Post-Quantum Secure k-Times Traceable Ring Signature

Vishal Pareek[1], Aditi Kar Gangopadhyay[1], and Sugata Gangopadhyay[2]

[1] Department of Mathematics, IIT Roorkee, Haridwar, Uttarakhand 247667, India
vishal1@ma.iitr.ac.in, aditi.gangopadhyay@ma.iitr.ac.in
[2] Department of Computer Science, IIT Roorkee, Haridwar, Uttarakhand 247667, India
sugata.gangopadhyay@cs.iitr.ac.in

Abstract. Ring signatures are cryptographic primitives that enable a user to sign a message on behalf of a group of users in an anonymous manner. The anonymity of the signer is the fundamental security property of ring signatures. However, unrestricted anonymity can be misused, as a malicious signer could use it to send spam or excessive messages. To address this issue, a controlled restriction on signer anonymity is required, which makes such schemes more practical.
In this paper, we propose a k-times traceable ring signature scheme that allows the public key of the signer to be traced publicly if the signer exceeds a predetermined signing limit k. The novelty of our construction lies in its reliance on lattice-based assumptions, which ensure post-quantum security. Consequently, the proposed scheme is well-suited for practical deployment in the presence of emerging quantum threats. Our approach achieves competitive performance while providing stronger security guarantees, making it an appropriate candidate for modern cryptographic applications. We also present efficiency analysis and compare our scheme with existing constructions.

Keywords: Lattice · Post-Quantum Cryptography · Ring Signature Scheme · Merkle Tree.

1 Introduction

One of the most well-known cryptographic primitives is ring signatures [12], which allow a user to sign a message anonymously by creating a ring with other users' public keys. These schemes have been widely used in real-world applications, such as Internet of Medical Things (IoMT) environments, cryptocurrency and blockchain systems, and electronic voting. Unrestricted anonymity can be abused in certain situations, such as when a signer repeatedly signs messages for spamming. Therefore, it becomes necessary to impose some limitations on it. To overcome this restriction, Fujisaki and Suzuki introduced traceable ring signatures [7], which enable a signer's identity to be disclosed if predetermined conditions are broken. In some applications, it is necessary to restrict the number of times a user can sign a message or access a service anonymously to a

maximum of k times; if a user exceeds this limit, the system can trace the user's identity. A ring signature scheme that offers such controlled anonymity is known as a k-times traceable ring signature.
Related Work: Rivest et al. [12] introduced the concept of ring signature. Many variants of ring signatures have been proposed over time, such as traceable ring signatures [7], unique ring signatures [6], linkable ring signatures [10], and k-times traceable ring signatures [3]. Canard et al. [4] designed a 1-time traceable signature scheme, which served as an inspiration for many later works [6,8,15,19,5,3,16] in this field. Furthermore, the schemes in [3,16] create k-times traceable ring signatures based on the Discrete Logarithm problem, which is vulnerable to quantum attacks because of Shor's revolutionary algorithm [14]. To overcome this issue, lattice-based traceable ring signature schemes that are resistant to quantum attacks have been proposed in [19,5]. To the best of our knowledge, there is no existing post-quantum k-times traceable ring signature scheme in the literature, indicating a clear research gap.

To fill this gap, we propose a lattice-based k-times traceable ring signature, denoted as kTRS^+. Our proposed signature scheme limits the number of times a signer can sign the same or different messages anonymously to at most k times. This is achieved by restricting the number of bases to k, each of which can be used to generate a unique tag by utilizing a lattice-based weak pseudorandom function (wPRF) [17]. A signer's identity can be detected publicly if they use the same tag to generate two distinct signatures. As a result, a signer can anonymously produce up to k signatures before traceability is enforced, making the ring signature scheme k-times traceable. Our main contributions to this paper are as follows:

- We design a k-times traceable ring signature scheme that imposes a constraint on the anonymity of the signer. The key advantage of our construction is that it is quantum-resistant, with security based on hard lattice problems, namely the SIS (Shortest Integer Solution) and LWR (Learning with Rounding) assumptions. To the best of our knowledge, this is the first post-quantum secure k-times traceable ring signature scheme.
- To prove that a signer is a member of the ring, we employ a modified Merkle tree accumulator, as described in Section 3. This Merkle tree also contains the accumulation of tags from a tag-bases, which are used during signing. These tag bases are important for applying the threshold on the number of times a signer can anonymously generate signatures.
- We provide a detailed concrete efficiency and computational analysis of our kTRS^+ scheme in Section 5. A comparative analysis with existing schemes is also conducted to demonstrate the effectiveness of our construction. Moreover, the concrete parameter setup facilitates a more comprehensive understanding and practical evaluation of the scheme.

2 Preliminaries

2.1 Notations

We adopt the following conventions throughout the paper. Bold lowercase letters like $\mathbf{u}, \tilde{\mathbf{u}}$ denote column vectors, while bold uppercase letters like $\mathbf{A}, \tilde{\mathbf{A}}$ represent matrices. The transpose of a vector $\mathbf{u}$ is denoted by $\mathbf{u}^\top$. For vectors $\mathbf{u}$ and $\mathbf{v}$, their concatenation is written as $(\mathbf{u}^\top \parallel \mathbf{v}^\top)^\top$, resulting in a column vector. The infinity norm is denoted by $\|\cdot\|_\infty$. The function $\mathsf{bin}(\cdot)$ denotes the binary decomposition of an integer or a vector. The $\mathsf{neg}(\cdot)$ represents a negligible function in the security parameter.

Lattice Hard Assumptions

Definition 1. ***Short Integer Solution (SIS) [1]:*** *For parameters (m, n, q, β), the $\mathsf{SIS}_{m,n,q,\beta}$ problem is defined as follows: given a matrix $\widetilde{\mathbf{B}} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$ chosen uniformly at random, find a non-zero vector $\mathbf{s} \in \mathbb{Z}_q^m$ such that*

$$\widetilde{\mathbf{B}}\mathbf{s} = \mathbf{0} \bmod q \quad \textit{and} \quad \|\mathbf{s}\|_\infty \leq \beta.$$

Definition 2. ***Learning With Rounding (LWR) [2]:*** *Let $\mathbf{s} \in \mathbb{Z}_q^m$ be a secret vector. The $\mathsf{LWR}_{m,q,p}$ distribution $L_{\mathbf{s}}$ is defined over $\mathbb{Z}_q^m \times \mathbb{Z}_p$ as follows: sample $\mathbf{a} \xleftarrow{\$} \mathbb{Z}_q^m$ uniformly at random and compute $b = \lfloor \langle \mathbf{a}, \mathbf{s} \rangle \rceil_p$, where $\langle \cdot, \cdot \rangle$ denotes the standard inner product and the rounding maps elements from $\mathbb{Z}_q$ to $\mathbb{Z}_p$.*

The decision-LWR problem asks to distinguish between m_1 independent samples drawn from $L_{\mathbf{s}}$ and m_1 samples chosen uniformly at random from $\mathbb{Z}_q^m \times \mathbb{Z}_p$.

