mk$
$\boldsymbol{\mathfrak{R}_{(3)}}$	$mk_q + nk_q + mk_p$	$mk_q + nk_q + mk_p$
$\boldsymbol{\mathfrak{R}_{(4)}}$	$L + nk_q$	$L + nk_q$
$\boldsymbol{\mathfrak{R}_{(5)}}$	$3L + 2l + n$	$3L + 2l + n$
$\boldsymbol{\mathfrak{R}_{(6)}}$	$2\ell + 4n\ell + 2nk\ell$	$\ell + 2n\ell + 2nk\ell$

Since witness $(\boldsymbol{s}_\pi, pk_\pi, \boldsymbol{b}_\pi, r, w, \boldsymbol{d}_\pi)$ is reused in the relation $\mathfrak{R}$, it only needs to be computed once. In summary, the size of the witness is: $\mathfrak{W} = (2n+m)k_q + mk_p + 3L + 2l + 2\ell + 4n\ell + 2nk\ell + n$ bit.

The size of: $\boldsymbol{\mathcal{M}}$ is: $\mathfrak{M} = (2n+m)k_q + mk_p + \ell + 2n\ell + 2nk\ell + 3L + 2l + n$ bit.

In summary, the length of the proof sequence is: $\|\delta\| = (log(2\hat{p}+1) + \hat{\kappa} + (3\hat{l}_1 + 2\hat{l}_2) \cdot k + 2\mathfrak{W} + 2\mathfrak{M}) \cdot \widehat{N} + (\hat{l}_1 + \mathfrak{W}) \cdot k$ bit, where $\hat{p}, \hat{\kappa}, \hat{l}_1, \hat{l}_2, \widehat{N}$ are the parameters in the proof of zero knowledge in literatures [16], The length of the signature is: $\|\sigma\| = (log(2\hat{p}+1) + \hat{\kappa} + (3\hat{l}_1 + 2\hat{l}_2) \cdot k + 2(\mathfrak{W} + \mathfrak{M})) \cdot \widehat{N} + (\hat{l}_1 + \mathfrak{W}) \cdot k + mk_p$ bit.

Our ring signature scheme provides ring members with the ability to hide sensitive information from the ring members, while allowing for verification within a certain range. In terms of functionality compared to other schemes as shown in Table 3.

Table 3. Functional comparison of ring signatures on lattice.

Scheme	Post-quantum	Linkable	Traceable	Verifiable Range
Libert [12]	√	×	×	×
Cao [23]	√	√	×	×
Feng [14]	√	√	√	×
Ours	√	√	√	√

The schemes of Libert et al. [12] and Feng et al. [14] are constructed based on the Stern protocol, its computational complexity is higher than the efficient zero-knowledge proof protocol used in this paper. Additionally the soundness error in this paper is lower than that of the Stern protocol, the number of times of zero-knowledge proof protocol execution is less than that of the Stern protocol, i.e., $\hat{t} < t$, so the communication overhead of this scheme is less; the signature length in Cao et al. [23] is linearly related to the number of ring members, in this paper, we use an accumulator to reduce the signature length to a logarithmic relationship, so there is a clear advantage in the case of a large number of ring members, and a comparison between the literature and the present scheme in terms of communication overhead is given in Table 4.

Table 4. Theoretical estimation of communication overheads lattice.

Scheme	Public Key	Private Key	Signature Length
Libert [12]	$\mathcal{O}(n \log q)$	$\mathcal{O}(n \log q)$	$t \cdot \mathcal{O}(n \log^3 q + \log N \cdot n \log^2 q)$
Cao [23]	$\mathcal{O}(n \log n \cdot \log q)$	$\mathcal{O}(n^2 \log q)$	$\mathcal{O}(N \cdot n \log q)$
Feng [14]	$\mathcal{O}(n \log^2 q)$	$\mathcal{O}(n \log q)$	$t \cdot \mathcal{O}(n \log^2 q + \log N \cdot n \log q)$
Ours	$\mathcal{O}(n \log q)$	$\mathcal{O}(n \log q)$	$\hat{t} \cdot \mathcal{O}(n \log^2 q + \log N \cdot n \log q)$

