Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Verifier-local revocation group signatures with backward unlinkability from lattices*

Yanhua ZHANG†‡[1], Ximeng LIU[2], Yupu HU[3], Yong GAN[4], Huiwen JIA†‡[5]

[1]*College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450001, China*

[2]*College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China*

[3]*State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an 710071, China*

[4]*College of Information Engineering, Zhengzhou University of Technology, Zhengzhou 450044, China*

[5]*School of Mathematics and Information Science, Guangzhou University, Guangzhou 510006, China*

†E-mail: yhzhang@email.zzuli.edu.cn; hwjia@gzhu.edu.cn

Received Sept. 28, 2020; Revision accepted Mar. 15, 2021; Crosschecked Feb. 21, 2022

Abstract: For group signature (GS) supporting membership revocation, verifier-local revocation (VLR) mechanism seems to be a more flexible choice, because it requires only that verifiers download up-to-date revocation information for signature verification, and the signers are not involved. As a post-quantum secure cryptographic counterpart of classical number-theoretic cryptographic constructions, the first lattice-based VLR group signature (VLR-GS) was introduced by Langlois et al. (2014). However, none of the contemporary lattice-based VLR-GS schemes provide backward unlinkability (BU), which is an important property to ensure that previously issued signatures remain anonymous and unlinkable even after the corresponding signer (i.e., member) is revoked. In this study, we introduce the first lattice-based VLR-GS scheme with BU security (VLR-GS-BU), and thus resolve a prominent open problem posed by previous works. Our new scheme enjoys an $\mathcal{O}(\log N)$ factor saving for bit-sizes of the group public-key (GPK) and the member's signing secret-key, and it is free of any public-key encryption. In the random oracle model, our scheme is proven secure under two well-known hardness assumptions of the short integer solution (SIS) problem and learning with errors (LWE) problem.

Key words: Group signature; Lattice-based cryptography; Verifier-local revocation; Backward unlikability; Short integer solution

https://doi.org/10.1631/FITEE.2000507 **CLC number:** TP309.2

1 Introduction

Group signature (GS) (Chaum and van Heyst, 1991) has two privacy-preserving properties: anonymity and traceability. The former enables each registered member of a group to issue signature on a message on behalf of the whole group without divulging his/her identity information; the latter enables an opening authority, in cases of any disputes, to revoke the anonymity and track the real identity of a misbehaving member. With these two appealing properties, GS has found several applications in real-life scenarios, such as direct anonymous attestation in trusted computing, anonymous vehicular ad-hoc network online communications, e-commerce systems, and many more.

To design an efficient GS theoretically, three critical cryptographic ingredients are required, in a relatively sophisticated combination. These building blocks consist of a digital signature scheme,

‡ Corresponding authors

* Project supported by the National Natural Science Foundation of China (Nos. 61802075 and 61772477) and the Natural Science Foundation of Henan Province, China (Nos. 222300420371 and 202300410508)

ORCID: Yanhua ZHANG, https://orcid.org/0000-0001-7946-5262; Huiwen JIA, https://orcid.org/0000-0002-9289-5918

CCA-secure public-key encryption, and an efficient non-interactive zero-knowledge proof protocol. Since its first introduction three decades ago, much progress has been made in the construction of GS, and some creative schemes based on different mathematical hardness assumptions, with different levels of service functionality and operating efficiency, have been proposed (Bellare et al., 2003, 2005; Boneh and Shacham, 2004; Gordon et al., 2010; Bootle et al., 2016; Emura and Hayashi, 2018; Huang et al., 2020).

Supporting membership revocation, i.e., disabling the signing ability of misbehaving members or honest members who voluntarily leave, is a desirable functionality of many multi-member (or multi-user) signature systems. Revocation not affecting the remaining members is also a non-trivial problem. Specifically, for GS with membership revocation, the verifier-local revocation (VLR) mechanism is a more flexible choice compared with re-initialization of the whole system or a dynamic accumulator, when considering a large group, and more practical and suitable for mobile environments where signers are often off-line or some computationally weak devices (e.g., smart cards) are pervasively adopted. The concept of VLR group signature (VLR-GS) was first formalized by Boneh and Shacham (2004) and subsequently investigated and extended (Nakanishi and Funabiki, 2005, 2006; Libert and Vergnaud, 2009; Ishida et al., 2018). However, all these constructions operate in the bilinear map setting and they may not be able to resist the effective attack of quantum computers in the future post-quantum cryptography era. As the old saying goes, "don't put all your eggs in one basket;" it is encouraging to consider some alternative instantiations, post-quantum constructions, e.g., based on a lattice-based cryptosystem.

Owing to the creative work of Ajtai (1996), Regev (2005), and Gentry et al. (2008), lattice-based cryptography has become a hot field and GS over lattices has been extensively studied.

Lattice-based VLR-GS, introduced by Langlois et al. (2014), is the first quantum-resistant design that supports revocation. Subsequently, improved schemes were proposed (Zhang et al., 2016; Gao et al., 2017; Ling et al., 2018; Perera and Koshiba, 2018a, 2018b, 2018c), but the schemes of Langlois et al. (2014), Ling et al. (2018), and Perera and Koshiba (2018a, 2018b, 2018c) operate within the Bonsai tree (Cash et al., 2010), and feature bit-sizes of group public-key (GPK) and member's signing secret-key proportional to $\log N$; therefore, none of these constructions are suitable for a large group. As two exceptions, Zhang et al. (2016) and Gao et al. (2017) adopted a new identity-encoding function (Nguyen et al., 2015) to encode a member's index and saved a factor $\mathcal{O}(\log N)$ for both bit-sizes, but both needed a series of sophisticated encryptions in the signing phase, which resulted in stronger hardness assumptions. To overcome these somewhat unsatisfactory situations, Zhang et al. (2019a, 2019b) designed an improved Stern-type zero-knowledge proof (ZKP) for the identity-encoding function and a lattice-based VLR-GS achieving smaller key-sizes and explicit traceability.

Backward unlinkability (BU), a significant security first introduced by Song (2001) for GS supporting membership revocation, ensures that the previously issued signatures will remain anonymous and unlinkable even after the corresponding signer is revoked. BU security is essential to ensure privacy for honest members who voluntarily leave the group or inadvertently lose the signing secret-keys. Note that none of the existing lattice-based VLR-GSs provide BU security, because in a conventional lattice-based VLR-GS, the revocation list (RL) is provided for verification, containing a list of revocation tokens (RTs) for the revoked members. Once a member is revoked, the issued signatures cannot be accepted any more. Following this process, it is fairly easy to test whether two different but legal signatures are issued by the same revoked member by performing the verification algorithm twice with the before- and post-revocation RLs, respectively. As a result, all signatures issued by the revoked member will become linkable, which inevitably undermines privacy. This yields two interesting open questions on lattice-based VLR-GS: Is it possible to design a scheme providing BU security? How can we construct a more efficient scheme for a large group?

In this study, we introduce the first lattice-based VLR-GS with BU security (VLR-GS-BU). Our scheme operates on the model of Nakanishi and Funabiki (2005) and is proven secure under two well-known worst-case hardness assumptions: the short integer solution (SIS) problem and the learning with

errors (LWE) problem.

As we discussed earlier, each registered member is given a vector called the RT, which is added to the RL once this member is revoked. Using this unique and immutable vector RT, all signatures issued by the honest revoked members who lose signing secret-keys inadvertently or leave voluntarily can be linked. So, to realize BU security, we require a mechanism in which the RT is updated, which means that each member has many RTs over the lifetime, and the signer may adopt different RTs in the signature algorithm. Once a group member is revoked, the manager can add his/her unused RTs to the RL, and the used ones remain anonymous. Therefore, to achieve BU security, a classical concept called time-periods (TPs), adopted by Song (2001) to realize forward-secure GS, is introduced in our new member RT design. Let the entire time period TP of each group member be t discrete periods, and the member obtains $\mathrm{RT}_1, \mathrm{RT}_2, \ldots, \mathrm{RT}_t$, corresponding to the discrete periods $1, 2, \ldots, t$ from the manager. If a member is revoked at period $j \in \{1, 2, \ldots, t\}$, all his/her RTs after (and including) the current period, $\mathrm{RT}_j, \mathrm{RT}_{j+1}, \ldots, \mathrm{RT}_t$, are added to the RL. Therefore, any signature issued by the revoked member after (and including) j is judged invalid using the verification algorithm, while signatures issued before (not including) j are still valid and remain anonymous. Obviously, BU security holds. Furthermore, during the generation of RTs, the main challenge is to ensure one-way security, which means leaking the revoked member's RTs at period j, i.e., $\mathrm{RT}_j, \mathrm{RT}_{j+1}, \ldots, \mathrm{RT}_t$, no one including an adversary can compute any RT before period j, i.e., RT_1, or RT_2, ..., or RT_{j-1}.

Thus, to reach our goal, that is, to meet the following two requirements–RTs must be generated over TP and the generation of RTs satisfies one-way security, we first assume that the periods are elements in $\mathbb{Z}_q^n$ (the discrete integer TPs can be represented in their binary form $\{0,1\}^*$, and then hashed into $\mathbb{Z}_q^n$ using the collision-resistant hash function). In construction and proof of security, we require an injective encoding function $\mathcal{H}_1 : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ map TPs in $\mathbb{Z}_q^n$ to matrices in $\mathbb{Z}_q^{n\times n}$; a concrete construction of this function, called encoding with full-rank differences (FRD), was first introduced by Agrawal et al. (2010). For t different TPs, i.e., $\mathrm{TP}_1, \mathrm{TP}_2, \ldots, \mathrm{TP}_t$, we obtain t different encodings $\mathcal{H}_1(\mathrm{TP}_1), \mathcal{H}_1(\mathrm{TP}_2), \ldots, \mathcal{H}_1(\mathrm{TP}_t)$. As in Zhang et al. (2019a, 2019b), we adopt a constant number of matrices in Gpk to encode the member's identity index, i.e., $\mathrm{Gpk} = (\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1)$. For a member id with an index $i \in \{0, 1, \ldots, N-1\}$ (in this study, N is the group size, $N = 2^\ell = \mathrm{poly}(n)$, and n is the security parameter), the identity-encoding matrix is $\boldsymbol{A}_{\mathrm{id}} = [\boldsymbol{A} | \boldsymbol{A}_0 + i\boldsymbol{A}_1]$. However, these encoding TPs cannot be directly used for modular multiplication or to replace the public matrices in all known lattice-based VLR-GS schemes (because there is a deadly attack on the underlying SIS problem and BU security). We introduce a simple but insightful tweak that allows us to upgrade the above strategy directly into a secure and one-way setting. By sampling $\boldsymbol{B}_0 \in \mathbb{Z}_q^{n\times m}$ as a perturbation, we can obtain the new RTs for each member, an id with an index $i \in \{0, 1, \ldots, N-1\}$, i.e., $\mathbf{grt}_{i,1}, \mathbf{grt}_{i,2}, \ldots, \mathbf{grt}_{i,t} \in \mathbb{Z}_q^n$, where $\mathbf{grt}_{i,j} = (\boldsymbol{B}_0 + \mathcal{H}_1(\mathrm{TP}_j)\boldsymbol{B}_1)\boldsymbol{e}_{i,0} \bmod q$, $1 \le j \le t$. Here, $\boldsymbol{e}_{i,0} \in \mathbb{Z}^m$ is a short Gaussian vector and the first part of signing secret-key $\boldsymbol{e}_i \in \mathbb{Z}^{2m}$, and $\boldsymbol{B}_1 \in \mathbb{Z}_q^{n\times m}$ is a public matrix used for modular multiplication.

We first construct a lattice-based Stern-type interactive ZKP for all membership relations. Then the protocol is repeated $\omega(\log n)$ times to reduce the soundness error to some negligible value and transform it into a non-interactive one, also, a signature using the Fiat-Shamir paradigm in the random oracle model. To summarize, by incorporating an efficient and improved FRD, a creative RT design, and a corresponding Stern-type statistical ZKP protocol into lattice-based VLR-GS, we introduce the first lattice-based VLR-GS-BU, and thus resolve a prominent open problem.

2 Preliminaries

Table 1 refers to some notations used in this paper.

2.1 Parameters

Our lattice-based VLR-BU-GS scheme involves three main parameters: a security parameter n, the maximum number of members (i.e., group size) $N = 2^\ell = \mathrm{poly}(n)$, and the number of TPs $t = \mathrm{poly}(n)$. The other parameters are listed in Table 2.

Table 1 Notations used in this paper

Notation	Definition
$\mathbb{Z}$	Set of integers
$\mathbb{R}$	Set of real numbers
$\boldsymbol{a}$, $\boldsymbol{b}$	Vectors
$\boldsymbol{A}$, $\boldsymbol{B}$	Matrices
id	Member identity
$\boldsymbol{m}$	Message
$\mathcal{S}_k$	All permutations of k elements
$\xleftarrow{\$}$	Sampling uniformly at random
$\|\cdot\|$	Euclidean norm ℓ_2
$\|\cdot\|_\infty$	Infinity norm ℓ_∞
$\lceil e \rceil$	The smallest integer not less than e
$\mathrm{Parse}(\boldsymbol{e}, k_1, k_2)$	$(e_{k_1}, e_{k_1+1}, \ldots, e_{k_2}) \in \mathbb{R}^{k_2-k_1+1}$, $1 \le k_1 \le k_2 \le n$, $\boldsymbol{e} = (e_1, e_2, \ldots, e_n)$
$\mathcal{O}$, $\widetilde{\mathcal{O}}$, ω	Standard asymptotic notations
$\log e$	Logarithm of e with base 2
PPT	Probabilistic polynomial-time

Table 2 Parameters of our VLR-GS-BU scheme

Parameter	Value or asymptotic bound
Modulus q	$\omega(n^2 \log n) > N$
Dimension m	$2n\lceil \log q \rceil$
Gaussian parameter s	$\omega(\sqrt{n \log q \log n})$
Integer norm bound β	$\lceil s \log m \rceil$, s.t. $(4\beta+1)^2 < q$
FRD function $\mathcal{H}_1$	$\mathbb{Z}_q^n \to \mathbb{Z}_q^{n \times n}$
Hash function $\mathcal{G}$	$\{0,1\}^* \to \mathbb{Z}_q^{n \times m}$
Hash function $\mathcal{H}_2$	$\{0,1\}^* \to \{1,2,3\}^\kappa$
Number of repetitions κ	$\omega(\log n)$

2.2 VLR-GS definition and security model

In this subsection, we present the definition and security model of VLR-GS with BU security, which is extended from Boneh and Shacham (2004) and Nakanishi and Funabiki (2005).