2.2 Weak Pseudorandom Function (wPRF)

Using the approach of [17], we describe a weak pseudorandom function (wPRF) based on the Learning With Rounding (LWR) problem. Let $\mathcal{F}$ be a function keyed by elements from $\mathbb{Z}_q^{m_1}$ and defined over the domain $\mathbb{Z}_q^{m_2\times m_1}$ with range $\mathbb{Z}_p^{m_2}$. In this case, $\gamma' = q/p$ is an odd integer, and m_1, m_2, p, and q are integers such that $p \geq 2$. The parameters must meet the following requirements for pseudorandomness:

$$m_2 \geq m_1 \cdot \frac{\log q + 1}{\log p - 1}.$$

The function $\mathcal{F}$ consists of the following algorithms:

- Key Generation: On input the security parameter λ, sample a key $\mathsf{usk} \xleftarrow{\$} \mathbb{Z}_q^{m_1}$ uniformly at random.
- Evaluation: Given an input matrix $\widetilde{\mathbf{T}} \in \mathbb{Z}_q^{m_2\times m_1}$ and a key $\mathsf{usk} \in \mathbb{Z}_q^{m_1}$, compute

$$\mathcal{F}_{\mathsf{usk}}(\widetilde{\mathbf{T}}) = \left\lfloor \widetilde{\mathbf{T}} \cdot \mathsf{usk} \right\rceil_p = \left\lfloor \frac{p}{q} \cdot \widetilde{\mathbf{T}} \cdot \mathsf{usk} \right\rceil \bmod p.$$

The security of $\mathcal{F}$ is captured by the following result:

Theorem 1 ([17]). *Assuming the hardness of the* LWR *problem, the function* $\mathcal{F}$ *is a secure weak pseudorandom function when the parameter condition* $m_2 \geq m_1 \cdot \frac{\log q+1}{\log p-1}$ *is satisfied.*

3 Modified Merkle Tree Accumulator

Libert et al. [9] proposed a Merkle tree accumulator scheme for proving group membership. In our construction, however, the signature must incorporate a tag vector derived from a predefined set of tag bases. Specifically, we modify the hash function from [9] to suit our intended setting. First, we define the modified family of hash functions:

Definition 3.1: Hash Family

The function family

$$\widetilde{\mathcal{H}} = \Big\{ \widetilde{h}_{\mathbf{D}} : \{0,1\}^{3\widetilde{m}k} \to \{0,1\}^{\widetilde{m}k} \;\Big|\; \mathbf{D} = [\mathbf{D}_{0,0} \mid \mathbf{D}_{0,1} \mid \mathbf{D}_{0,2}],\ \mathbf{D}_{0,i} \in \mathbb{Z}_q^{\widetilde{m}\times\widetilde{m}k} \Big\}.$$

For $(\mathbf{b}, \mathbf{u}_0, \mathbf{u}_1) \in \{0,1\}^{\widetilde{m}k}$,

$$\widetilde{h}_{\mathbf{D}}(\mathbf{b}, \mathbf{u}_0, \mathbf{u}_1) = \mathrm{bin}\big(\mathbf{D}_{0,2}\mathbf{b} + \mathbf{D}_{0,0}\mathbf{u}_0 + \mathbf{D}_{0,1}\mathbf{u}_1 \bmod q\big).$$

Equivalently,

$$\widetilde{h}_{\mathbf{D}}(\mathbf{b}, \mathbf{u}_0, \mathbf{u}_1) = \mathbf{u} \iff \mathbf{D}_{0,2}\mathbf{b} + \mathbf{D}_{0,0}\mathbf{u}_0 + \mathbf{D}_{0,1}\mathbf{u}_1 = \widetilde{\mathbf{G}}\mathbf{u} \pmod q,$$

where $\widetilde{\mathbf{G}} = \mathbf{I}_{\widetilde{m}} \otimes \left(1\ 2\ 2^2 \cdots 2^{k-1}\right)$.

Lemma 1. *The hash function (Definition 3) is collision-resistant, assuming the hardness of the* SIS *problem.*

Proof. Let $\mathbf{D} = [\mathbf{D}_{0,0} \mid \mathbf{D}_{0,1} \mid \mathbf{D}_{0,2}] \xleftarrow{\$} \mathbb{Z}_q^{\widetilde{m}\times 3\widetilde{m}k}$. Suppose one can find two distinct triples $(\mathbf{b}, \mathbf{u}_0, \mathbf{u}_1), (\mathbf{b}', \mathbf{u}_0', \mathbf{u}_1') \in \left(\{0,1\}^{\widetilde{m}k}\right)^3$ such that

$$\widetilde{h}_{\mathbf{D}}(\mathbf{b}, \mathbf{u}_0, \mathbf{u}_1) = \widetilde{h}_{\mathbf{D}}(\mathbf{b}', \mathbf{u}_0', \mathbf{u}_1') \pmod q.$$

Then one can obtain a non-zero vector

$$\mathbf{z} = \begin{pmatrix} \mathbf{b} - \mathbf{b}' \\ \mathbf{u}_0 - \mathbf{u}_0' \\ \mathbf{u}_1 - \mathbf{u}_1' \end{pmatrix} \in \{-1, 0, 1\}^{3\widetilde{m}k}$$

such that

$$\begin{aligned}\mathbf{D}\cdot\mathbf{z} &= \mathbf{D}_{0,2}(\mathbf{b}-\mathbf{b}') + \mathbf{D}_{0,0}(\mathbf{u}_0-\mathbf{u}_0') + \mathbf{D}_{0,1}(\mathbf{u}_1-\mathbf{u}_1')\\ &= \widetilde{\mathbf{G}}\cdot\Big(\widetilde{h}_{\mathbf{D}}(\mathbf{b},\mathbf{u}_0,\mathbf{u}_1) - \widetilde{h}_{\mathbf{D}}(\mathbf{b}',\mathbf{u}_0',\mathbf{u}_1')\Big)\\ &= \mathbf{0} \pmod q.\end{aligned}$$

Hence, $\mathbf{z}$ is a valid solution to the $\mathsf{SIS}^{\infty}_{\widetilde{m},q,1}$ problem associated with matrix $\mathbf{D}$.