8. Conclusion

We have developed a traceable ring signature scheme with verifiable range, incorporating an efficient zero-knowledge proof protocol. While the scheme currently allows for tracing the identity of potential abusers, it is limited to tracking no more than k signatures. This threshold enhances the flexibility of the scheme, making it more adaptable to various application scenarios. For instance, in large-scale e-voting systems or financial transactions where multiple signatures are involved, our scheme ensures that the anonymity of honest participants is preserved while allowing for the identification of malicious users. Furthermore, the current traceability algorithm is designed for one-time signatures. Moving forward, we plan to enhance the tracking algorithm to extend its capabilities, allowing it to handle up to k signatures, thereby making the scheme more versatile and scalable for diverse environments such as blockchain, IoT, and large-scale anonymous reporting systems.

Acknowledgements

This thesis does not require the support of any grant program.

References

[1] R. L. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret," in Advances in Cryptology — ASIACRYPT 2001, C. Boyd, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 552–565.

[2] J. K. Liu, V. K. Wei, and D. S. Wong, "Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups," in Information Security and Privacy, H. Wang, J. Pieprzyk, and V. Varadharajan, Eds., Berlin, Heidelberg: Springer, 2004, pp. 325–335. doi: 10.1007/978-3-540-27800-9_28.

[3] E. Fujisaki and K. Suzuki, "Traceable Ring Signature," in Public Key Cryptography – PKC 2007, T. Okamoto and X. Wang, Eds., Berlin, Heidelberg: Springer, 2007, pp. 181–200. doi: 10.1007/978-3-540-71677-8_13.

[4] M. Ajtai, "Generating hard instances of lattice problems (extended abstract)," in Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing, in STOC '96. New York, NY, USA: Association for Computing Machinery, Jul. 1996, pp. 99–108. doi: 10.1145/237814.237838.

[5] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in Proceedings of the fortieth annual ACM symposium on Theory of computing, in STOC '08. New York, NY, USA: Association for Computing Machinery, May 2008, pp. 197–206. doi: 10.1145/1374376.1374407.

[6] Y. Ren, H. Guan, and Q. Zhao, "An efficient lattice-based linkable ring signature scheme with scalability to multiple layer," J Ambient Intell Human Comput, vol. 13, no. 3, pp. 1547–1556, Mar. 2022, doi: 10.1007/s12652-021-03092-1.

[7] S. Dong, Y. Zhou, Y. Yang, and Y. Yao, "A certificateless ring signature scheme based on lattice," Concurrency and Computation: Practice and Experience, vol. 34, no. 28, p. e7385, 2022, doi: 10.1002/cpe.7385.

[8] T. H. Yuen, M. F. Esgin, J. K. Liu, M. H. Au, and Z. Ding, "DualRing: Generic Construction of Ring Signatures with Efficient Instantiations," in Advances in Cryptology – CRYPTO 2021, T. Malkin and C. Peikert, Eds., Cham: Springer International Publishing, 2021, pp. 251–281. doi: 10.1007/978-3-030-84242-0_10.

[9] Q. Ye, M. Wang, H. Meng, F. Xia, and X. Yan, "Efficient Linkable Ring Signature Scheme over NTRU Lattice with Unconditional Anonymity," Computational Intelligence and Neuroscience, vol. 2022, p. e8431874, May 2022, doi: 10.1155/2022/8431874.

[10] Y. Zhou, S. Dong, and Y. Yang, "Ring Signature Scheme Based on Lattice and Its Application on Anonymous Electronic Voting," KSII Transactions on Internet and Information Systems, vol. 16, no. 1, pp. 287–304, Jan. 2022.

[11] Q. Ye, Y. Lang, H. Guo, and Y. Tang, "Efficient lattice-based traceable ring signature scheme with its application in blockchain," Information Sciences, vol. 648, p. 119536, Nov. 2023, doi: 10.1016/j.ins.2023.119536.