Definition 1 (VLR-GS-BU) A VLR-GS with BU security consists of three algorithms:

1. KeyGen$(1^n, N, t)$: a PPT algorithm takes as input security parameter n, group size N, and number of time-periods t, and outputs Gpk, a set of signing secret-keys $\mathrm{Gsk} = (\mathbf{gsk}_0, \mathbf{gsk}_1, \ldots, \mathbf{gsk}_{N-1})$, and a set of RTs $\mathrm{Grt} = (\mathbf{grt}_{0,1}, \mathbf{grt}_{0,2}, \ldots, \mathbf{grt}_{N-1,t})$. Here, $\mathbf{grt}_{i,j}$ denotes the RT_j for member id with index i at period j.

2. Sign$(\mathrm{Gpk}, j, \mathbf{gsk}_i, \boldsymbol{m})$: a PPT algorithm takes as input Gpk, a signing secret-key $\mathbf{gsk}_i$, current period j for id with index i, and a message $\boldsymbol{m} \in \{0,1\}^*$, and outputs a signature σ.

3. Verify$(\mathrm{Gpk}, j, \mathrm{RL}_j, \sigma, \boldsymbol{m})$: a deterministic algorithm takes as input Gpk, a set of RTs for period j, RL_j, a signature σ, and a message $\boldsymbol{m} \in \{0,1\}^*$, and outputs either invalid or valid. Valid indicates that σ is a valid signature on $\boldsymbol{m}$ at period j, and the signer has not been revoked at period j.

Remark 1 Any VLR-GS, as introduced by Boneh and Shacham (2004), has an implicit-tracing algorithm: given a message-signature pair $(\boldsymbol{m}, \sigma)$ for period j, the party owning Grt can determine the signer of σ by executing the verification algorithm successively, i.e., $\mathrm{Verify}(\mathrm{Gpk}, j, \mathrm{RL}_j = \mathbf{grt}_{i,j}, \sigma, \boldsymbol{m})$ for $i = 0, 1, \ldots, N$, and outputting the first index $i^* \in \{0, 1, \ldots, N-1\}$ when the verification algorithm returns invalid.

A VLR-GS-BU scheme has three properties: correctness, BU-anonymity, and traceability.

Definition 2 (Correctness) A VLR-GS-BU scheme is correct if for all (Gpk, Gsk, Grt) outputted by KeyGen, all periods $j \in \{1, 2, \ldots, t\}$, any member with index $i \in \{0, 1, \ldots, N-1\}$, all $\mathrm{RL}_j \subseteq \mathrm{Grt}$, and $\boldsymbol{m} \in \{0,1\}^*$, we have

$$\mathrm{Verify}(\mathrm{Gpk}, j, \mathrm{RL}_j, \mathrm{Sign}(\mathrm{Gpk}, j, \mathbf{gsk}_i, \boldsymbol{m}), \boldsymbol{m}) = \mathrm{valid} \Leftrightarrow \mathbf{grt}_{i,j} \notin \mathrm{RL}_j.$$

Definition 3 (BU-anonymity) A VLR-GS scheme is BU-anonymous if no PPT adversary $\mathcal{A}$ has a non-negligible advantage $\mathrm{Adv}_{\mathcal{A}}^{\text{BU-anon}}$ in the following game (between a challenger $\mathcal{C}$ and $\mathcal{A}$):

1. Initialization: $\mathcal{C}$ gets $(\mathrm{Gpk}, \mathrm{Gsk}, \mathrm{Grt})$ using KeyGen and provides Gpk to $\mathcal{A}$ (not including Gsk).

2. Query phase: at the beginning of a time-period $j \in \{1, 2, \ldots, t\}$, $\mathcal{C}$ declares j to $\mathcal{A}$, and j must be incremental. During period j, $\mathcal{A}$ adaptively makes a polynomially bounded number of queries:

Signing: requesting for a signature on a message $\boldsymbol{m} \in \{0,1\}^*$ for id with an index i at period j, $\mathcal{C}$ returns $\sigma \leftarrow \mathrm{Sign}(\mathrm{Gpk}, j, \mathbf{gsk}_i, \boldsymbol{m})$.

Corrupting: requesting for a signing secret-key for id with an index i, $\mathcal{C}$ returns $\mathbf{gsk}_i$ to $\mathcal{A}$.

Revoking: requesting for a revocation token RT_j of id with an index i for current period j, $\mathcal{C}$ returns $\mathbf{grt}_{i,j}$ to $\mathcal{A}$.

3. Challenge: $\mathcal{A}$ outputs a period $j^* \in \{1, 2, \ldots, t\}$, a message $\boldsymbol{m}^* \in \{0,1\}^*$, and two distinct members id_0 and id_1, with indices i_0 and i_1, respectively. $\mathcal{A}$ cannot make a corrupting query or a revoking query at either member; i.e., the secret-keys of id_0 and id_1 cannot be corrupted, and id_0 and id_1 have not been revoked before or at j^*. $\mathcal{C}$ chooses a bit $b \in \{0,1\}$, defines σ^* using $\mathrm{Sign}(\mathrm{Gpk}, j^*, \mathbf{gsk}_{i_b}, \boldsymbol{m}^*)$ as a challenge on $\boldsymbol{m}^*$ by id_b, and returns it to $\mathcal{A}$.

4. Restricted query: once σ^* is obtained, $\mathcal{A}$ can still make queries as before, but with restrictions that do not allow a corrupting query or a revoking query for id_0 or id_1 for the periods before (and including) j^*, i.e., in incremental form, and the opening query for $(\boldsymbol{m}^*, \sigma^*)$.

5. Guessing: $\mathcal{A}$ outputs a bit $b^* \in \{0,1\}$, and wins if $b^* = b$.

The advantage of $\mathcal{A}$ in the above game is defined as $\mathrm{Adv}_{\mathcal{A}}^{\text{BU-anon}} = |\Pr[b^* = b] - 1/2|$.

Definition 4 (Traceability) A VLR-GS-BU scheme is traceable if no PPT adversary $\mathcal{A}$ has a non-negligible advantage $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{Trace}}$ in the following game:

1. Initialization: $\mathcal{C}$ gets $(\mathrm{Gpk}, \mathrm{Gsk}, \mathrm{Grt})$ using KeyGen, and provides $(\mathrm{Gpk}, \mathrm{Grt})$ to $\mathcal{A}$. Also, an initial corruption set $\mathrm{Corr} = \varnothing$ is defined.

2. Query phase: $\mathcal{A}$ adaptively makes a polynomially bounded number of queries:

Signing: requesting for a signature on a message $\boldsymbol{m} \in \{0,1\}^*$ for id with an index i at period j, $\mathcal{C}$ returns σ using $\mathrm{Sign}(\mathrm{Gpk}, j, \mathbf{gsk}_i, \boldsymbol{m})$.

Corrupting: requesting for the signing secret-key of id with an index i, $\mathcal{C}$ returns $\mathbf{gsk}_i$ and adds id with its index i to Corr.

3. Forgery: $\mathcal{A}$ outputs $\boldsymbol{m}^* \in \{0,1\}^*$, a period j^*, a set of tokens $\mathrm{RL}_{j^*}^* \subseteq \mathrm{Grt}$, and a signature σ^*. $\mathcal{A}$ wins the game if $\mathrm{Verify}(\mathrm{Gpk}, j^*, \mathrm{RL}_{j^*}^*, \sigma^*, \boldsymbol{m}^*) =$ valid.

The implicit-tracing algorithm fails, or traces to a member outside $\mathrm{Corr}\backslash\mathrm{RL}_{j^*}^*$ (Because σ^* cannot be traced to $i^* \in (\mathrm{Corr} \cap \mathrm{RL}_{j^*}^*)$, $\mathrm{Corr}\backslash\mathrm{RL}_{j^*}^*$ can also be modified to Corr).

The signature σ^* is non-trivial; i.e., $\mathcal{A}$ has not obtained σ^* by making a signing query on $\boldsymbol{m}^*$.

The advantage of $\mathcal{A}$ in this game is defined as its probability of winning, denoted by $\mathrm{Adv}_{\mathcal{A}}^{\mathrm{Trace}} = \mathrm{SuccPT}_{\mathcal{A}}$.

2.3 Background on lattices

Definition 5 (Lattices) For positive integers n, m, $q \geq 2$, and a random $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, an m-dimensional q-ary orthogonal lattice $\Lambda_q^{\perp}(\boldsymbol{A})$ is defined as

$$\Lambda_q^{\perp}(\boldsymbol{A}) = \{\boldsymbol{e} \in \mathbb{Z}^m \mid \boldsymbol{A}\boldsymbol{e} = \mathbf{0} \bmod q\}.$$

For $s > 0$, the Gaussian function on $\mathbb{R}^m$ with center $\boldsymbol{c}$ is

$$\rho_{s,\boldsymbol{c}}(\boldsymbol{e}) = \exp(-\pi\|\boldsymbol{e} - \boldsymbol{c}\|^2/s^2),\ \forall \boldsymbol{e} \in \mathbb{R}^m.$$

For $\boldsymbol{c} \in \mathbb{R}^m$, the discrete Gaussian distribution over Λ is

$$\mathcal{D}_{\Lambda,s,\boldsymbol{c}} = \rho_{s,\boldsymbol{c}}(\boldsymbol{e})/\textstyle\sum_{\boldsymbol{e}\in\Lambda}\rho_{s,\boldsymbol{c}}(\boldsymbol{e}),\ \forall \boldsymbol{e} \in \mathbb{Z}^m,$$

where $\mathcal{D}_{\Lambda,s,\boldsymbol{c}}$ is denoted as $\mathcal{D}_{\Lambda,s}$ if $\boldsymbol{c} = \mathbf{0}$.

Lemma 1 (Gentry et al., 2008) For integers n, $q \geq 2$, $m \geq 2n\lceil\log q\rceil$, let a real number $s \geq \omega(\sqrt{\log m})$; the following properties hold:

1. For all but a $2q^{-n}$ fraction of all $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and $\boldsymbol{e} \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^m,s}$, the distribution of $\boldsymbol{A}\boldsymbol{e} \bmod q$ is statistically close to uniform over $\mathbb{Z}_q^n$.

2. For $\boldsymbol{e} \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^m,s}$ and $\beta = \lceil s\log m\rceil$, $\Pr[\|\boldsymbol{e}\|_\infty \leq \beta]$ is overwhelming.

3. The min-entropy of $\mathcal{D}_{\mathbb{Z}^m,s}$ is at least $m-1$.

Ajtai (1996) introduced how to obtain a matrix $\boldsymbol{A}$ statistically close to uniform together with a low Gram-Schmidt norm basis for $\Lambda_q^{\perp}(\boldsymbol{A})$. Then two improvements were investigated by Alwen and Peikert (2011) and Micciancio and Peikert (2012).

Lemma 2 (Alwen and Peikert, 2011; Micciancio and Peikert, 2012) Let $n \geq 1$, $q \geq 2$, and $m = 2n\lceil\log q\rceil$. There exists a PPT algorithm $\mathrm{TrapGen}(q,n,m)$ that outputs $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and $\boldsymbol{R}_{\boldsymbol{A}}$, such that $\boldsymbol{A}$ is statistically close to a uniform matrix in $\mathbb{Z}_q^{n\times m}$ and $\boldsymbol{R}_{\boldsymbol{A}}$ is a trapdoor for orthogonal lattice $\Lambda_q^{\perp}(\boldsymbol{A})$.

Given a short basis matrix of $\Lambda_q^{\perp}(\boldsymbol{A})$, Gentry et al. (2008) introduced a creative algorithm to sample short vectors from some discrete Gaussian distribution over lattices. Then an improved algorithm was introduced by Micciancio and Peikert (2012).

Lemma 3 (Gentry et al., 2008; Micciancio and Peikert, 2012) Let $n \geq 1$, $q \geq 2$, and $m = 2n\lceil\log q\rceil$. Given $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, a trapdoor $\boldsymbol{R}_{\boldsymbol{A}}$ for $\Lambda_q^{\perp}(\boldsymbol{A})$, a parameter $s = \omega(\sqrt{n\log q\log n})$, and $\boldsymbol{u} \in \mathbb{Z}_q^n$, there exists a PPT algorithm $\mathrm{SamplePre}(\boldsymbol{A}, \boldsymbol{R}_{\boldsymbol{A}}, \boldsymbol{u}, s)$ that returns a short vector $\boldsymbol{e} \in \Lambda_q^{\boldsymbol{u}}(\boldsymbol{A})$ sampled from a distribution statistically close to $\mathcal{D}_{\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A}),s}$, where $\Lambda_q^{\boldsymbol{u}}(\boldsymbol{A})$ is a coset of $\Lambda_q^{\perp}(\boldsymbol{A})$.

We recall two average-case lattice problems: SIS and LWE.

Definition 6 (SIS) The $\mathrm{SIS}_{n,m,q,\beta}^{\infty}$ problem is defined as follows: given a random $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and a real $\beta > 0$, obtain a vector $\boldsymbol{e} \in \mathbb{Z}^m$ s.t. $\boldsymbol{A}\boldsymbol{e} = \mathbf{0} \bmod q$ and $0 < \|\boldsymbol{e}\|_\infty \leq \beta$.

The inhomogeneous small integer solution (ISIS) problem is a new variant of SIS, also given a random syndrome vector $\boldsymbol{u} \in \mathbb{Z}_q^n$. The $\mathrm{ISIS}_{n,m,q,\beta}^{\infty}$ problem is to obtain $\boldsymbol{e} \in \mathbb{Z}^m$ such that $\boldsymbol{A}\boldsymbol{e} = \boldsymbol{u} \bmod q$

and $\|\boldsymbol{e}\|_\infty \leq \beta$. Both problems are as hard as certain worst-case problems, such as the shortest independent vector problem (SIVP).