The modified Merkle-tree accumulator is defined as a collection of four algorithms $\mathsf{MTA}^{+} = \{\mathsf{Setup}, \mathsf{Accm}, \mathsf{Wtn}, \mathsf{Vrf}\}$. These algorithms are described below.

Modified Merkle Tree Accumulator

–$\mathsf{Setup}(1^{\lambda})$. Sample a random matrix $\mathbf{D} \leftarrow \mathbb{Z}_q^{\widetilde{m}\times 3\widetilde{m}k}$ and publish the public parameters $\mathsf{pp} = \mathbf{D}$.

–$\mathsf{Accm}(\mathsf{pp}, \mathbf{b}, \mathcal{S} = \{\mathbf{f}_j\}_{j=0}^{N-1} \subseteq (\{0,1\}^{\widetilde{m}k})^N)$. Let the binary Merkle tree have depth $l = \log N$, with leaf nodes $\mathbf{u}_{0,0,\ldots,0} = \mathbf{f}_0,\ \ldots,\ \mathbf{u}_{1,1,\ldots,1} = \mathbf{f}_{N-1}$.

1. For each internal node at depth $i \in \{l-1,\ldots,1\}$, compute

$$\mathbf{u}_{j_1,\ldots,j_i} = \widetilde{h}_{\mathbf{D}}\big(\mathbf{b},\ \mathbf{u}_{j_1,\ldots,j_i,0},\ \mathbf{u}_{j_1,\ldots,j_i,1}\big).$$

2. The root is obtained as

$$\mathfrak{u} = \widetilde{h}_{\mathbf{D}}\big(\mathbf{b},\ \mathbf{u}_0,\ \mathbf{u}_1\big),$$

which serves as the accumulator value.

–$\mathsf{Wtn}(\mathsf{pp}, \mathbf{f}, \mathbf{b}, \mathcal{S})$. If $\mathbf{f} \notin \mathcal{S}$, output $\perp$. Otherwise, output the witness consisting of the position of $\mathbf{f}$, its authentication path, and the corresponding sibling nodes:

$$\mathbf{wn} = \big\{(j_1,\ldots,j_l),\ \{\mathbf{p}_i\}_{i=1}^{l},\ \{\mathbf{w}_i\}_{i=1}^{l}\big\}.$$

–$\mathsf{Vrf}(\mathsf{pp}, \mathfrak{u}, \mathbf{f}, \mathbf{b}, \mathbf{wn})$. Using $\mathbf{wn}$, reconstruct the path $\mathbf{p}_l, \mathbf{p}_{l-1}, \ldots, \mathbf{p}_0$ by initializing $\mathbf{p}_l = \mathbf{f}$ and computing

$$\mathbf{p}_i = \bar{j}_{i+1}\cdot\widetilde{h}_{\mathbf{D}}\big(\mathbf{b}, \mathbf{p}_{i+1}, \mathbf{w}_{i+1}\big) + j_{i+1}\cdot\widetilde{h}_{\mathbf{D}}\big(\mathbf{b}, \mathbf{w}_{i+1}, \mathbf{p}_{i+1}\big),$$

for all $i \in \{l-1,\ldots,0\}$. Accept if $\mathbf{p}_0 = \mathfrak{u}$; otherwise reject.

Definition 3. *The modified accumulator scheme* $\mathsf{MTA}^{+} = \{\mathsf{Setup}, \mathsf{Accm}, \mathsf{Wtn}, \mathsf{Vrf}\}$ *is called secure if for all* PPT *adversaries* $\mathcal{A}$*:*

$$\Pr\Big[\,\mathsf{pp} \leftarrow \mathsf{MTA}^{+}.\mathsf{Setup}(1^{\lambda});\ (\mathcal{S}, \mathbf{p}^*, \mathbf{wn}^*) \leftarrow \mathcal{A}(\mathsf{pp}) : \mathbf{p}^* \notin \mathcal{S}$$
$$\wedge\ \mathsf{MTA}^{+}.\mathsf{Vrf}(\mathsf{Accm}(\mathsf{pp}, \mathbf{b}, \mathcal{S}), \mathbf{p}^*, \mathbf{wn}^*) = 1\,\Big] = \mathsf{neg}(\lambda).$$

Theorem 2. *The modified Merkle tree accumulator scheme achieves security as defined in Definition 3, under the assumption that the* SIS *problem is hard.*

The proof follows [9], additionally making use of Lemma 1.

4 k-Times Traceable Ring Signature

Our k-*Times Traceable Ring Signature* scheme enables the identification of a signer who exceeds the predefined limit k of signatures within the system. Such a scheme enforces a threshold on the number of allowable signatures and is applicable in various practical scenarios. We denote our scheme as kTRS^{+}, which consists of the following five algorithms: $\{\mathsf{Setup}, \mathsf{UKGen}, \mathsf{Sign}, \mathsf{Verf}, \mathsf{Trace}\}$. Our scheme employs two hash functions, denoted as $\mathcal{H}_1 : \{0,1\}^* \to \mathbb{Z}_q^{\widetilde{m}\times\widetilde{n}} \times \mathbb{Z}_q^{\widetilde{m}\times\widetilde{n}}$ and $\mathcal{H}_2 : \{0,1\}^* \to \mathbb{Z}_p$. The functionality of each algorithm is as follows:

- $\mathsf{kTRS}^{+}.\mathsf{Setup}(1^\lambda)$: The input to this algorithm is the security parameter λ. It outputs the public parameters for protocol initialization, denoted as

$$\mathsf{pp}_{\mathsf{TRS}} = \{\widetilde{n}, \widetilde{m}, q, p, p_1, \mathsf{k}, k, k_1, \gamma, \mathbf{D}_1, \mathbf{D}_2\},$$