[12] B. Libert, S. Ling, K. Nguyen, and H. Wang, "Zero-Knowledge Arguments for Lattice-Based Accumulators: Logarithmic-Size Ring Signatures and Group Signatures Without Trapdoors," J Cryptol, vol. 36, no. 3, p. 23, May 2023, doi: 10.1007/s00145-023-09470-6.

[13] J. Stern, "A new identification scheme based on syndrome decoding," in Advances in Cryptology — CRYPTO' 93, D. R. Stinson, Ed., Berlin, Heidelberg: Springer, 1994, pp. 13–21. doi: 10.1007/3-540-48329-2_2.

[14] H. Feng, J. Liu, D. Li, Y.-N. Li, and Q. Wu, "Traceable ring signatures: general framework and post-quantum security," Des. Codes Cryptogr., vol. 89, no. 6, pp. 1111–1145, Jun. 2021, doi: 10.1007/s10623-021-00863-x.

[15] T. N. Nguyen et al., "Efficient Unique Ring Signatures from Lattices," in Computer Security – ESORICS 2022, V. Atluri, R. Di Pietro, C. D. Jensen, and W. Meng, Eds., Cham: Springer Nature Switzerland, 2022, pp. 447–466. doi: 10.1007/978-3-031-17146-8_22.

[16] R. Yang, M. H. Au, Z. Zhang, Q. Xu, Z. Yu, and W. Whyte, "Efficient Lattice-Based Zero-Knowledge Arguments with Standard Soundness: Construction and Applications," in Advances in Cryptology – CRYPTO 2019, A. Boldyreva and D. Micciancio, Eds., Cham: Springer International Publishing, 2019, pp. 147–175. doi: 10.1007/978-3-030-26948-7_6.

[17] J. Liang, J. Huang, Q. Huang, L. Lan, and M. H. A. Au, "A Lattice-Based Certificateless Traceable Ring Signature Scheme," Information, vol. 14, no. 3, Art. no. 3, Mar. 2023, doi: 10.3390/info14030160.

[18] J. Liang, Q. Huang, J. Huang, L. Lan, and M. H. A. Au, "An identity-based traceable ring signatures based on lattice," Peer-to-Peer Netw. Appl., vol. 16, no. 2, pp. 1270–1285, Mar. 2023, doi: 10.1007/s12083-023-01474-0.

[19] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," in Proceedings of the thirty-seventh annual ACM symposium on Theory of computing, in STOC '05. New York, NY, USA: Association for Computing Machinery, May 2005, pp. 84–93. doi: 10.1145/1060590.1060603.

[20] R. Yang, M. Au, J. Lai, Q. Xu, and Z. Yu, "Lattice-Based Techniques for Accountable Anonymity: Composition of Abstract Stern's Protocols and Weak PRF with Efficient Protocols from LWR," IACR Cryptol. ePrint Arch., 2017, Accessed: May 07, 2024. [Online]. Available: https://www.semanticscholar.org/paper/Lattice-Based-Techniques-for-Accountable-Anonymity%3A-Yang-Au/e46b7e3534aad60748e7fc2c19cf46b7d1e387e1

[21] A. Banerjee, C. Peikert, and A. Rosen, "Pseudorandom Functions and Lattices," in Advances in Cryptology – EUROCRYPT 2012, D. Pointcheval and T. Johansson, Eds., Berlin, Heidelberg: Springer, 2012, pp. 719–737. doi: 10.1007/978-3-642-29011-4_42.

[22] A. Kawachi, K. Tanaka, and K. Xagawa, "Concurrently Secure Identification Schemes Based on the Worst-Case Hardness of Lattice Problems," in Advances in Cryptology - ASIACRYPT 2008, J. Pieprzyk, Ed., Berlin, Heidelberg: Springer, 2008, pp. 372–389. doi: 10.1007/978-3-540-89255-7_23.

[23] C. Cao, L. You, and G. Hu, "A Novel Linkable Ring Signature on Ideal Lattices," Entropy, vol. 25, no. 2, Art. no. 2, Feb. 2023, doi: 10.3390/e25020237.