Lemma 4 (Gentry et al., 2008; Micciancio and Peikert, 2013) For m, $\beta = \text{poly}(n)$ and $q \geq \beta\widetilde{\mathcal{O}}(\sqrt{n})$, the average-case $\text{SIS}^{\infty}_{n,m,q,\beta}$ and $\text{ISIS}^{\infty}_{n,m,q,\beta}$ problems are at least as hard as the $\text{SIVP}_{\beta\widetilde{\mathcal{O}}(n)}$ problem in the worst case.

Definition 7 (LWE) The $\text{LWE}_{n,q,\chi}$ problem is defined as follows: given a random vector $\boldsymbol{s} \xleftarrow{\$} \mathbb{Z}_q^n$, and a probability distribution χ over $\mathbb{Z}$, let $\mathcal{A}_{\boldsymbol{s},\chi}$ be a distribution obtained by sampling $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, $\boldsymbol{e} \xleftarrow{\$} \chi^m$, output $(\boldsymbol{A}, \boldsymbol{A}^{\mathrm{T}}\boldsymbol{s} + \boldsymbol{e} \bmod q)$, and distinguish between $\mathcal{A}_{\boldsymbol{s},\chi}$ and a uniform distribution $\mathcal{U} \xleftarrow{\$} \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m$. Let $\beta \geq \sqrt{n}\omega(\log n)$. For a prime power q, given a β-bounded distribution χ, the $\text{LWE}_{n,q,\chi}$ problem is at least as hard as $\text{SIVP}_{\widetilde{\mathcal{O}}(nq/\beta)}$.

Now let us recall two important facts and a new sampling algorithm that is used in the security proofs of this work:

Lemma 5 (Agrawal et al., 2010) Let $n \geq 1$, and assume that $m > (n+1)\log q + \omega(\log n)$ and q is a prime. Let $\boldsymbol{A}$ and that $\boldsymbol{B}$ be two matrices chosen uniformly in $\mathbb{Z}_q^{n\times m}$, and $\boldsymbol{R}$ an $m \times m$ matrix chosen uniformly in $\{-1,1\}^{m\times m}$. Then, for all $\boldsymbol{w} \in \mathbb{Z}_q^m$, the distribution $(\boldsymbol{A}, \boldsymbol{AR}, \boldsymbol{R}^{\mathrm{T}}\boldsymbol{w})$ is statistically close to $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{R}^{\mathrm{T}}\boldsymbol{w})$.

Lemma 6 (Agrawal et al., 2010) Let $\boldsymbol{R}$ be an $m\times m$ matrix chosen at random from $\{-1,1\}^{m\times m}$. For $\boldsymbol{e} \in \mathbb{R}^m$, $\Pr[\|\boldsymbol{Re}\|_\infty > \|\boldsymbol{e}\|_\infty\omega(\sqrt{\log m})] < \text{negl}(m)$.

Lemma 7 (Agrawal et al., 2010) Let prime $q \geq 3$, $m > n$, matrices $\boldsymbol{A}$, $\boldsymbol{B} \in \mathbb{Z}_q^{n\times m}$, and a real $s \geq \|\widetilde{\boldsymbol{R}}_{\boldsymbol{B}}\|\sqrt{m}\omega(\log m)$. There exists a PPT algorithm $\text{SampleRight}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{R}, \boldsymbol{R}_{\boldsymbol{B}}, \boldsymbol{u}, s)$ that, given a trapdoor $\boldsymbol{R}_{\boldsymbol{B}}$ for $\Lambda_q^{\perp}(\boldsymbol{B})$, a low-norm $\boldsymbol{R} \in \{-1,1\}^{m\times m}$, and $\boldsymbol{u} \in \mathbb{Z}_q^n$, outputs $\boldsymbol{e} \in \mathbb{Z}^{2m}$ distributed statistically close to $\mathcal{D}_{\Lambda_q^{\boldsymbol{u}}(\boldsymbol{F}),s}$, where $\boldsymbol{F} = [\boldsymbol{A}|\boldsymbol{AR} + \boldsymbol{B}]$.

An injective encoding function $\mathcal{H}_1 : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ is adopted for our lattice-based VLR-GS-BU, and a concrete construction of such a function, called FRD, was introduced by Agrawal et al. (2010).

Definition 8 (FRD) A function $\mathcal{H}_1 : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ is called FRD if the following hold:

1. $\forall \boldsymbol{e}_1, \boldsymbol{e}_2 \in \mathbb{Z}_q^n$, $\boldsymbol{e}_1 \neq \boldsymbol{e}_2$, $\mathcal{H}_1(\boldsymbol{e}_1) - \mathcal{H}_1(\boldsymbol{e}_2)$ is invertible; i.e., the rank is n.

2. $\mathcal{H}_1$ is computed in polynomial time, i.e., $n\log q$.

3 Preparations

In this section we describe the main techniques and provide the main building block in our new design of a Stern-type statistical ZKP. First, we adopt the design techniques in Zhang et al. (2019a, 2019b) for member identity encoding. Thus, only a constant number of public matrices are included in Gpk, e.g., $\text{Gpk} = (\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1)$ (actually, two more matrices $\boldsymbol{B}_0$ and $\boldsymbol{B}_1$ are needed in our scheme). We focus on describing our design of member RTs to achieve BU security.

3.1 New design of RTs

The FRD function $\mathcal{H}_1$ in Definition 8 is adopted to realize two essential conditions (many RTs generated over TPs and one-way security) for member RTs with BU security. Thus, for member id with an index i, t different TPs and t different encoding matrices $\mathcal{H}_1(\text{TP}_1), \mathcal{H}_1(\text{TP}_2), \ldots, \mathcal{H}_1(\text{TP}_t)$ are achieved. Further, to avoid some deadly attacks of the underlying SIS problem and BU security, we sample two random matrices $\boldsymbol{B}_0, \boldsymbol{B}_1 \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$ as perturbations. So, for time period j, the revocation token RT_j of member id is $\mathbf{grt}_{i,j} = (\boldsymbol{B}_0 + \mathcal{H}_1(\text{TP}_j)\boldsymbol{B}_1)\boldsymbol{e}_{i,0} \bmod q$, where $\boldsymbol{e}_{i,0}$ is a short Gaussian vector (see the descriptions in Section 4.1 and $\boldsymbol{e}_{i,0}$ is the first part of the signing secret-key $\boldsymbol{e}_i = (\boldsymbol{e}_{i,0}, \boldsymbol{e}_{i,1}) \in \mathbb{Z}^{(2m)}$) satisfying

$$\boldsymbol{A}_{\text{id}}\boldsymbol{e}_i = \boldsymbol{u} \bmod q, \tag{1}$$

where $\boldsymbol{A}_{\text{id}} = [\boldsymbol{A}|\boldsymbol{A}_0 + i\boldsymbol{A}_1] \in \mathbb{Z}_q^{n\times(2m)}$.

So, $\mathbf{grt}_{i,j}$ is generated from the member's signing secret-key. For the revocation mechanism, as stated by Ling et al. (2018), due to a flaw in Langlois et al. (2014) that an inequality test method was adopted to check whether the signer's RT belongs to a given RL, a new and corrected technique which realizes revocation by binding the signer's RT to an LWE function (in our design, the concept of TP is adopted, and for id with an index i at period j, $\text{RT}_j = \mathbf{grt}_{i,j}$) was proposed:

$$\begin{aligned}\boldsymbol{b}_j &= \boldsymbol{B}^{\mathrm{T}}\mathbf{grt}_{i,j} + \boldsymbol{e}_0 \\ &= (\boldsymbol{B}^{\mathrm{T}}\underbrace{(\boldsymbol{B}_0 + \mathcal{H}_1(\text{TP}_j)\boldsymbol{B}_1)}_{\hat{\boldsymbol{B}}_j})\boldsymbol{e}_{i,0} + \boldsymbol{e}_0 \bmod q,\end{aligned} \tag{2}$$

where $\boldsymbol{B}$ is from a random oracle as in Ling et al. (2018) and $\boldsymbol{e}_0 \in \mathbb{Z}^m$ is sampled from an LWE error χ^m.

Putting all innovative ideas, design approaches, and the Stern-type argument system introduced by Ling et al. (2013) together, we have designed a Stern-type interactive ZKP protocol to prove Eqs. (1) and (2), which is described in the next subsection.

3.2 Underlying ZKP protocol

This subsection introduces an underlying interactive Stern-type statistical ZKP protocol that allows $\mathcal{P}$ (a member id with index i) to convince $\mathcal{V}$ that $\mathcal{P}$ is indeed a valid member who signs $\boldsymbol{m} \in \{0,1\}^*$.

In our design of the underlying Stern-type ZKP protocol, decomposition (Dec), extension (Ext), and matrix-extension (Mat-Ext) techniques are adopted. Specific sets, e.g., $\boldsymbol{B}_{2\ell}$, $\boldsymbol{B}_{3m}$, $\mathbf{Sec}_\beta(\mathrm{id})$, $\mathbf{SecExt}(\mathrm{id}^*)$, permutations, e.g., π, $\varphi \in \mathcal{S}_{3m}$, $\tau \in \mathcal{S}_{2\ell}$, and a composition $\mathcal{F}$ are used. We omit all these duplicate concepts and the detailed definitions can be found in the literature (Ling et al., 2013, 2018; Zhang et al., 2016, 2019a, 2019b; Gao et al., 2017).

We first define a function Bin to denote a binary representation of a member's index, i.e., the member $\mathrm{id} = \mathrm{Bin}(i) \in \{0,1\}^\ell$ for $i \in \{0,1,\ldots,N-1\}$, where $N = 2^\ell = \mathrm{poly}(n)$ is the group size. In addition, we define a series of integers $k = \lfloor \log \beta \rfloor + 1$, $\beta_1 = \lceil \frac{\beta}{2} \rceil$, $\beta_2 = \lceil \frac{\beta-\beta_1}{2} \rceil$, $\beta_3 = \lceil \frac{\beta-\beta_1-\beta_2}{2} \rceil, \ldots, \beta_k = 1$.

For Eq. (2), we prove that id is a certified member without leaking $\boldsymbol{e}_i$. As in Zhang et al. (2019a), we transform $\boldsymbol{A}_{\mathrm{id}}$ to $\boldsymbol{A}' = [\boldsymbol{A}|\boldsymbol{A}_0|\boldsymbol{g}_\ell \otimes \boldsymbol{A}_1]$, which is independent of index i. $\boldsymbol{g}_\ell = (1, 2, 2^2, \ldots, 2^{\ell-1})$ is a power-of-two vector (thus, i can be rewritten as $i = \boldsymbol{g}_\ell^{\mathrm{T}}\mathrm{Bin}(i)$), and $\otimes$ denotes a concatenation with vectors or matrices; e.g., given $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, $\boldsymbol{e}' \in \mathbb{Z}_q^m$, and $\boldsymbol{e} = (e_1, e_2, \ldots, e_\ell) \in \mathbb{Z}_q^\ell$, we have the following two definitions: $\boldsymbol{e} \otimes \boldsymbol{e}' = (e_1\boldsymbol{e}', e_2\boldsymbol{e}', \ldots, e_\ell\boldsymbol{e}') \in \mathbb{Z}_q^{m\ell}$, $\boldsymbol{e} \otimes \boldsymbol{A} = [e_1\boldsymbol{A}|e_2\boldsymbol{A}|\ldots|e_\ell\boldsymbol{A}] \in \mathbb{Z}_q^{n\times m\ell}$. A corresponding change to i's signing secret-key $\boldsymbol{e}_i = (\boldsymbol{e}_{i,0}, \boldsymbol{e}_{i,1})$ is transformed into $\boldsymbol{e}'_i = (\boldsymbol{e}_{i,0}, \boldsymbol{e}_{i,1}, \mathrm{bin}(i) \otimes \boldsymbol{e}_{i,1})$. Thus, to argue the relation $\boldsymbol{A}_{\boldsymbol{id}}\boldsymbol{e}_i = \boldsymbol{u} \bmod q$, we instead prove a new relation $\boldsymbol{A}'\boldsymbol{e}'_i = \boldsymbol{u} \bmod q$. For Eqs. (1) and (2), to prove BU security, we need only to prove that they hide the same short Gaussian vector $\boldsymbol{e}_{i,0} \in \mathbb{Z}^m$.

The underlying ZKP protocol between $\mathcal{P}$ and $\mathcal{V}$ can be summarized as follows:

Public input: $\boldsymbol{A}' = [\boldsymbol{A}|\boldsymbol{A}_0|\boldsymbol{g}_\ell \otimes \boldsymbol{A}_1] \in \mathbb{Z}_q^{n\times(\ell+2)m}$, $\boldsymbol{B}$, $\boldsymbol{B}_0$, $\boldsymbol{B}_1 \in \mathbb{Z}_q^{n\times m}$, $\boldsymbol{u} \in \mathbb{Z}_q^n$, $\boldsymbol{b}_j \in \mathbb{Z}_q^m$, and current period $j \in \{1,2,\ldots,t\}$, where $\boldsymbol{B}$ is from a random oracle (Section 4).

$\mathcal{P}$'s witness: $\boldsymbol{e}'_i = (\boldsymbol{e}'_{i,0}, \boldsymbol{e}'_{i,1}, \mathrm{Bin}(i) \otimes \boldsymbol{e}'_{i,1}) \in \mathbf{Sec}_\beta(\mathrm{id})$ for a secret index $i \in \{0,1,\ldots,N-1\}$, and a short vector $\boldsymbol{e}_0 \in \chi^m$, which is an LWE error.

$\mathcal{P}$'s goal: to convince $\mathcal{V}$ in zero-knowledge (ZK) that:

g_1: $\boldsymbol{A}'\boldsymbol{e}'_i = \boldsymbol{u} \bmod q$, where $\boldsymbol{e}'_i \in \mathbf{Sec}_\beta(\mathrm{id})$ and $\mathrm{id} = \mathrm{Bin}(i)$ is kept secret.

g_2: $\boldsymbol{b}_j = (\boldsymbol{B}^{\mathrm{T}}\hat{\boldsymbol{B}}_j)\boldsymbol{e}'_{i,0} + \boldsymbol{e}_0 \bmod q$, where $\boldsymbol{e}'_{i,0}$ is a part of $\boldsymbol{e}'_i$ and $\|\boldsymbol{e}'_{i,0}\|_\infty, \|\boldsymbol{e}_0\|_\infty \le \beta$.