  where $\widetilde{n}, \widetilde{m}, q, p$ are positive integers, p_1 is a large prime, $k = \lceil \log q \rceil$, $\widetilde{m} \geq \widetilde{n} \cdot (\log q + 1)/(\log p - 1)$, $k_1 = \lceil \log p \rceil$, $\gamma = q/p$ is positive integer, $\mathbf{D}_1 \xleftarrow{\$} \mathbb{Z}_q^{\widetilde{m}\times\widetilde{n}}$, $\mathbf{D}_2 \xleftarrow{\$} \mathbb{Z}_q^{\widetilde{m}\times 3\widetilde{n}\widetilde{m}}$ for accumulation scheme and $\mathbf{D}_3 \xleftarrow{\$} \mathbb{Z}_q^{\widetilde{m}\times 2\widetilde{n}\widetilde{m}k}$ to map the pair of tag bases vector to the value space of accumulator. The tag bases are computed publicly by the use of the hash function $\mathcal{H}_1$, such as $\mathcal{B} = \{(\mathbf{T}_j, \widetilde{\mathbf{T}}_j) \in \mathbb{Z}_q^{\widetilde{m}\times\widetilde{n}} \times \mathbb{Z}_q^{\widetilde{m}\times\widetilde{n}} = \mathcal{H}_1(\mathsf{pp}_{\mathsf{TRS}}, j)\}_{j\in[1,\mathsf{k}]}$.
- $\mathsf{kTRS}^{+}.\mathsf{UKGen}(\mathsf{pp}_{\mathsf{TRS}})$: This algorithm generates keys for user by sampling the signing secret key uniformly at random, i.e., $\mathsf{ssk} \xleftarrow{\$} \mathbb{Z}_q^{\widetilde{n}}$, and defines the corresponding public key as $\mathsf{upk} = \mathsf{bin}(\mathcal{F}_{\mathsf{ssk}}(\mathbf{D}_1)) \in \{0,1\}^{\widetilde{m}k_1}$. The output is $(\mathsf{upk}, \mathsf{ssk})$.
- $\mathsf{kTRS}^{+}.\mathsf{Sign}(\mathsf{pp}_{\mathsf{TRS}}, \mathsf{upk}, \mathsf{ssk}, \mathrm{msg}, \mathcal{S})$: This algorithm is executed by the signer holding the key pair $(\mathsf{upk}, \mathsf{ssk})$. If $\mathsf{upk} \notin \mathcal{S}$, it outputs $\perp$, where $\mathcal{S} = \{\mathsf{upk}_i\}_{i\in[0,N-1]}$ denotes the ring of public keys. The signer uses the secret key ssk to produce the j^{th} signature on a message msg. It first derives the tag base $\mathcal{B}_j = (\mathbf{T}_j, \widetilde{\mathbf{T}}_j) = \mathcal{H}_1(\mathsf{pp}_{\mathsf{TRS}}, j)$. With each run of the Sign algorithm, a fresh random salt is generated as $\mathbf{s} \xleftarrow{\$} \{0,1\}^*$. Next, the signer computes

$$\mathbf{t} = \mathcal{F}_{\mathsf{ssk}}(\mathbf{T}_j), \quad \widetilde{\mathbf{t}} = \mathcal{F}_{\mathsf{ssk}}(\widetilde{\mathbf{T}}_j) + \widetilde{c} \cdot \mathbf{G}_1 \cdot \mathsf{upk},$$

  where $\widetilde{c} = \mathcal{H}_2(\mathsf{pp}_{\mathsf{TRS}}, \mathbf{s}, j, \mathrm{msg}, \mathcal{S}) \in \mathbb{Z}_p$. The binary tag vector is obtained as

$$\mathbf{b}_j = \mathsf{bin}\left(\mathbf{D}_3 \cdot \mathsf{bin}(\mathbf{T}_j[1]^\top \parallel \cdots \parallel \mathbf{T}_j[\widetilde{n}]^\top \parallel \widetilde{\mathbf{T}}_j[1]^\top \parallel \cdots \parallel \widetilde{\mathbf{T}}_j[\widetilde{n}]^\top)\right)^\top,$$

  where $\mathbf{T}_j[r]$ and $\widetilde{\mathbf{T}}_j[r]$ denote the r^{th} columns of $\mathbf{T}_j$ and $\widetilde{\mathbf{T}}_j$; bin is the binary decomposition function. The signer then computes an accumulated root $\mathsf{u} \leftarrow$

$\mathsf{MTA}^{+}.\mathsf{Accm}(\mathsf{pp}_{\mathsf{TRS}}, \mathbf{b}_j, \mathcal{S})$, and the corresponding witness for $\mathsf{ssk} \in \mathcal{S}$ along with the j^{th} tag as $\mathbf{wn} \leftarrow \mathsf{MTA}^{+}.\mathsf{Wtn}(\mathsf{pp}_{\mathsf{TRS}}, \mathsf{ssk}, \mathbf{b}_j, \mathcal{S})$. Finally, the signer produces a non-interactive zero-knowledge proof, called the signature proof of knowledge:

$$\begin{aligned} \mathit{\Pi}_{\mathsf{TRS}} = \mathsf{SPK}\big\{(\mathsf{upk}, \mathsf{ssk}, \mathbf{wn}) : \mathsf{upk} = \mathsf{bin}(\mathcal{F}_{\mathsf{ssk}}(\mathbf{D}_1)) \wedge \mathbf{t} = \\ \mathcal{F}_{\mathsf{ssk}}(\mathbf{T}_j) \wedge \widetilde{\mathbf{t}} = \mathcal{F}_{\mathsf{ssk}}(\widetilde{\mathbf{T}}_j) + \widetilde{c} \cdot \mathbf{G}_1 \cdot \mathsf{upk} \wedge \\ \mathsf{MTA}^{+}.\mathsf{Vrf}(\mathsf{pp}_{\mathsf{TRS}}, \mathfrak{u}, \mathsf{ssk}, \mathbf{b}_j, \mathbf{wn}) = 1\big\}[\mathrm{msg}], \end{aligned}$$

where $\mathbf{G}_1 = \mathbf{I}_{\widetilde{m}} \otimes \big(1\ 2\ 2^2 \cdots 2^{k_1-1}\big)$ is a gadget matrix which maps a binary-decomposed vector back to its integer representation. The resultant matrix of zero-knowledge argument is discussed in Section 4.2. The final signature is $\mathit{\Sigma} = \{\mathit{\Pi}_{\mathsf{TRS}}, \widetilde{\mathbf{t}}, \mathbf{t}, j, \mathbf{s}\}$.

- $\mathsf{kTRS}^{+}.\mathsf{Verf}(\mathsf{pp}_{\mathsf{TRS}}, \mathrm{msg}, \mathcal{S}, \mathit{\Sigma})$: The verifier checks the validity of $\mathit{\Sigma}$. It recomputes $\widetilde{c} = \mathcal{H}_2(\mathsf{pp}_{\mathsf{TRS}}, \mathbf{s}, j, \mathrm{msg}, \mathcal{S}), \mathbf{b}_j, \mathfrak{u} \leftarrow \mathsf{MTA}^{+}.\mathsf{Accm}(\mathsf{pp}_{\mathsf{TRS}}, \mathbf{b}_j, \mathcal{S})$, and verifies $\mathit{\Pi}_{\mathsf{TRS}}$. If the proof holds, output 1; otherwise, output 0. Valid signatures are stored in the authentication log $\mathcal{AL} = \{(\mathit{\Sigma}_i, \widetilde{c}_i)\}_i$.
- $\mathsf{kTRS}^{+}.\mathsf{Trace}(\mathsf{pp}_{\mathsf{TRS}}, \mathcal{AL})$: If two signatures $\mathit{\Sigma}_{i_1}, \mathit{\Sigma}_{i_2} \in \mathcal{AL}$ satisfy $\mathbf{t}_{i_1} = \mathbf{t}_{i_2}$ and $\widetilde{c}_{i_1} \neq \widetilde{c}_{i_2}$, then the signer has exceeded the limit k. To trace the signer, compute $\widetilde{\mathbf{t}}_{i_1} - \widetilde{\mathbf{t}}_{i_2} = (\widetilde{c}_{i_1} - \widetilde{c}_{i_2}) \cdot \mathbf{G}_1 \cdot \mathsf{upk}$, which reveals the cheater's public key as
$$\mathsf{upk} = \mathsf{bin}\big((\widetilde{c}_{i_1} - \widetilde{c}_{i_2})^{-1}(\widetilde{\mathbf{t}}_{i_1} - \widetilde{\mathbf{t}}_{i_2})\big).$$

Remark: To protect against replay attacks, the verifier computes a signature identifier $\mathsf{sid} = \mathcal{H}_3(\mathit{\Sigma})$, $\mathcal{H}_3$ is a cryptographic hash function. The verifier checks whether sid already exists in the authentication log $\mathcal{AL}$. If $\mathsf{sid} \in \mathcal{AL}$, the signature is rejected as a replay; otherwise, the signature is accepted, and sid is added to $\mathcal{AL}$.

4.1 Correctness of scheme

The correctness of our traceable ring signature scheme relies on the correctness of the accumulator scheme and the completeness of the zero-knowledge proof. The verification algorithm always accepts the non-interactive zero-knowledge proof $\mathit{\Pi}_{\mathsf{TRS}}$ from a honest signer.

Traceability: The correctness of the tracing algorithm follows as below. Consider two valid signatures $\mathit{\Sigma}_{i_1} = \{\mathit{\Pi}_{\mathsf{TRS},i_1}, \widetilde{\mathbf{t}}_{i_1}, \mathbf{t}_{i_1}, j'_{i_1}, \mathbf{s}_{i_1}\}$ and $\mathit{\Sigma}_{i_2} = \{\mathit{\Pi}_{\mathsf{TRS},i_2}, \widetilde{\mathbf{t}}_{i_2}, \mathbf{t}_{i_2}, j'_{i_2}, \mathbf{s}_{i_2}\}$, generated by the same user with the same tag base $\mathcal{B}_j = (\mathbf{T}_j, \widetilde{\mathbf{T}}_j)$. Calculate,

$$\begin{aligned} \widetilde{\mathbf{t}}_{i_1} - \widetilde{\mathbf{t}}_{i_2} &= \mathcal{F}_{\mathsf{ssk}}(\widetilde{\mathbf{T}}_j) + \widetilde{c}_{i_1} \cdot \mathbf{G}_1 \cdot \mathsf{upk} - (\mathcal{F}_{\mathsf{ssk}}(\widetilde{\mathbf{T}}_j) + \widetilde{c}_{i_2} \cdot \mathbf{G}_1 \cdot \mathsf{upk}) \\ &= (\widetilde{c}_{i_1} - \widetilde{c}_{i_2}) \cdot \mathbf{G}_1 \cdot \mathsf{upk} \end{aligned}$$

Since $c_{i_1}, c_{i_2} \in \mathbb{Z}_p$ and $c_{i_1} \neq c_{i_2}$ due to $\mathbf{s}_{i_1} \neq \mathbf{s}_{i_2}$, the inverse $(\widetilde{c}_{i_1} - \widetilde{c}_{i_2})^{-1}$ exists. Therefore, we can recover

$$\mathsf{upk} = \mathsf{bin}\big((\widetilde{c}_{i_1} - \widetilde{c}_{i_2})^{-1}(\widetilde{\mathbf{t}}_{i_1} - \widetilde{\mathbf{t}}_{i_2})\big).$$

4.2 Zero-knowledge argument

The resulting system for the zero-knowledge argument takes the form $\mathbf{A} \cdot \mathbf{x} = \mathbf{y} \bmod q$, where

$$\mathbf{A} = \begin{bmatrix} \mathbf{D}_1\mathbf{G}_2 & \mathbf{E}_1 & -\gamma\mathbf{G}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{T}_j & \mathbf{0} & \mathbf{0} & \mathbf{E}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \widetilde{\mathbf{T}}_j & \mathbf{0} & \gamma\widetilde{c}\mathbf{G}_1 & \mathbf{0} & \mathbf{E}_1 & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{D}_{0,j_1} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{D}_{0,\bar{j}_1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{D}_{0,j_2} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{D}_{0,\bar{j}_2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{D}_{0,j_l} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{D}_{0,\bar{j}_l} \end{bmatrix}$$

$$\mathbf{x}^\top = \left[\overline{\mathsf{ssk}}^\top\ \bar{\mathbf{e}}_1^\top\ \mathsf{upk}^\top\ \bar{\mathbf{e}}_2^\top\ \bar{\mathbf{e}}_3^\top\ \mathbf{p}_1^\top\ \mathbf{p}_2^\top \cdots \mathbf{p}_l^\top\ \mathbf{w}_1^\top\ \mathbf{w}_2^\top \cdots \mathbf{w}_l^\top\right]^\top,$$

$$\mathbf{y} = \left[\mathbf{0}^\top\ \gamma \cdot \mathbf{t}^\top\ \gamma \cdot \widetilde{\mathbf{t}}^\top\ (\mathbf{p}_0 - \mathbf{D}_{0,2}\mathbf{b})^\top\ -\mathbf{D}_{0,2}\mathbf{b}^\top \cdots -\mathbf{D}_{0,2}\mathbf{b}^\top\right].$$

The length of the secret vector is $\widetilde{n}k + \widetilde{m}(3\bar{k}_1 + k_1(1+2l))$.