For the group membership mechanism (i.e., $\mathcal{P}$'s goal is g_1), as in Zhang et al. (2019a, 2019b), the following steps are taken:

1. Extend $\boldsymbol{A}'$ to $\boldsymbol{A}^* \in \mathbb{Z}_q^{n\times((2\ell+2)3m)}$, where $\boldsymbol{A}^* = [\boldsymbol{A}|\boldsymbol{0}^{n\times(2m)}|\boldsymbol{A}_0|\boldsymbol{0}^{n\times(2m)}|\ldots|2^{\ell-1}\boldsymbol{A}_1|\boldsymbol{0}^{n\times(2m)}|\boldsymbol{0}^{n\times(3m\ell)}]$, using the Mat-Ext technique.

2. Extend $\mathrm{id} = \mathrm{Bin}(i) = (d_1, d_2, \ldots, d_\ell) \in \{0,1\}^\ell$ to $\mathrm{id}^* = (d_1, d_2, \ldots, d_{2\ell}) \in \boldsymbol{B}_{2\ell}$ using the Ext technique.

3. Extend $\boldsymbol{e}'_{i,0}$ to $\boldsymbol{e}'_{i,0,1}, \boldsymbol{e}'_{i,0,2}, \ldots, \boldsymbol{e}'_{i,0,k} \in \boldsymbol{B}_{3m}$ and $\boldsymbol{e}'_{i,1}$ to $\boldsymbol{e}'_{i,1,1}, \boldsymbol{e}'_{i,1,2}, \ldots, \boldsymbol{e}'_{i,1,k} \in \boldsymbol{B}_{3m}$ using the Dec and Ext techniques.

4. For each $r \in \{1,2,\ldots,k\}$, define $\boldsymbol{e}'_{i,r} = (\boldsymbol{e}'_{i,0,r}, \boldsymbol{e}'_{i,1,r}, d_1\boldsymbol{e}'_{i,1,r}, \ldots, d_{2\ell}\boldsymbol{e}'_{i,1,r})$.

5. It can be checked that $\boldsymbol{e}'_{i,r} \in \mathbf{SecExt}(\mathrm{id}^*)$.

So, $\mathcal{P}$'s goal is transformed into

$$\begin{cases} \boldsymbol{A}^*\left(\sum_{r=1}^{k}\beta_r\boldsymbol{e}'_{i,r}\right) = \boldsymbol{u} \bmod q, \\ \boldsymbol{e}'_{i,r} \in \mathbf{SecExt}(\mathrm{id}^*). \end{cases} \tag{3}$$

To prove Eq. (3), as in Zhang et al. (2019a, 2019b), we take the following two steps:

1. Sample $\boldsymbol{r}'_1, \boldsymbol{r}'_2, \ldots, \boldsymbol{r}'_k \xleftarrow{\$} \mathbb{Z}_q^{(2\ell+2)\times(3m)}$ to mask $\boldsymbol{e}'_{i,1}, \boldsymbol{e}'_{i,2}, \ldots, \boldsymbol{e}'_{i,k}$, and it can be checked that

$$\boldsymbol{A}^*\left(\sum_{r=1}^{k}\beta_r(\boldsymbol{e}'_{i,r} + \boldsymbol{r}'_r)\right) - \boldsymbol{u} = \boldsymbol{A}^*\left(\sum_{r=1}^{k}\beta_r\boldsymbol{r}'_r\right) \bmod q.$$

2. Sample permutations π, $\varphi \in \mathcal{S}_{3m}$, $\tau \in \mathcal{S}_{2\ell}$, and it can be checked that

$$\forall r \in \{1,2,\ldots,k\},\ \mathcal{F}_{\pi,\varphi,\tau}(\boldsymbol{e}'_{i,r}) \in \mathbf{SecExt}(\tau(\mathrm{id}^*)),$$

where $\mathrm{id}^* \in \boldsymbol{B}_{2\ell}$ is an extension of $\mathrm{id} = \mathrm{Bin}(i)$.

For the revocation mechanism (i.e., $\mathcal{P}$'s goal is g_2), the following steps are taken:

1. Define $\boldsymbol{B}' = \boldsymbol{B}^{\mathrm{T}}\hat{\boldsymbol{B}}_j \bmod q \in \mathbb{Z}_q^{m\times m}$.

2. Let $\boldsymbol{e}'_{i,0,r} = \mathrm{Parse}(\boldsymbol{e}'_{i,r}, 1, m)$.

3. Parse $\boldsymbol{e}_0 = (e_{0,1}, e_{0,2}, \ldots, e_{0,m})$ and extend it to $\boldsymbol{e}_{0,1}, \boldsymbol{e}_{0,2}, \ldots, \boldsymbol{e}_{0,k} \in \boldsymbol{B}_{3m}$ using the Dec and Ext techniques.

4. Define $\boldsymbol{B}^*=[\boldsymbol{B}'|\boldsymbol{I}_m|\boldsymbol{0}^{m\times(2m)}]$, where $\boldsymbol{I}_m$ is the identity matrix of order m.

So, $\mathcal{P}$'s goal is transformed into a new relation:

$$\begin{cases}\boldsymbol{b}_j=\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{e}'_{i,0,r},\boldsymbol{e}_{0,r}))\bmod q,\\ \boldsymbol{e}_{0,r}\in\boldsymbol{B}_{3m}.\end{cases}\tag{4}$$

To prove Eq. (4), we take the following three steps:

1. Let $\boldsymbol{r}'_{r,0}=\mathrm{Parse}(\boldsymbol{r}'_r,1,m)$.

2. Sample $\boldsymbol{r}_{0,1},\boldsymbol{r}_{0,2},\dots,\boldsymbol{r}_{0,k}\xleftarrow{\$}\mathbb{Z}_q^{3m}$ to mask $\boldsymbol{e}_{0,1},\boldsymbol{e}_{0,2},\dots,\boldsymbol{e}_{0,k}$, and it can be checked that

$$\begin{aligned}&\boldsymbol{B}^*(\textstyle\sum_{r=1}^k\beta_r(\boldsymbol{e}'_{i,0,r}+\boldsymbol{r}'_{r,0},\boldsymbol{e}_{0,r}+\boldsymbol{r}_{0,r}))-\boldsymbol{b}_j\\&\quad=\boldsymbol{B}^*(\textstyle\sum_{r=1}^k\beta_r(\boldsymbol{r}'_{r,0},\boldsymbol{r}_{0,r}))\bmod q.\end{aligned}$$

3. Sample one permutation $\phi\in\mathcal{S}_{3m}$, and it can be checked that $\phi(\boldsymbol{e}_{0,r})\in\boldsymbol{B}_{3m}$.

Putting all the techniques together, we design an interactive Stern-type statistical ZKP protocol. In our lattice-based VLR-GS-BU, we also use a statistically hiding and computationally blinding commitment scheme (COM) proposed by Kawachi et al. (2008).

The protocol between $\mathcal{P}$ (a member id with an index i) and $\mathcal{V}$ is as follows:

Commitments: $\mathcal{P}$ picks the randomness of COM, $\theta_1,\theta_2,\theta_3\xleftarrow{\$}\{0,1\}^{2n\log q}$, and several objects:

$$\begin{cases}\boldsymbol{r}'_1,\boldsymbol{r}'_2,\dots,\boldsymbol{r}'_k\xleftarrow{\$}\mathbb{Z}_q^{(2\ell+2)\times(3m)},\\ \pi_1,\pi_2,\dots,\pi_k\xleftarrow{\$}\mathcal{S}_{3m},\\ \boldsymbol{r}_{0,1},\boldsymbol{r}_{0,2},\dots,\boldsymbol{r}_{0,k}\xleftarrow{\$}\mathbb{Z}_q^{3m},\\ \varphi_1,\varphi_2,\dots,\varphi_k\xleftarrow{\$}\mathcal{S}_{3m},\\ \phi_1,\phi_2,\dots,\phi_k\xleftarrow{\$}\mathcal{S}_{3m},\\ \tau\xleftarrow{\$}\mathcal{S}_{2\ell}.\end{cases}$$

Let $\boldsymbol{r}'_{r,0}=\mathrm{Parse}(\boldsymbol{r}'_r,1,m)$. $\mathcal{P}$ sends the commitment $\mathrm{CMT}=(\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{c}_3)$ to $\mathcal{V}$, where

$$\begin{cases}\boldsymbol{c}_1=&\mathrm{COM}(\{\pi_r,\varphi_r,\phi_r\}_{r=1}^k,\tau,\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{r}'_r),\\&\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{r}'_{r,0},\boldsymbol{r}_{0,r}));\theta_1),\\ \boldsymbol{c}_2=&\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r),\phi_r(\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_2),\\ \boldsymbol{c}_3=&\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{e}'_{i,r}+\boldsymbol{r}'_r),\\&\phi_r(\boldsymbol{e}_{0,r}+\boldsymbol{r}_{0,r})\}_{r=1}^k;\ \theta_3).\end{cases}$$

Challenge: $\mathcal{V}$ chooses a challenge $\mathrm{Ch}\xleftarrow{\$}\{1,2,3\}$, and sends it to $\mathcal{P}$.

Response: depending on Ch, $\mathcal{P}$ replies as follows:

1. Ch=1. Let $\boldsymbol{v}'_r=\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{e}'_{i,r})$, $\boldsymbol{v}_r=\phi_r(\boldsymbol{e}_{0,r})$, $\boldsymbol{w}'_r=\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r)$, $\boldsymbol{w}_r=\phi_r(\boldsymbol{r}_{0,r})$, and $\boldsymbol{t}_{\mathrm{id}}=\tau(\mathrm{id}^*)$. Define

$$\mathrm{RSP}=(\{\boldsymbol{v}'_r,\boldsymbol{w}'_r,\boldsymbol{v}_r,\boldsymbol{w}_r\}_{r=1}^k,\boldsymbol{t}_{\mathrm{id}},\theta_2,\theta_3).\tag{5}$$

2. Ch=2. Let $\hat{\pi}_r=\pi_r$, $\hat{\varphi}_r=\varphi_r$, $\hat{\phi}_r=\phi_r$, $\hat{\tau}=\tau$, $\boldsymbol{y}'_r=\boldsymbol{e}'_{i,r}+\boldsymbol{r}'_r$, and $\boldsymbol{y}_r=\boldsymbol{e}_{0,r}+\boldsymbol{r}_{0,r}$. Define

$$\mathrm{RSP}=(\{\hat{\pi}_r,\hat{\varphi}_r,\hat{\phi}_r,\boldsymbol{y}'_r,\boldsymbol{y}_r\}_{r=1}^k,\hat{\tau},\theta_1,\theta_3).\tag{6}$$

3. Ch=3. Let $\tilde{\pi}_r=\pi_r$, $\tilde{\varphi}_r=\varphi_r$, $\tilde{\phi}_r=\phi_r$, $\tilde{\tau}=\tau$, $\boldsymbol{h}'_r=\boldsymbol{r}'_r$, and $\boldsymbol{h}_r=\boldsymbol{r}_{0,r}$. Define

$$\mathrm{RSP}=(\{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\phi}_r,\boldsymbol{h}'_r,\boldsymbol{h}_r\}_{r=1}^k,\tilde{\tau},\theta_1,\theta_2).\tag{7}$$

Verification: after receiving RSP, $\mathcal{V}$ begins to check the following:

1. Ch=1. Check $\boldsymbol{t}_{\mathrm{id}}\in\boldsymbol{B}_{2\ell}$, $\boldsymbol{v}'_r\in\mathbf{SecExt}(\boldsymbol{t}_{\mathrm{id}})$, $\boldsymbol{v}_r\in\boldsymbol{B}_{3m}$, and

$$\begin{cases}\boldsymbol{c}_2=\mathrm{COM}(\{\boldsymbol{w}'_r,\boldsymbol{w}_r\}_{r=1}^k;\theta_2),\\ \boldsymbol{c}_3=\mathrm{COM}(\{\boldsymbol{v}'_r+\boldsymbol{w}'_r,\boldsymbol{v}_r+\boldsymbol{w}_r\}_{r=1}^k;\theta_3).\end{cases}$$

2. Ch=2. Let $\boldsymbol{y}'_{r,0}=\mathrm{Parse}(\boldsymbol{y}'_r,1,m)$, and check

$$\begin{cases}\boldsymbol{c}_1=\mathrm{COM}(\{\hat{\pi}_r,\hat{\varphi}_r,\hat{\phi}_r\}_{r=1}^k,\hat{\tau},\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{y}'_r)\\\qquad-\boldsymbol{u},\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{y}'_{r,0},\boldsymbol{y}_r))-\boldsymbol{b}_j;\theta_1),\\ \boldsymbol{c}_3=\mathrm{COM}(\{\mathcal{F}_{\hat{\pi}_r,\hat{\varphi}_r,\hat{\tau}}(\boldsymbol{y}'_r),\hat{\phi}_r(\boldsymbol{y}_r)\}_{r=1}^k;\theta_3).\end{cases}$$

3. Ch=3. Let $\boldsymbol{h}'_{r,0}=\mathrm{Parse}(\boldsymbol{h}'_r,1,m)$, and check

$$\begin{cases}\boldsymbol{c}_1=\mathrm{COM}(\{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\phi}_j\}_{r=1}^k,\tilde{\tau},\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{h}'_r),\\\qquad\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{h}'_{r,0},\boldsymbol{h}_r));\theta_1),\\ \boldsymbol{c}_2=\mathrm{COM}(\{\mathcal{T}_{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\tau}}(\boldsymbol{h}'_r),\tilde{\phi}_r(\boldsymbol{h}_r)\}_{r=1}^k;\theta_2).\end{cases}$$

Finally, $\mathcal{V}$ returns valid if all the conditions hold; otherwise, it returns invalid.