5 Efficiency analysis

The efficiency and comparative analysis of our proposed scheme kTRS^+ are covered in this section. We utilize the zero-knowledge proof from [18] and analyze both the communication cost and the computational cost of the signing and verification algorithms. Following the construction presented in [18], the parameters for 80-bit security in our proposed scheme are set as follows: $\widetilde{n} = 10$, $q = 2^{120}$, $p = 2^6$, $\widetilde{m} = 242$, $k = 120$, $k_1 = 8$, $\bar{k}_1 = 112$, and $l = 10$. The length of the secret vector is $\eta = 89056$. The length of constraints must also be the same because the secret vectors' components are binary. The communication cost of the non-interactive zero-knowledge proof (NIZKP) [18] is

$$\left(\log(2\hat{p}+1) + \hat{\kappa} + \left(3\hat{l}_1 + 2\hat{l}_2 + 2\eta + 2|\mathcal{M}|\right) \cdot k\right) \cdot \Gamma + \left(\hat{l}_1 + \eta\right) \cdot k \text{ bits.}$$

The remaining parameters are $\hat{p} = 2^{40}$, $\hat{l}_1 = 3700$, $\hat{l}_2 = 4000$, and $\Gamma = 2$. The signature size of kTRS^+ for 80-bit security is 12.07 MB. Table 1 compares our scheme with several existing ring signature schemes and shows that our scheme achieves comparable performance while providing k-times traceability and resistance against quantum attacks. Figure 1 presents a comparison of the signature sizes of our scheme with existing post-quantum schemes [9,18,11] for 128-bit security under varying ring sizes. Our scheme performs comparably to [11,18] and significantly outperforms [9]; however, in contrast to these schemes, it also supports k-times traceability. This additional functionality makes our scheme suitable for a wider range of real-world applications. We evaluate the signing and verification time of our proposed kTRS^+ scheme and compare it with existing k-times traceable ring signature schemes under varying ring sizes

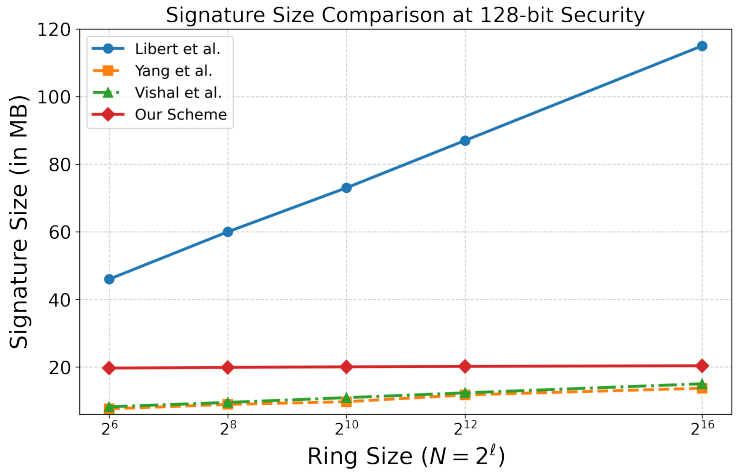
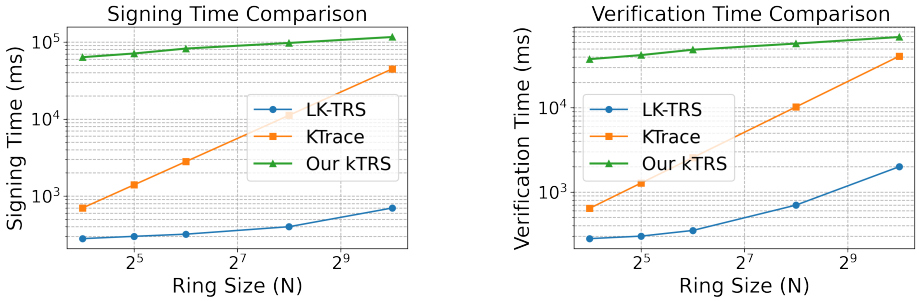


Fig. 1: Signature size comparison at 128-bit security between our scheme and existing post-quantum schemes [9,18,11] for different ring sizes.

in Figure 2. Although our construction does not outperform prior schemes in terms of absolute computational efficiency, its primary advantage lies in being lattice-based and thus secure against quantum attacks. On the other hand, the schemes in [3] and [16] are vulnerable to quantum adversaries because they rely on classical hardness assumptions.

Scheme	**Quantum Safe**	**Traceable**	**Hardness**	**Model**	**Sig. Size**
[7]	No	Yes	DL+DDH	RO	$O(N)$
[3]	No	k-times	DDH	RO	$O(N)$
[13]	Yes	Yes	Hash func.	RO	$O(N)$
[16]	No	k-times	DL+DDH	RO	$O(1)$
[9]	Yes	No	SIVP	RO	$O(\log N)$
[18]	Yes	N0	GapSVP	RO	$O(\log N)$
[5]	Yes	Yes	SIS+LWE	QRO	$O(\log N)$
[11]	Yes	No	SIS+LWR	RO	$O(\log N)$
kTRS^+(Our)	Yes	k-times	SIS+LWR	RO	$O(\log N)$

Table 1: Comparison of our scheme with existing ring signature schemes. Here, N denotes the number of users in the ring, RO represents the random oracle model, and QRO denotes the quantum random oracle model.



(a) Signing time comparison (b) Verification time comparison

Fig. 2: Computation time comparison of our kTRS^+ with existing k-times schemes LK-TRS [16] and KTrace [3].

6 Security Analysis

Our signature scheme kTRS^+ ensures anonymity, unforgeability, and k-traceability. The adversary is given access to the following oracles:

- $\mathcal{O}_{\mathsf{sign}}(j, \mathrm{msg}, \mathcal{R} \subseteq \mathcal{S}_{pk})$: Outputs $\Sigma \leftarrow \mathsf{Sign}(j, \mathrm{msg}, \mathcal{R})$, where $\mathsf{upk}_j \in \mathcal{R}$, and records $(\mathrm{msg}, \mathcal{R}, \Sigma)$ in the signing oracle set $\mathcal{SS}$.
- $\mathcal{O}_{\mathsf{crpt}}(j)$: Returns the randomness ω_j used in generating $(\mathsf{ssk}_j, \mathsf{upk}_j)$, and adds upk_j to the corruption set $\mathcal{CS}$.

Some experiments and theorems are discussed below:
Experiment $\mathsf{Exp}^{\mathsf{anon}}_{\mathsf{kTRS}^+}$:

1. $\mathsf{pp}_{\mathsf{TRS}} \leftarrow \mathsf{Setup}(1^\lambda)$; for $j \in [N]$, $(\mathsf{upk}_j, \mathsf{ssk}_j) \leftarrow \mathsf{UKGen}(\mathsf{pp}; \omega_j)$. Set $\mathcal{SS}, \mathcal{CS} \leftarrow \emptyset$ and give $(\mathsf{pp}_{\mathsf{TRS}}, \mathcal{S}_{pk} = \{\mathsf{upk_j}\}_{j=1}^N)$ to $\mathcal{A}$.
2. $\mathcal{A}$ accesses $\mathcal{O}_{\mathsf{sign}}$ and $\mathcal{O}_{\mathsf{crpt}}$.
3. $\mathcal{A}$ outputs $(\mathrm{msg}^*, j_0^*, j_1^*, \mathcal{R}^*)$ with $\mathsf{upk}_{j_0^*}, \mathsf{upk}_{j_1^*} \in \mathcal{R}^* \subseteq \mathcal{S}_{pk}$. Challenger samples $b \xleftarrow{\$} \{0,1\}$ and returns $\Sigma^* \leftarrow \mathsf{Sign}(\mathrm{msg}^*, \mathcal{R}^*, \mathsf{ssk}_{j_b^*})$.
4. Continue queries with restriction: no $\mathcal{O}_{\mathsf{sign}}$ query on j_0^*, j_1^* and $j_0^*, j_1^* \notin \mathcal{CS}$.
5. Finally, $\mathcal{A}$ outputs b' and wins if $b' = b$.