The associated relation $\mathcal{R}(n,k,\ell,t,q,m,\beta)$ in the above protocol is defined as

$$\mathcal{R}=\begin{cases}\boldsymbol{A},\boldsymbol{A}_0,\boldsymbol{A}_1,\boldsymbol{B},\boldsymbol{B}_0,\boldsymbol{B}_1\in\mathbb{Z}_q^{n\times m},\boldsymbol{u}\in\mathbb{Z}_q^n,\\ \mathrm{id}=\mathrm{Bin}(i),j\in\{1,2,\dots,t\},\boldsymbol{b}_j\in\mathbb{Z}_q^m,\\ \boldsymbol{e}'_i=(\boldsymbol{e}'_{i,0},\boldsymbol{e}'_{i,1},\mathrm{Bin}(i)\otimes\boldsymbol{e}'_{i,1})\in\mathbf{Sec}_\beta(\mathrm{id}),\\ \boldsymbol{e}_0\in\mathbb{Z}^m\text{ s.t. }0<\|\boldsymbol{e}'_i\|_\infty,\|\boldsymbol{e}_0\|_\infty\le\beta,\\ \boldsymbol{b}_j=(\boldsymbol{B}^{\mathrm{T}}\hat{\boldsymbol{B}}_j)\boldsymbol{e}'_{i,0}+\boldsymbol{e}_0\bmod q,\\ [\boldsymbol{A}|\boldsymbol{A}_0|\boldsymbol{g}_\ell\otimes\boldsymbol{A}_1]\boldsymbol{e}'_i=\boldsymbol{u}\bmod q.\end{cases}$$

3.3 Analysis of the protocol

We summarize the main properties of the above protocol in the following theorem, including communication cost, perfect completeness, statistical ZK, and argument of knowledge:

Theorem 1 Let COM be a statistically hiding and computationally binding commitment scheme. Then the proposed protocol is a statistical ZK argument of knowledge for the relation $\mathcal{R}(n,k,\ell,t,q,m,\beta)$, where each round has perfect completeness, soundness error of 2/3, the argument of knowledge property, and communication cost $\ell\widetilde{\mathcal{O}}(n)$.

Proof The proof includes the following aspects:

Communication cost:

1. The output of COM, a vector of $\mathbb{Z}_q^n$, has bit-sizes $n\log q$, and thus $\mathcal{P}$ sends three commitments amounting to $3n\log q$ bits.

2. The challenge Ch can be represented by 2 bits.

3. The response RSP from $\mathcal{P}$ consists of:

(1) one permutation in $\mathcal{S}_{2\ell}$;

(2) $3k$ permutations in $\mathcal{S}_{3m}$;

(3) $2k$ vectors in $\mathbb{Z}_q^{(2\ell+2)3m}$, $2k$ vectors in $\mathbb{Z}_q^{3m}$, and one vector in $\{0,1\}^{2\ell}$.

So, the bit-size of RSP is bounded by $\mathcal{O}(\ell mk)\log q$. Recall that $k=\lfloor\log\beta\rfloor+1$. The communication cost of the proposed Stern-type statistical ZKP protocol is bounded by $\mathcal{O}(\ell m\log\beta)\log q=\ell\widetilde{\mathcal{O}}(n)$.

Perfect completeness:

Given a public tuple $(\boldsymbol{A},\boldsymbol{A}_0,\boldsymbol{A}_1,\boldsymbol{u},\boldsymbol{B},\boldsymbol{B}_0,\boldsymbol{B}_1,j,\boldsymbol{b}_j)$, if an honest $\mathcal{P}$ has witness $=(\mathrm{id}=\mathrm{Bin}(i),\boldsymbol{e}'_i\in\mathbf{Sec}_\beta(\mathrm{id}),\boldsymbol{e}_0\in\mathbb{Z}^m)$ and follows the proposed protocol, it can generate an efficient Stern-type statistical ZKP satisfying the verification process, and is accepted by $\mathcal{V}$ with a high probability.

The inputs and witness are transformed into $\boldsymbol{A}^*$, $\boldsymbol{B}^*$, id^*, $\{\boldsymbol{e}'_{i,r},\boldsymbol{e}_{0,r},\boldsymbol{e}'_{i,0,r}=\mathrm{Parse}(\boldsymbol{e}'_{i,r},1,m)\}_{r=1}^k$ by $\mathcal{P}$ using the Dec, Ext, and Mat-Ext techniques; thus, these results satisfy the following new structures:

$$\boldsymbol{A}^*(\textstyle\sum_{r=1}^k\beta_r\boldsymbol{e}'_{i,r})=\boldsymbol{u}\bmod q,\ \boldsymbol{e}'_{i,r}\in\mathbf{SecExt}(\mathrm{id}^*),$$
$$\boldsymbol{B}^*(\textstyle\sum_{r=1}^k\beta_r(\boldsymbol{e}'_{i,0,r},\boldsymbol{e}_{0,r}))=\boldsymbol{b}_j\bmod q,\ \boldsymbol{e}_{0,r}\in\boldsymbol{B}_{3m}.$$

As in Zhang et al. (2019a, 2019b), to show that $\mathcal{P}$ can pass all verification checks correctly for each $\mathrm{Ch}\in\{1,2,3\}$ with a high probability without considering the checks for correct computations, we need only to note that:

1. Ch=1. Since $\mathrm{id}=\mathrm{Bin}(i)\in\{0,1\}^\ell$, $\mathrm{id}^*\in\boldsymbol{B}_{2\ell}$ is an extension of id, $\boldsymbol{B}_{2\ell}$ is invariant under $\tau\in\mathcal{S}_{2\ell}$ and $\boldsymbol{t}_{\mathrm{id}}=\tau(\mathrm{id}^*)\in\boldsymbol{B}_{2\ell}$. Similarly, for $r\in\{1,2,\dots,k\}$, $\boldsymbol{e}_{0,r}\in\boldsymbol{B}_{3m}$, $\boldsymbol{B}_{3m}$ is invariant under $\phi_r\in\mathcal{S}_{3m}$; thus, $\boldsymbol{v}_r=\phi_r(\boldsymbol{e}_{0,r})\in\boldsymbol{B}_{3m}$. Thus, we have

$$\boldsymbol{v}'_r=\mathcal{F}_{\pi_r,\varphi_r,\tau,}(\boldsymbol{e}'_{i,r})\in\mathbf{SecExt}(\tau(\mathrm{id}^*)=\boldsymbol{t}_{\mathrm{id}}).$$

2. Ch=2. The key point is to check $\boldsymbol{c}_1$. $\mathcal{P}$ can pass this step by generating $\boldsymbol{y}'_r$, $\boldsymbol{r}'_r$, $\boldsymbol{y}_r$, $\boldsymbol{y}'_{r,0}$, $\boldsymbol{r}'_{r,0}$, and $\boldsymbol{r}_{0,r}$, such that the following hold true:

$$\begin{aligned}&\boldsymbol{A}^*(\textstyle\sum_{r=1}^k\beta_r\boldsymbol{y}'_r)-\boldsymbol{u}\\&\quad=\boldsymbol{A}^*(\textstyle\sum_{r=1}^k\beta_r(\boldsymbol{e}'_{i,r}+\boldsymbol{r}'_r))-\boldsymbol{u}\\&\quad=\boldsymbol{A}^*(\textstyle\sum_{r=1}^k\beta_r\boldsymbol{r}'_r)\bmod q,\\&\boldsymbol{B}^*(\textstyle\sum_{r=1}^k\beta_r(\boldsymbol{y}'_{r,0},\boldsymbol{y}_r))-\boldsymbol{b}_j\\&\quad=\boldsymbol{B}^*(\textstyle\sum_{j=1}^k\beta_j(\boldsymbol{e}'_{i,0,r}+\boldsymbol{r}'_{r,0},\boldsymbol{e}_{0,r}+\boldsymbol{r}_{0,r}))-\boldsymbol{b}_j\\&\quad=\boldsymbol{B}^*(\textstyle\sum_{r=1}^k\beta_r(\boldsymbol{r}'_{r,0},\boldsymbol{r}_{0,r}))\bmod q.\end{aligned}$$

3. Ch=3. We need to consider the checks for correct computations, and obviously these are true.

According to the previous discussions, the proposed protocol enjoys perfect completeness.

Statistical ZK:

As in Zhang et al. (2019a, 2019b), we should design a PPT simulator $\hat{\mathcal{S}}$ that interacts with verifier $\mathcal{V}'$ (may be dishonest) to output a simulated transcript that is statistically close to the one generated by honest $\mathcal{P}$ in the real interaction with a probability negligibly close to 2/3. The design is as follows:

$\hat{\mathcal{S}}$ first samples a random value $\widetilde{\mathrm{Ch}}\xleftarrow{\$}\{1,2,3\}$ as a prediction that verifier $\mathcal{V}'$ may not choose.

1. If $\widetilde{\mathrm{Ch}}=1$, the following steps are taken:

(1) Use the basic linear algebra (BLA) algorithm to compute vectors $\boldsymbol{e}''_{i,1},\boldsymbol{e}''_{i,2},\dots,\boldsymbol{e}''_{i,k}\in\mathbb{Z}_q^{(2\ell+1)3m}$ s.t. $\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{e}''_{i,r})=\boldsymbol{u}\bmod q$.

(2) Define $\boldsymbol{e}''_{i,0,r}=\mathrm{Parse}(\boldsymbol{e}''_{i,r},1,m)$ and use the BLA to compute k vectors $\widehat{\boldsymbol{e}}_{0,1},\widehat{\boldsymbol{e}}_{0,2},\dots,\widehat{\boldsymbol{e}}_{0,k}\in\mathbb{Z}_q^{3m}$ s.t. $\boldsymbol{b}_j=\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{e}''_{i,0,r},\widehat{\boldsymbol{e}}_{0,r}))\bmod q$.

(3) Sample the randomness of COM, θ_1, θ_2, θ_3, and several random vectors and permutations:

$$\begin{cases}\boldsymbol{r}'_1,\boldsymbol{r}'_2,\dots,\boldsymbol{r}'_k\in\mathbb{Z}_q^{(2\ell+2)\times(3m)},\\ \boldsymbol{r}_{0,1},\boldsymbol{r}_{0,2},\dots,\boldsymbol{r}_{0,k}\in\mathbb{Z}_q^{3m},\\ \pi_1,\pi_2,\dots,\pi_k,\varphi_1,\varphi_2\dots,\varphi_k,\phi_1,\phi_2,\dots,\phi_k\in\mathcal{S}_{3m},\\ \tau\in\mathcal{S}_{2\ell}.\end{cases}$$

(4) Let $\boldsymbol{r}'_{r,0}=\mathrm{Parse}(\boldsymbol{r}'_r,1,m)$, and compute $\mathrm{CMT}=(\boldsymbol{c}'_1,\boldsymbol{c}'_2,\boldsymbol{c}'_3)$, where

$$\begin{cases}\boldsymbol{c}'_1=\mathrm{COM}(\{\pi_r,\varphi_r,\phi_r\}_{r=1}^k,\tau,\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{r}'_r),\\ \qquad\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{r}'_{r,0},\boldsymbol{r}_{0,r}));\theta_1),\\ \boldsymbol{c}'_2=\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r),\phi_r(\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_2),\\ \boldsymbol{c}'_3=\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{e}''_{i,r}+\boldsymbol{r}'_r),\\ \qquad\phi_r(\boldsymbol{e}_{0,r}+\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_3).\end{cases}$$

(5) Send CMT to $\mathcal{V}'$.

After receiving $\mathrm{Ch}\in\{1,2,3\}$, $\hat{\mathcal{S}}$ replies as follows:

(1) If Ch=1, output $\perp$ and abort.

(2) If Ch=2, send

$$\mathrm{RSP}=(\{\pi_r,\varphi_r,\phi_r,\boldsymbol{e}''_{i,r}+\boldsymbol{r}'_r,\widehat{\boldsymbol{e}}_{0,r}+\boldsymbol{r}_{0,r}\}_{r=1}^k,\tau\theta_1,\theta_3).$$

(3) If Ch=3, send

$\mathrm{RSP}=(\{\pi_r,\varphi_r,\phi_r,\boldsymbol{r}'_r,\boldsymbol{r}_{0,r}\}_{r=1}^k,\tau,\theta_1,\theta_2)$.

2. If $\widetilde{\mathrm{Ch}}=2$, the following steps are taken:

(1) Sample the randomness of COM, θ_1, θ_2, θ_3, and several random vectors and permutations:

$$\begin{cases}\boldsymbol{r}'_1,\boldsymbol{r}'_2,\ldots,\boldsymbol{r}'_k\in\mathbb{Z}_q^{(2\ell+2)\times(3m)},\\ \boldsymbol{r}_{0,1},\boldsymbol{r}_{0,2},\ldots,\boldsymbol{r}_{0,k}\in\mathbb{Z}_q^{3m},\\ \pi_1,\pi_2,\ldots,\pi_k,\ \varphi_1,\varphi_2,\ldots,\varphi_k,\ \phi_1,\phi_2,\ldots,\phi_k\in\mathcal{S}_{3m},\\ \tau\in\mathcal{S}_{2\ell},\\ \widehat{\boldsymbol{e}}_{0,1},\widehat{\boldsymbol{e}}_{0,2},\ldots,\widehat{\boldsymbol{e}}_{0,k}\in\boldsymbol{B}_{3m},\\ \mathrm{id}'\in\boldsymbol{B}_{2\ell},\\ \boldsymbol{e}''_{i,1},\boldsymbol{e}''_{i,2},\ldots,\boldsymbol{e}''_{i,k}\in\mathbf{SecExt}(\mathrm{id}').\end{cases}$$

(2) Let $\boldsymbol{r}'_{r,0}=\mathrm{Parse}(\boldsymbol{r}'_r,1,m)$, and compute $\mathrm{CMT}=(\boldsymbol{c}'_1,\boldsymbol{c}'_2,\boldsymbol{c}'_3)$, where

$$\begin{cases}\boldsymbol{c}'_1=\mathrm{COM}(\{\pi_r,\varphi_r,\phi_r\}_{r=1}^k,\tau,\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{r}_r),\\ \qquad\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{r}'_{r,0},\boldsymbol{r}_{0,r}));\theta_1),\\ \boldsymbol{c}'_2=\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r),\phi_r(\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_2),\\ \boldsymbol{c}'_3=\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{e}''_{i,r}+\boldsymbol{r}'_r),\\ \qquad\phi_r(\widehat{\boldsymbol{e}}_{0,r}+\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_3).\end{cases}$$

(3) Send CMT to $\mathcal{V}'$.

After receiving $\mathrm{Ch}\in\{1,2,3\}$, $\hat{\mathcal{S}}$ replies as follows:

(1) If $\mathrm{Ch}=1$, send

$$\begin{aligned}\mathrm{RSP}=(&\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{e}''_{i,r}),\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r),\phi_r(\widehat{\boldsymbol{e}}_{0,r}),\\ &\phi_r(\boldsymbol{r}_{0,r})\}_{r=1}^k,\tau(\mathrm{id}'),\theta_2,\theta_3).\end{aligned}$$

(2) If $\mathrm{Ch}=2$, output $\perp$ and abort.

(3) If $\mathrm{Ch}=3$, send

$\mathrm{RSP}=(\{\pi_r,\varphi_r,\phi_r,\boldsymbol{r}'_r,\boldsymbol{r}_{0,r}\}_{r=1}^k,\tau,\theta_1,\theta_2)$.

3. If $\widetilde{\mathrm{Ch}}=3$, the following steps are taken:

(1) Sample the randomness of COM, θ_1, θ_2, θ_3, and several random vectors and permutations:

$$\begin{cases}\boldsymbol{r}'_1,\boldsymbol{r}'_2,\ldots,\boldsymbol{r}'_k\in\mathbb{Z}_q^{(2\ell+2)\times(3m)},\\ \boldsymbol{r}_{0,1},\boldsymbol{r}_{0,2},\ldots,\boldsymbol{r}_{0,k}\in\mathbb{Z}_q^{3m},\\ \pi_1,\pi_2,\ldots,\pi_k,\ \varphi_1,\varphi_2,\ldots,\varphi_k,\ \phi_1,\phi_2,\ldots,\phi_k\in\mathcal{S}_{3m},\\ \tau\in\mathcal{S}_{2\ell},\\ \widehat{\boldsymbol{e}}_{0,1},\widehat{\boldsymbol{e}}_{0,2},\ldots,\widehat{\boldsymbol{e}}_{0,k}\in\boldsymbol{B}_{3m},\\ \mathrm{id}'\in\boldsymbol{B}_{2\ell},\\ \boldsymbol{e}''_{i,1},\boldsymbol{e}''_{i,2},\ldots,\boldsymbol{e}''_{i,k}\in\mathbf{SecExt}(\mathrm{id}').\end{cases}$$

(2) Let $\boldsymbol{e}''_{i,0,r}=\mathrm{Parse}(\boldsymbol{e}''_{i,r},1,m)$, $\boldsymbol{r}'_{r,0}=\mathrm{Parse}(\boldsymbol{r}'_r,1,m)$, and set $\mathrm{CMT}=(\boldsymbol{c}'_1,\boldsymbol{c}'_2,\boldsymbol{c}'_3)$:

$$\begin{cases}\boldsymbol{c}'_1=\mathrm{COM}(\{\pi_r,\varphi_r,\phi_r\}_{r=1}^k,\tau,\\ \qquad\boldsymbol{A}^*(\sum_{r=1}^k\beta_r(\boldsymbol{e}''_{i,r}+\boldsymbol{r}'_r))-\boldsymbol{u},\\ \qquad\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{e}''_{i,0,r}+\boldsymbol{r}'_{r,0},\widehat{\boldsymbol{e}}_{0,r}+\boldsymbol{r}_{0,r}))\\ \qquad-\boldsymbol{b}_j;\theta_1),\\ \boldsymbol{c}'_2=\mathrm{COM}(\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r),\phi_j(\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_2),\\ \boldsymbol{c}'_3=\mathrm{COM}(\{\mathcal{F}_{\pi_j,\varphi_r,\tau}(\boldsymbol{e}''_{i,r}+\boldsymbol{r}'_r),\\ \qquad\phi_r(\widehat{\boldsymbol{e}}_{0,r}+\boldsymbol{r}_{0,r})\}_{r=1}^k;\theta_3).\end{cases}$$

(3) Send CMT to $\mathcal{V}'$.

After receiving $\mathrm{Ch}\in\{1,2,3\}$, $\hat{\mathcal{S}}$ replies as follows:

(1) If $\mathrm{Ch}=1$, send

$$\begin{aligned}\mathrm{RSP}=(&\{\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{e}''_{i,r}),\mathcal{F}_{\pi_r,\varphi_r,\tau}(\boldsymbol{r}'_r),\phi_r(\widehat{\boldsymbol{e}}_{0,r}),\\ &\phi_r(\boldsymbol{r}_{0,r})\}_{r=1}^k,\tau(\mathrm{id}'),\theta_2,\theta_3).\end{aligned}$$

(2) If $\mathrm{Ch}=2$, send

$$\begin{aligned}\mathrm{RSP}=(&\{\pi_r,\varphi_r,\phi_r,\boldsymbol{e}''_{i,r}+\boldsymbol{r}'_r,\widehat{\boldsymbol{e}}_{0,r}+\boldsymbol{r}_{0,r}\}_{r=1}^k,\\ &\tau,\theta_1,\theta_3).\end{aligned}$$

(3) If $\mathrm{Ch}=3$, output $\perp$ and abort.

Based on a statistically hiding property of COM, the distributions of CMT, Ch, and RSP are statistically close to those in the real interaction, and $\hat{\mathcal{S}}$ outputs $\perp$ and aborts with a probability negligibly close to 1/3. Furthermore, once $\hat{\mathcal{S}}$ does not halt, a valid transcript will be given, and the distribution of the transcript is statistically close to that in the real interaction, so $\hat{\mathcal{S}}$ can impersonate an honest $\mathcal{P}$ with a probability negligibly close to 2/3.

Argument of knowledge:

To prove that the proposed protocol is an argument of knowledge for the relation $\mathcal{R}(n,k,\ell,t,q,m,\beta)$, we need to prove that the given protocol satisfies the special soundness property.

If there is a $\mathcal{P}'$ (may be cheating) who can respond to three challenges correctly corresponding to the same commitment CMT with the inputs $\Delta=(\boldsymbol{A},\boldsymbol{A}_0,\boldsymbol{A}_1,\boldsymbol{B},\boldsymbol{B}_0,\boldsymbol{B}_1,\boldsymbol{u},j,\boldsymbol{b}_j)$, then there is an extractor $\mathcal{K}$ who can produce

$$(\mathrm{id}=\mathrm{Bin}(i),\boldsymbol{e}'_i=(\boldsymbol{e}'_{i,0},\boldsymbol{e}'_{i,1},\mathrm{Bin}(i)\otimes\boldsymbol{e}'_{i,1}),\boldsymbol{e}_0)$$

s.t. $(\Delta;\mathrm{id}=\mathrm{Bin}(i),\boldsymbol{e}'_i,\boldsymbol{e}_0)\in\mathcal{R}$.

Indeed, based on three valid RSP_1, RSP_2, and RSP_3 given by $\mathcal{P}'$, the extractor $\mathcal{K}$ can extract

$$\begin{cases}\boldsymbol{t}_{\mathrm{id}}\in\boldsymbol{B}_{2\ell},\\ \boldsymbol{v}'_r\in\mathbf{SecExt}(\boldsymbol{t}_{\mathrm{id}}),\\ \boldsymbol{v}_r\in\boldsymbol{B}_{3m},\\ \boldsymbol{c}_1=\mathrm{COM}(\{\hat{\pi}_r,\hat{\varphi}_r,\hat{\phi}_r\}_{r=1}^k,\hat{\tau},\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{y}'_r)\\ \quad-\boldsymbol{u},\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{y}'_{r,0},\boldsymbol{y}_r))-\boldsymbol{b}_j;\theta_1)\\ \quad=\mathrm{COM}(\{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\phi}_r\}_{r=1}^k,\tilde{\tau},\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{h}'_r),\\ \quad\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{h}'_{r,0},\boldsymbol{h}_r));\theta_1),\\ \boldsymbol{c}_2=\mathrm{COM}(\{\boldsymbol{w}'_r,\boldsymbol{w}_r\}_{r=1}^k;\theta_2)\\ \quad=\mathrm{COM}(\{\mathcal{F}_{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\tau}}(\boldsymbol{h}'_r),\tilde{\phi}_r(\boldsymbol{h}_r)\}_{r=1}^k;\theta_2),\\ \boldsymbol{c}_3=\mathrm{COM}(\{\boldsymbol{v}'_r+\boldsymbol{w}'_r,\boldsymbol{v}_r+\boldsymbol{w}_r\}_{r=1}^k;\theta_3)\\ \quad=\mathrm{COM}(\{\mathcal{F}_{\hat{\pi}_r,\hat{\varphi}_r,\hat{\tau}}(\boldsymbol{y}'_r),\hat{\phi}_r(\boldsymbol{y}_r)\}_{r=1}^k;\theta_3).\end{cases}$$

Based on the computationally binding property of COM, the extractor $\mathcal{K}$ can deduce

$$\begin{cases}\boldsymbol{t}_{\mathrm{id}}\in\boldsymbol{B}_{2\ell},\hat{\tau}=\tilde{\tau},\hat{\phi}_r=\tilde{\phi}_r,\hat{\pi}_r=\tilde{\pi}_r,\hat{\varphi}_r=\tilde{\varphi}_r,\\ \boldsymbol{A}^*\left(\sum_{r=1}^k\beta_r\boldsymbol{y}'_r\right)-\boldsymbol{u}=\boldsymbol{A}^*\left(\sum_{r=1}^k\beta_r\boldsymbol{h}'_r\right)\bmod q,\\ \boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{y}'_{r,0},\boldsymbol{y}_r))-\boldsymbol{b}_j\\ \quad=\boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{h}'_{r,0},\boldsymbol{h}_r))\bmod q,\\ \boldsymbol{w}'_r=\mathcal{F}_{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\tau}}(\boldsymbol{h}'_r),\boldsymbol{v}'_r+\boldsymbol{w}'_r=\mathcal{F}_{\hat{\pi}_r,\hat{\varphi}_r,\hat{\tau}}(\boldsymbol{y}'_r),\\ \boldsymbol{v}'_r\in\mathbf{SecExt}(\boldsymbol{t}_{\mathrm{id}}),\boldsymbol{v}_r\in\boldsymbol{B}_{3m},\\ \boldsymbol{w}_r=\tilde{\phi}_r(\boldsymbol{h}_r),\boldsymbol{v}_r+\boldsymbol{w}_r=\hat{\phi}_r(\boldsymbol{y}_r).\end{cases}$$

Let $\boldsymbol{e}'_{i,r}=\boldsymbol{y}'_r-\boldsymbol{h}'_r=\mathcal{T}^{-1}_{\tilde{\pi}_r,\tilde{\varphi}_r,\tilde{\tau}}(\boldsymbol{v}'_r)$, $\boldsymbol{e}_{0,r}=\boldsymbol{y}_r-\boldsymbol{h}_r=\tilde{\phi}_r^{-1}(\boldsymbol{v}_r)$; thus, $\boldsymbol{e}'_{i,r}\in\mathbf{SecExt}(\tilde{\tau}^{-1}(\boldsymbol{t}_{\mathrm{id}})=\mathrm{id}^*)$, $\boldsymbol{e}_{0,r}\in\boldsymbol{B}_{3m}$. Let $\boldsymbol{e}'_{i,0,r}=\mathrm{Parse}(\boldsymbol{e}'_{i,r},1,m)$, we have

$$\begin{cases}\boldsymbol{A}^*(\sum_{r=1}^k\beta_r\boldsymbol{e}'_{i,r})=\boldsymbol{u}\bmod q,\\ \boldsymbol{B}^*(\sum_{r=1}^k\beta_r(\boldsymbol{e}'_{i,0,r},\boldsymbol{e}_{0,r}))=\boldsymbol{b}_j\bmod q.\end{cases}$$

The extractor $\mathcal{K}$ produces $\mathrm{id}=\mathrm{Bin}(i)\in\{0,1\}^\ell$, $\boldsymbol{e}'_i\in\mathbf{Sec}_\beta(\mathrm{id})$, and $\boldsymbol{e}_0\in\mathbb{Z}^m$ as follows:

1. Let $\mathrm{id}^*=(d_1,d_2,\ldots,d_\ell,\ldots,d_{2\ell})=\tilde{\tau}^{-1}(\boldsymbol{t}_{\mathrm{id}})$. We obtain $\mathrm{Bin}(i)=\mathrm{id}=(d_1,d_2,\ldots,d_\ell)$, and the index $i=\boldsymbol{g}_\ell^{\mathrm{T}}\mathrm{Bin}(i)$.