Experiment $\mathsf{Exp}^{\mathsf{Unf}}_{\mathsf{kTRS}^+}$:

1. $\mathsf{pp}_{\mathsf{TRS}} \leftarrow \mathsf{Setup}(1^\lambda)$; for $j \in [N]$, $(\mathsf{upk}_j, \mathsf{ssk}_j) \leftarrow \mathsf{UKGen}(\mathsf{pp}; \omega_j)$. Set $\mathcal{SS}, \mathcal{CS} \leftarrow \emptyset$ and give $(\mathsf{pp}_{\mathsf{TRS}}, \mathcal{S}_{pk} = \{\mathsf{upk_j}\}_{j=1}^N)$ to $\mathcal{A}$.
2. $\mathcal{A}$ accesses $\mathcal{O}_{\mathsf{sign}}$ and $\mathcal{O}_{\mathsf{crpt}}$.
3. $\mathcal{A}$ outputs $(\mathrm{msg}^*, \mathcal{R}^*, \Sigma^*)$.
4. $\mathcal{A}$ wins iff $\mathsf{Verf}(\mathsf{pp}_{\mathsf{TRS}}, \mathrm{msg}^*, \mathcal{R}^*, \Sigma^*) = 1$, $(\mathrm{msg}^*, \mathcal{R}^*, \Sigma^*) \notin \mathcal{SS}$, $\mathcal{R}^* \neq \emptyset, \mathcal{R}^* \subseteq \mathbf{S}_{pk} \setminus \mathcal{CS}$.

Our proposed scheme kTRS^+ is secure under these theorems. We discuss the outline of the proofs due to page restrictions.

Theorem 3. *The proposed scheme* kTRS^+ *ensures signer anonymity under the random oracle model* (ROM)*, assuming the hardness of the Learning With Rounding* (LWR) *problem.*

Proof. The proof follows a sequence of hybrid games. Let P_i denote the adversary's success probability in game $\mathbf{G}_i$.

- $\mathbf{G}_0$: This is the real anonymity experiment $\mathsf{Exp}^{\mathsf{anon}}_{\mathsf{kTRS}^+}$.
- $\mathbf{G}_1$: Same as $\mathbf{G}_0$, except Π^* is simulated using the zero-knowledge property. Therefore, we get $|\mathsf{P}_1 - \mathsf{P}_0| \leq \mathsf{neg}(\lambda)$.
- $\mathbf{G}_2$: Similar to $\mathbf{G}_1$, but the challenger samples $i_0^*, i_1^* \xleftarrow{\$} [N]$ and aborts if either index is queried and returning a random bit. Then, we find

$$\mathsf{P}_2 = \tfrac{1}{2} + \tfrac{1}{N^2}\left(\mathsf{P}_1 - \tfrac{1}{2}\right).$$

- $\mathbf{G}_3$: Identical to $\mathbf{G}_2$, but replace $\mathsf{upk}_{i_0^*}$ (for $b = 0$) with uniform values from the wPRF range. Indistinguishability between $\mathbf{G}_2$ and $\mathbf{G}_3$ comes from the LWE hardness of weak pseudorandomness, we get $|\mathsf{P}_3 - \mathsf{P}_2| \leq \mathsf{neg}(\lambda)$.
- $\mathbf{G}_4$: Same as $\mathbf{G}_3$, but replace $\mathsf{upk}_{i_1^*}$ (for $b = 1$) similarly. Thus, we have $|\mathsf{P}_4 - \mathsf{P}_3| \leq \mathsf{neg}(\lambda)$.

Since the adversary's perspective in $\mathbf{G}_4$ is independent of b, $\mathsf{P}_4 = \frac{1}{2}$. When we combine the aforementioned transitions, each of which results in a negligible change or a bounded loss, we obtain $|\mathsf{P}_0 - \frac{1}{2}| \leq \mathsf{neg}(\lambda)$. This completes the proof of anonymity.

Theorem 4. *The proposed scheme* kTRS^+ *ensures unforgeability under the random oracle model* (ROM)*, assuming the hardness of the Shortest Integer Solution* (SIS) *problem.*

Proof. Let P_i denote the success probability of $\mathcal{A}$ in game $\mathbf{G}_i$.

- $\mathbf{G}_0$: This is the real unforgeability experiment $\mathsf{Exp}^{\mathsf{Unf}}_{\mathsf{kTRS}^+}$.
- $\mathbf{G}_1$: Same as $\mathbf{G}_0$, except signature use simulated proof ZKP Π^*. Due to the underlying zero-knowledge proof, we get $|\mathsf{P}_1 - \mathsf{P}_0| \leq \mathsf{neg}(\lambda)$.
- $\mathbf{G}_2$: Identical to $\mathbf{G}_1$, but upon a forgery $(\mathrm{msg}^*, \mathcal{R}^*, \Sigma^*)$, the challenger extracts a witness $(\mathsf{upk}^*, \mathsf{ssk}^*, \mathbf{wn})$ from π^*. The challenger can extract a valid witness by simulating the extractability of the underlying zero-knowledge proof. Thus, we get $|\mathsf{P}_2 - \mathsf{P}_1| \leq \mathsf{neg}(\lambda)$.
- $\mathbf{G}_3$: Similar to $\mathbf{G}_2$, except the challenger aborts if $\mathsf{upk}^* \notin \mathcal{R}^*$. By the SIS-based accumulator security, $|\mathsf{P}_3 - \mathsf{P}_2| \leq \mathsf{neg}(\lambda)$.
- $\mathbf{G}_4$: Same as $\mathbf{G}_3$, but sample $i^* \xleftarrow{\$} [N]$ and abort if $\mathsf{upk} \neq \mathsf{bin}(\mathcal{F}_{\mathsf{ssk}}(\mathbf{D}_1))$. Since i^* is independent, $\mathsf{P}_4 = \frac{1}{N}\mathsf{P}_3$.