2. Let $\boldsymbol{e}_i^*=\sum_{r=1}^k\beta_r\boldsymbol{e}'_{i,r}$, we have

$$0<\|\boldsymbol{e}_i^*\|_\infty\le\textstyle\sum_{r=1}^k\beta_r\|\boldsymbol{e}'_{i,r}\|_\infty\le\beta.$$

Since $\boldsymbol{e}'_{i,r}\in\mathbf{SecExt}(\mathrm{id}^*)$, there are two vectors $\boldsymbol{e}^*_{i,0}$ and $\boldsymbol{e}^*_{i,1}\in\mathbb{Z}^{3m}$ that satisfy $\|\boldsymbol{e}^*_{i,0}\|_\infty,\|\boldsymbol{e}^*_{i,1}\|_\infty\le\beta$, and $\boldsymbol{e}_i^*=(\boldsymbol{e}^*_{i,0},\boldsymbol{e}^*_{i,1},d_1\boldsymbol{e}^*_{i,1},\ldots,d_{2\ell}\boldsymbol{e}^*_{i,1})$. Set

$$\begin{aligned}\boldsymbol{e}'_i&=(\boldsymbol{e}'_{i,0},\boldsymbol{e}'_{i,1},d_1\boldsymbol{e}'_{i,1},\ldots,d_\ell\boldsymbol{e}'_{i,1})\\&=(\boldsymbol{e}'_{i,0},\boldsymbol{e}'_{i,1},\mathrm{Bin}(i)\otimes\boldsymbol{e}'_{i,1}),\end{aligned}$$

where $\boldsymbol{e}'_{i,0}$ and $\boldsymbol{e}'_{i,1}$ are obtained from $\boldsymbol{e}^*_{i,0}$ and $\boldsymbol{e}^*_{i,1}$, respectively, by removing the last $2m$ coordinates. Thus, $\boldsymbol{e}'_i\in\mathbf{Sec}_\beta(\mathrm{id})$ and

$$[\boldsymbol{A}|\boldsymbol{A}_0|\boldsymbol{g}_\ell\otimes\boldsymbol{A}_1](\boldsymbol{e}'_{i,0},\boldsymbol{e}'_{i,1},\mathrm{Bin}(i)\otimes\boldsymbol{e}'_{i,1})=\boldsymbol{u}\bmod q.$$

3. Let $\hat{\boldsymbol{e}}_0=\sum_{r=1}^k\beta_r\boldsymbol{e}_{0,r}$, we have

$$0<\|\hat{\boldsymbol{e}}_0\|_\infty\le\textstyle\sum_{r=1}^k\beta_r\|\boldsymbol{e}_{0,r}\|_\infty\le\beta.$$

Let $\boldsymbol{e}_0\in\mathbb{Z}^m$ be obtained from $\hat{\boldsymbol{e}}_0$ by removing the last $2m$ coordinates. Thus, $\boldsymbol{e}_0\in\mathbb{Z}^m$, $0<\|\boldsymbol{e}_0\|_\infty\le\beta$, and $\boldsymbol{b}_j=(\boldsymbol{B}^{\mathrm{T}}\hat{\boldsymbol{B}}_j)\boldsymbol{e}'_i+\boldsymbol{e}_0\bmod q$.

Finally, $\mathcal{K}$ outputs

$$\mathrm{witness}=(\mathrm{Bin}(i)=\mathrm{id},\boldsymbol{e}'_i\in\mathbf{Sec}_\beta(\mathrm{id}),\boldsymbol{e}_0\in\mathbb{Z}^m),$$

which is a valid witness for $\mathcal{R}=(n,k,\ell,t,m,\beta,p,t)$.

4 VLR-GS-BU scheme

In this section, we describe a lattice-based VLR-GS-BU scheme and prove the construction satisfying three requirements, correctness, BU-anonymity, and traceability, as defined in Section 2.1. The parameters will also be specified.

4.1 Description of the scheme

$\mathrm{KeyGen}(1^n,N,t)$: input a parameter n, group size $N=2^\ell=\mathrm{poly}(n)$, and number of periods $t=\mathrm{poly}(n)$; other parameters are as listed in Table 2. This algorithm works as follows:

1. Run $\mathrm{TrapGen}(q,n,m)$ to obtain $\boldsymbol{A}\in\mathbb{Z}_q^{n\times m}$ and a trapdoor $\boldsymbol{R}_{\boldsymbol{A}}$.

2. Choose matrices $\boldsymbol{A}_0,\boldsymbol{A}_1,\boldsymbol{B}_0,\boldsymbol{B}_1\xleftarrow{\$}\mathbb{Z}_q^{n\times m}$ and a vector $\boldsymbol{u}\xleftarrow{\$}\mathbb{Z}_q^n$.

3. As in Zhang et al. (2019a, 2019b), for id with $i \in \{0,1,\ldots,N-1\}$, let $\boldsymbol{A}_{\mathrm{id}} = [\boldsymbol{A}|\boldsymbol{A}_0 + i\boldsymbol{A}_1] \in \mathbb{Z}_q^{n\times 2m}$, and proceed as follows:

(1) Choose $\boldsymbol{e}_{i,1} \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^m,s}$, let $\boldsymbol{u}_i = (\boldsymbol{A}_0 + i\boldsymbol{A}_1)\boldsymbol{e}_{i,1}$, and run SamplePre$(\boldsymbol{A}, \boldsymbol{R}_{\boldsymbol{A}}, \boldsymbol{u} - \boldsymbol{u}_i, s)$ to obtain $\boldsymbol{e}_{i,0} \in \mathbb{Z}^m$.

(2) Let $\boldsymbol{e}_i = (\boldsymbol{e}_{i,0}, \boldsymbol{e}_{i,1}) \in \mathbb{Z}^{2m}$, $\boldsymbol{A}_{\mathrm{id}}\boldsymbol{e}_i = \boldsymbol{u} \bmod q$, and $0 < \|\boldsymbol{e}_i\|_\infty \le \beta$.

(3) For the time-period $\mathrm{TP}_{j\in\{1,2,\ldots,t\}} \in \mathbb{Z}_q^n$, define $\mathbf{grt}_{i,j} = (\boldsymbol{B}_0 + \mathcal{H}_1(\mathrm{TP}_j)\boldsymbol{B}_1)\boldsymbol{e}_{i,0} \bmod q$.

(4) Let the signing secret-key of member id be $\mathbf{gsk}_i = \boldsymbol{e}_i \in \mathbb{Z}^{2m}$, and the revocation token be $\mathbf{grt}_i = \{\mathbf{grt}_{i,1}, \mathbf{grt}_{i,2}, \ldots, \mathbf{grt}_{i,t}\}$.

4. Output:

Gpk $= (\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{u}, \mathcal{G}, \mathcal{H}_1, \mathcal{H}_2)$,

Gsk $= (\mathbf{gsk}_0, \mathbf{gsk}_1, \ldots, \mathbf{gsk}_{N-1})$,

Grt $= (\mathbf{grt}_0, \mathbf{grt}_1, \ldots, \mathbf{grt}_{N-1})$.

Sign(Gpk, $j, \mathbf{gsk}_i, \boldsymbol{m}$): let $\chi \in \mathbb{Z}$ be a β-bounded distribution. Take Gpk, current period j, and $\boldsymbol{m} \in \{0,1\}^*$ as inputs; a member id with index i and secret-key $\mathbf{gsk}_i = \boldsymbol{e}_i$ proceeds as follows:

1. Choose $\boldsymbol{v} \xleftarrow{\$} \{0,1\}^n$, and let $\boldsymbol{B} = \mathcal{G}(\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{u}, \boldsymbol{m}, \boldsymbol{v}) \in \mathbb{Z}_q^{n\times m}$.

2. Choose $\boldsymbol{e}_0 \xleftarrow{\$} \chi^m$, and define

$$\begin{aligned}\boldsymbol{b}_j &= \boldsymbol{B}^{\mathrm{T}}\mathbf{grt}_{i,j} + \boldsymbol{e}_0\\ &= (\boldsymbol{B}^{\mathrm{T}}(\boldsymbol{B}_0 + \mathcal{H}_1(\mathrm{TP}_j)\boldsymbol{B}_1))\boldsymbol{e}_{i,0} + \boldsymbol{e}_0.\end{aligned}$$

3. Generate a ZKP protocol in which the signer id is a valid member. This is achieved by repeating the protocol in Section 3.2 $\omega(\log n)$ times with $\Delta = (\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{u}, \boldsymbol{B}, \boldsymbol{B}_0, \boldsymbol{B}_1, j, \boldsymbol{b}_j)$ and a witness $(\mathrm{id}, \mathbf{gsk}_i, \boldsymbol{e}_0)$, and making it non-interactive as $\Pi = (\{\mathrm{CMT}_r\}_{r=1}^{\kappa}, \mathrm{CH}, \{\mathrm{RSP}_r\}_{r=1}^{\kappa})$, where $\mathrm{CH} = \{\mathrm{Ch_r}\}_{\mathrm{r=1}}^{\kappa} = \mathcal{H}_2(\boldsymbol{m}, \Delta)$.

4. Output $\sigma = (\boldsymbol{m}, j, \Pi, \boldsymbol{v}, \boldsymbol{b}_j)$.

Verify(Gpk, $j, \mathrm{RL}_j, \boldsymbol{m}, \sigma$): input Gpk, a signature σ on $\boldsymbol{m} \in \{0,1\}^*$, and a set of tokens $\mathrm{RL}_j = \{\mathbf{grt}_{i',j}, \mathbf{grt}_{i',j+1}, \ldots, \mathbf{grt}_{i',t}\}_{i'\le N-1} \subseteq$ Grt for time-period j; the verifier proceeds as follows:

1. Parse $\sigma = (\boldsymbol{m}, j, \Pi, \boldsymbol{v}, \boldsymbol{b}_j)$.

2. Compute $\boldsymbol{B} = \mathcal{G}(\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{u}, \boldsymbol{m}, \boldsymbol{v})$.

3. If $\mathrm{CH} \ne \mathcal{H}_2(\boldsymbol{m}, \Delta)$, return invalid.

4. For $r = 1$ to κ, run the verification steps of the protocol as in Section 3.2 to check the validity of RSP_r with respect to CMT_r and Ch_r. If any condition does not hold, return invalid.

5. For each $\mathbf{grt}_{i',j} \in \mathrm{RL}_j$, compute $\boldsymbol{e}_{i'} = \boldsymbol{b}_j - \boldsymbol{B}^{\mathrm{T}}\mathbf{grt}_{i',j} \bmod q$. If there exists an index $i' \le N-1$ such that $\|\boldsymbol{e}_{i'}\|_\infty \le \beta$, then return invalid.

6. Return valid.

4.2 Analysis of the scheme

Efficiency: we first analyze the space complexity of our lattice-based VLR-GS-BU scheme with respect to the security parameter n.

Gpk needs only $(\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1)$ and a vector $\boldsymbol{u}$ for identity-encoding, two matrices $(\boldsymbol{B}_0, \boldsymbol{B}_1)$ and an FRD function $\mathcal{H}_1$ for the new RT design, and two hash functions $\mathcal{G}$, $\mathcal{H}_2$ modeled as random oracles. So, the bit-size of Gpk is $\mathcal{O}(3nm\log q + 2nm\log q + n\log q) = \widetilde{\mathcal{O}}(n^2)$.

The member signing secret-key $\mathbf{gsk}_i$ is a Gaussian vector $\boldsymbol{e}_i \in \mathbb{Z}^{2m}$ of bit-size $\mathcal{O}(2m) = \widetilde{\mathcal{O}}(n)$.

The member revocation token $\mathbf{grt}_i$ is composed of $t = \mathrm{poly}(n)$ vectors $\mathbf{grt}_{i,j} \in \mathbb{Z}_q^n$ of bit-size $\mathcal{O}(tn\log q) = \widetilde{\mathcal{O}}(tn)$.

The signature $\sigma = (\boldsymbol{m}, j, \Pi, \boldsymbol{v}, \boldsymbol{b}_j)$ is of bit-size $\mathcal{O}(\log t + (\ell m\log\beta)\log q + n + m\log q) = \ell\widetilde{\mathcal{O}}(n)$.

Next, we analyze the computation complexity of our lattice-based VLR-GS-BU scheme with respect to n. Here, we let $r < N$ denote the number of revoked members in the RL and t denote the number of time periods.

The KeyGen procedure involves one TrapGen operation, for each member, one SamplePre operation, t FRD operations for the vector over $\mathbb{Z}_q^n$, and $t+1$ matrix-vector multiplication operations over $\mathbb{Z}_q^{n\times m} \times \mathbb{Z}^m$. Thus, the computation complexity is $\widetilde{\mathcal{O}}(n^2) + N(\widetilde{\mathcal{O}}(n^2) + t\widetilde{\mathcal{O}}(n) + (t+1)\mathcal{O}(n^2)) = Nt\widetilde{\mathcal{O}}(n^2)$.

The Sign procedure involves one hash function operation, one matrix-vector multiplication operation, and a proof of the corresponding non-interactive zero-knowledge (NIZK) in Section 3.2. Thus, the computation complexity is $\widetilde{\mathcal{O}}(n^2) + \mathcal{O}(n^2) + \omega(\log n)\widetilde{\mathcal{O}}(n^2) = \widetilde{\mathcal{O}}(n^2)$.

The Verify procedure involves two hash function operations, r matrix-vector multiplication operations, and a verification of the corresponding NIZK in Section 3.2. Thus, the computation complexity is $2\widetilde{\mathcal{O}}(n^2) + r\mathcal{O}(n^2) + \omega(\log n)\widetilde{\mathcal{O}}(n^2) = r\widetilde{\mathcal{O}}(n^2)$.

The detailed comparisons between our construction and previous lattice-based VLR-GS schemes, in terms of asymptotic efficiency, functionality, and security, are given in Table 3 and Figs. 1 and 2.