If P_4 is non-negligible, the soundness of the underlying ZKAoK can be broken by extracting a valid witness, which is a contradiction. As a result, P_4 is negligible. By combining the aforementioned,

$$|\mathsf{P}_0 - N\mathsf{P}_4| \leq \mathsf{neg}(\lambda),$$

P_0 is implied to be negligible.

k-Traceability: It is computationally impossible to generate more than k legitimate signatures without being tracked, even if an adversary manages to gain the user's secret key.

Theorem 5. *The proposed scheme* kTRS^{+} *ensures* k*-traceability under the random oracle model* (ROM)*, assuming the hardness of the Learning With Rounding* (LWR) *problem.*

Proof. For the sake of contradiction, suppose that there is a PPT adversary $\mathcal{A}$ that avoids tracing while producing $\mathsf{k}+1$ valid signatures corresponding to the same signing key.

Each signature corresponds to a legitimate witness, i.e., a secret key and its corresponding tag, as all signatures validate the knowledge soundness of the underlying proof system. The function $\mathcal{F}_{\mathsf{ssk}}(\cdot)$ derived from the LWR-based construction is used to compute the tags. If $\mathcal{A}$ uses the same secret key to generate more than k valid signatures, then the associated tags must be connected by the same underlying function. According to the pigeonhole principle, at least two signatures have same tags, i.e.,

$$\mathbf{t} = \mathcal{F}_{\mathsf{ssk}}(\mathbf{T}), \quad \widetilde{\mathbf{t}} = \mathcal{F}_{\mathsf{ssk}}(\widetilde{\mathbf{T}}) + \widetilde{c} \cdot \mathbf{G}_1 \cdot \mathsf{upk}.$$

These relationships make it possible for Trace algorithms to accurately identify the signer. If tracing fails, then either: (i) the adversary distinguishes the outputs of $\mathcal{F}_{\mathsf{ssk}}(\cdot)$ from random, violating the LWR-based pseudorandomness, or (ii) generates valid signatures without a valid witness, which violates the soundness of underlying zero-knowledge proof.

The presumed security is contradicted in both situations. Therefore, the proposed scheme satisfies k-traceability.

7 Conclusion

In this paper, we proposed a lattice-based k-times traceable ring signature scheme that enhance level of security, even when dealing with quantum adversaries. As a result, our scheme is appropriate for the post-quantum setting, aligns with current research trends and practical security requirements. In order to increase the effectiveness of the proposed scheme, we intend to look at polynomial lattices, hybrid lattice constructions, and more efficient zero-knowledge proof in future work.

Acknowledgment

This work was partly supported by the Information Security Education and Awareness (ISEA) Phase-3 project funded by MeitY, Government of India (Grant No. L-14017/1/2022-HRD, dated 01.07.2024).

References

1. Ajtai, M.: Generating hard instances of lattice problems. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. pp. 99–108 (1996)
2. Banerjee, A., Peikert, C., Rosen, A.: Pseudorandom functions and lattices. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 719–737. Springer (2012)
3. Bultel, X., Lafourcade, P.: k-times full traceable ring signature. In: 2016 11th International Conference on Availability, Reliability and Security (ARES). pp. 39–48. IEEE (2016)
4. Canard, S., Schoenmakers, B., Stam, M., Traoré, J.: List signature schemes. Discrete Applied Mathematics **154**(2), 189–201 (2006)
5. Feng, H., Liu, J., Li, D., Li, Y.N., Wu, Q.: Traceable ring signatures: general framework and post-quantum security. Designs, Codes and Cryptography **89**(6), 1111–1145 (2021)
6. Franklin, M., Zhang, H.: A framework for unique ring signatures. Cryptology ePrint Archive (2012)
7. Fujisaki, E., Suzuki, K.: Traceable ring signature. In: International Workshop on Public Key Cryptography. pp. 181–200. Springer (2007)
8. Gu, K., Wang, L., Wu, N., Liao, N.: Traceable certificateless ring signature scheme for no full anonymous applications. Int. J. Netw. Secur. **20**(4), 762–773 (2018)
9. Libert, B., Ling, S., Nguyen, K., Wang, H.: Zero-knowledge arguments for lattice-based accumulators: logarithmic-size ring signatures and group signatures without trapdoors. In: Annual International Conference on the Theory and Applications of Cryptographic Techniques. pp. 1–31. Springer (2016)
10. Liu, J.K., Wei, V.K., Wong, D.S.: Linkable spontaneous anonymous group signature for ad hoc groups. In: Australasian Conference on Information Security and Privacy. pp. 325–335. Springer (2004)
11. Pareek, V., Gangopadhyay, A.K., Gangopadhyay, S.: Lattice-based linkable ring signatures for anonymous and accountable whistleblowing. Cryptology ePrint Archive, Paper 2025/2170 (2025), `https://eprint.iacr.org/2025/2170`
12. Rivest, R.L., Shamir, A., Tauman, Y.: How to leak a secret. In: International conference on the theory and application of cryptology and information security. pp. 552–565. Springer (2001)
13. Scafuro, A., Zhang, B.: One-time traceable ring signatures. In: European symposium on research in computer security. pp. 481–500. Springer (2021)
14. Shor, P.W.: Algorithms for quantum computation: discrete logarithms and factoring. In: Proceedings 35th annual symposium on foundations of computer science. pp. 124–134. Ieee (1994)
15. Tang, F., Pang, J., Cheng, K., Gong, Q.: Multiauthority traceable ring signature scheme for smart grid based on blockchain. Wireless Communications and Mobile Computing **2021**(1), 5566430 (2021)
16. Xie, J., Zhou, J., Cao, Z., Dong, X., Choo, K.K.R.: Linkable, k-times traceable and revocable ring signature for fine-grained accountability in blockchain transactions. IEEE Internet of Things Journal (2024)
17. Yang, R., Au, M.H., Lai, J., Xu, Q., Yu, Z.: Lattice-based techniques for accountable anonymity: composition of abstract stern's protocols and weak prf with efficient protocols from lwr. Cryptology ePrint Archive (2017)

18. Yang, R., Au, M.H., Zhang, Z., Xu, Q., Yu, Z., Whyte, W.: Efficient lattice-based zero-knowledge arguments with standard soundness: Construction and applications. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology – CRYPTO 2019. pp. 147–175. Springer International Publishing, Cham (2019)
19. Ye, Q., Lang, Y., Guo, H., Tang, Y.: Efficient lattice-based traceable ring signature scheme with its application in blockchain. Information Sciences **648**, 119536 (2023)