Our lattice-based VLR-GS enjoys better asymptotic efficiency (except a relatively high cost for

Table 3 Comparison of known lattice-based VLR-GS schemes ($N = 2^{\ell}$)

Scheme	|Gpk|	|**gsk**|	$\|\sigma\|$	Functionality	Free of encryption	BU-security
Ling et al. (2013)'s	$\ell\widetilde{\mathcal{O}}(n^2)$	$\ell\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	VLR	Yes	No
Zhang et al. (2016)'s	$\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\widetilde{\mathcal{O}}(n+\ell)$	VLR	No	No
Gao et al. (2017)'s	$\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\widetilde{\mathcal{O}}(n+\ell)$	VLR	No	No
Ling et al. (2018)'s	$\ell\widetilde{\mathcal{O}}(n^2)$	$\ell\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	VLR	Yes	No
Perera and Koshiba (2018a)'s	$\ell\widetilde{\mathcal{O}}(n^2)$	$\ell\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	VLR	Yes	No
Perera and Koshiba (2018b)'s	$\ell\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	Fully dynamic	No	No
Perera and Koshiba (2018c)'s	$\ell\widetilde{\mathcal{O}}(n^2)$	$\ell\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	Fully dynamic	No	No
Zhang et al. (2019a)'s	$\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	VLR	Yes	No
Zhang et al. (2019b)'s	$\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	VLR	No	No
Ours	$\widetilde{\mathcal{O}}(n^2)$	$\widetilde{\mathcal{O}}(n)$	$\ell\widetilde{\mathcal{O}}(n)$	VLR	Yes	Yes

|Gpk|: size of the group public-key; |**gsk**|: size of a member's signing secret-key; $|\sigma|$: size of the signature; VLR: verifier-local revocation

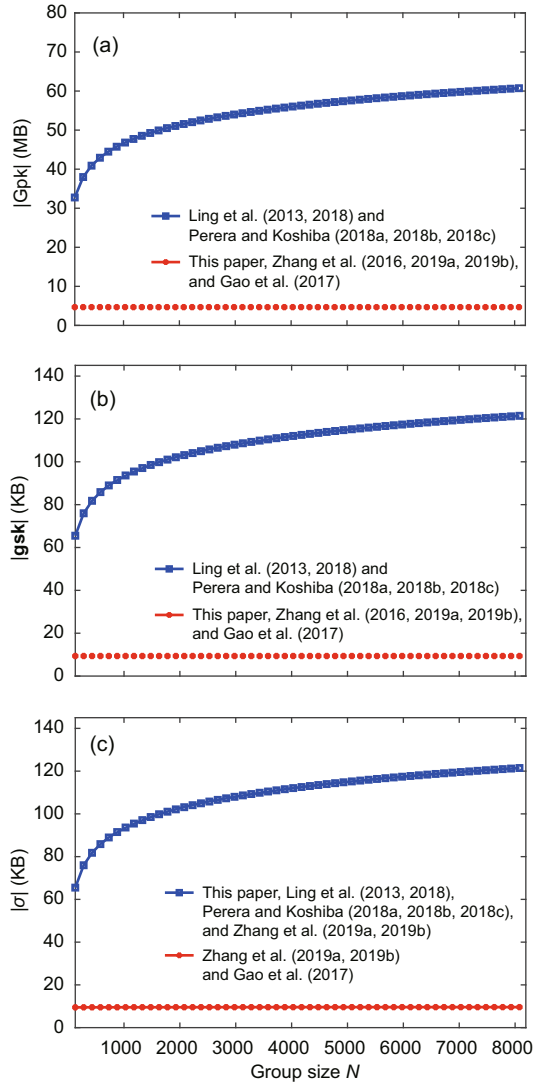

Fig. 1 Comparison of the 10 schemes in terms of |Gpk| (a), |gsk| (b), and $|\sigma|$ (c)

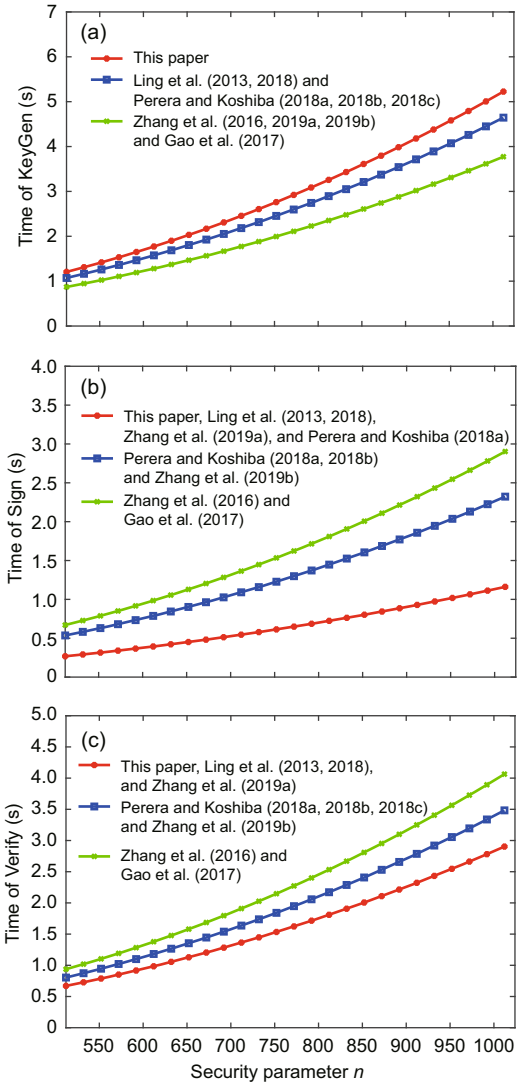

Fig. 2 Comparison of the 10 schemes in terms of KeyGen cost (a), Sign cost (b), and Verify cost (c)

our KeyGen procedure involving extra t FRD and t matrix-vector multiplication operations). Specifically, we have achieved BU security for the first time.

For correctness, BU-anonymity, and traceability, we show the following three theorems. The proof details are given in Appendix.

Theorem 2 The proposed scheme is correct with an overwhelming probability.

Theorem 3 If COM enjoys the statistically hiding property, the proposed scheme is BU-anonymous in the random oracle model.

Theorem 4 If the $\mathrm{SIS}^{\infty}_{n,m,q,2\beta(1+\omega(\sqrt{\log m}))}$ problem is hard, then the proposed scheme is traceable in the random oracle model.

5 Conclusions

In this study, we proposed the first lattice-based VLR-GS scheme with BU security, and thus resolved a prominent open problem. By adopting an injective encoding function with FRD, a compact identity-encoding technique, and the corresponding Stern-type statistical ZKP protocol creatively, our new scheme enjoys a $\mathcal{O}(\log N)$ factor saving for bit-sizes of GPK and member's signing secret-key, and is free of any public-key encryption. Moreover, with BU security, it is more suitable for some large groups with better security.

Contributors

Yanhua ZHANG and Huiwen JIA designed the research. Yanhua ZHANG processed the data and drafted the paper. Ximeng LIU helped organize the paper. Yupu HU and Yong GAN revised and finalized the paper.

Compliance with ethics guidelines

Yanhua ZHANG, Ximeng LIU, Yupu HU, Yong GAN, and Huiwen JIA declare that they have no conflict of interest.

References

Agrawal S, Boneh D, Boyen X, 2010. Efficient lattice (H)IBE in the standard model. Proc 29th Annual Int Conf on the Theory and Applications of Cryptographic Techniques, p.553-572. https://doi.org/10.1007/978-3-642-13190-5_28

Ajtai M, 1996. Generating hard instances of lattice problems (extended abstract). Proc 28th ACM Symp on Theory of Computing, p.99-108. https://doi.org/10.1145/237814.237838

Alwen J, Peikert C, 2011. Generating shorter bases for hard random lattices. *Theor Comput Syst*, 48(3):535-553. https://doi.org/10.1007/s00224-010-9278-3

Bellare M, Micciancio D, Warinschi B, 2003. Foundations of group signatures: formal definitions, simplified requirements, and a construction based on general assumptions. Proc 22nd Int Conf on the Theory and Applications of Cryptographic Techniques, p.614-629. https://doi.org/10.1007/3-540-39200-9_38

Bellare M, Shi HX, Zhang C, 2005. Foundations of group signatures: the case of dynamic groups. Cryptographers' Track at the RSA Conf, p.136-153. https://doi.org/10.1007/978-3-540-30574-3_11

Boneh D, Shacham H, 2004. Group signatures with verifier-local revocation. Proc 11th ACM Conf on Computer and Communications Security, p.168-177. https://doi.org/10.1145/1030083.1030106

Bootle J, Cerulli A, Chaidos P, et al., 2016. Foundations of fully dynamic group signatures. Proc 14th Int Conf on the Applied Cryptography and Network Security, p.117-136. https://doi.org/10.1007/978-3-319-39555-5_7

Cash D, Hofheinz D, Kiltz E, et al., 2010. Bonsai trees, or how to delegate a lattice basis. Proc 29th Int Conf on the Theory and Applications of Cryptographic Techniques, p.523-552. https://doi.org/10.1007/978-3-642-13190-5_27

Chaum D, van Heyst E, 1991. Group signatures. Workshop on the Theory and Application of Cryptographic Techniques, p.257-265. https://doi.org/10.1007/3-540-46416-6_22

Emura K, Hayashi T, 2018. A revocable group signature scheme with scalability from simple assumptions and its implementation. Proc 21st Int Conf on Information Security, p.442-460. https://doi.org/10.1007/978-3-319-99136-8_24

Gao W, Hu YP, Zhang YH, et al., 2017. Lattice-based group signature with verifier-local revocation. *J Shanghai Jiao Tong Univ (Sci)*, 22(3):313-321. https://doi.org/10.1007/s12204-017-1837-1

Gentry C, Peikert C, Vaikuntanathan V, 2008. Trapdoors for hard lattices and new cryptographic constructions. Proc 40th Annual ACM Symp on Theory of Computing, p.197-206. https://doi.org/10.1145/1374376.1374407

Gordon SD, Katz J, Vaikuntanathan V, 2010. A group signature scheme from lattice assumptions. Proc 16th Int Conf on the Theory and Application of Cryptology and Information Security, p.395-412. https://doi.org/10.1007/978-3-642-17373-8_23

Huang JY, Huang Q, Susilo W, 2020. Leakage-resilient group signature: definitions and constructions. *Inform Sci*, 509:119-132. https://doi.org/10.1016/j.ins.2019.09.004

Ishida A, Sakai Y, Emura K, et al., 2018. Fully anonymous group signature with verifier-local revocation. Proc 11th Int Conf on Security and Cryptography for Networks, p.23-42. https://doi.org/10.1007/978-3-319-98113-0_2

Kawachi A, Tanaka K, Xagawa K, 2008. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. Proc 14th Int Conf on the Theory and Application of Cryptology and Information Security, p.372-389. https://doi.org/10.1007/978-3-540-89255-7_23

Langlois A, Ling S, Nguyen K, et al., 2014. Lattice-based group signature scheme with verifier-local revocation. Proc 17th Int Conf on Practice and Theory in Public-Key Cryptography, p.345-361. https://doi.org/10.1007/978-3-642-54631-0_20

Libert B, Vergnaud D, 2009. Group signatures with verifier-local revocation and backward unlinkability in the standard model. Proc 8th Int Conf on Cryptology and Network Security, p.498-517. https://doi.org/10.1007/978-3-642-10433-6_34

Ling S, Nguyen K, Stehlé D, et al., 2013. Improved zero-knowledge proofs of knowledge for the ISIS problem, and applications. Proc 16th Int Conf on Practice and Theory in Public-Key Cryptography, p.107-124. https://doi.org/10.1007/978-3-642-36362-7_8

Ling S, Nguyen K, Roux-Langlois A, et al., 2018. A lattice-based group signature scheme with verifier-local revocation. *Theor Comput Sci*, 730:1-20. https://doi.org/10.1016/j.tcs.2018.03.027

Micciancio D, Peikert C, 2012. Trapdoors for lattices: simpler, tighter, faster, smaller. Proc 31st Annual Int Conf on the Theory and Applications of Cryptographic Techniques, p.700-718. https://doi.org/10.1007/978-3-642-29011-4_41

Micciancio D, Peikert C, 2013. Hardness of SIS and LWE with small parameters. Proc 33rd Annual Cryptology Conf, p.21-39. https://doi.org/10.1007/978-3-642-40041-4_2

Nakanishi T, Funabiki N, 2005. Verifier-local revocation group signature schemes with backward unlinkability from bilinear maps. Proc 11th Int Conf on the Theory and Application of Cryptology and Information Security, p.533-548. https://doi.org/10.1007/11593447_29

Nakanishi T, Funabiki N, 2006. A short verifier-local revocation group signature scheme with backward unlinkability. Proc 1st Int Workshop on Security, p.17-32. https://doi.org/10.1007/11908739_2

Nguyen PQ, Zhang J, Zhang ZF, 2015. Simpler efficient group signatures from lattices. Proc 18th IACR Int Conf on Practice and Theory in Public-Key Cryptography, p.401-426. https://doi.org/10.1007/978-3-662-46447-2_18

Perera MNS, Koshiba T, 2018a. Achieving full security for lattice-based group signatures with verifier-local revocation. Proc 20th Int Conf on Information and Communications Security, p.287-302. https://doi.org/10.1007/978-3-030-01950-1_17

Perera MNS, Koshiba T, 2018b. Zero-knowledge proof for lattice-based group signature schemes with verifier-local revocation. Proc 21st Int Conf on Network-Based Information Systems, p.772-782. https://doi.org/10.1007/978-3-319-98530-5_68

Perera MNS, Koshiba T, 2018c. Achieving strong security and verifier-local revocation for dynamic group signatures from lattice assumptions. Proc 14th Int Conf on Security and Trust Management, p.3-19. https://doi.org/10.1007/978-3-030-01141-3_1

Regev O, 2005. On lattices, learning with errors, random linear codes, and cryptography. Proc 37th Annual ACM Symp on Theory of Computing, p.84-93. https://doi.org/10.1145/1060590.1060603

Song DX, 2001. Practical forward secure group signature schemes. Proc 8th ACM Conf on Computer and Communications Security, p.225-234. https://doi.org/10.1145/501983.502015

Zhang YH, Hu YP, Gao W, et al., 2016. Simpler efficient group signature scheme with verifier-local revocation from lattices. *KSII Trans Int Inform Syst*, 10(1):414-430. https://doi.org/10.3837/tiis.2016.01.024

Zhang YH, Hu YP, Zhang QK, et al., 2019a. On new zero-knowledge proofs for lattice-based group signatures with verifier-local revocation. Proc 22nd Int Conf on Information Security, p.190-208. https://doi.org/10.1007/978-3-030-30215-3_10

Zhang YH, Liu XM, Hu YP, et al., 2019b. Lattice-based group signatures with verifier-local revocation: achieving shorter key-sizes and explicit traceability with ease. Proc 18th Int Conf on Cryptology and Network Security, p.120-140. https://doi.org/10.1007/978-3-030-31578-8_7

Appendix: Proofs for VLR-GS-BU

Proof of Theorem 2

For the first four steps of Verify, a member id with an identity index i having a valid witness $(\boldsymbol{e}_i', \boldsymbol{e}_0) \in \mathbf{Sec}_\beta(\mathrm{id}) \cdot \chi^m$ can return a signature meeting it. As for step 5, $\boldsymbol{e}_{i'}$ can be expressed as

$$\begin{aligned}\boldsymbol{e}_{i'} &= \boldsymbol{b}_j - \boldsymbol{B}^{\mathrm{T}}\mathbf{grt}_{i',j}\\ &= \boldsymbol{B}^{\mathrm{T}}(\mathbf{grt}_{i,j} - \mathbf{grt}_{i',j}) + \boldsymbol{e}_0 \bmod q.\end{aligned}$$

1. Prove $\mathbf{grt}_{i,j} \notin \mathrm{RL}_j \Rightarrow \mathrm{Verify}(\cdot) = \mathrm{valid}$.

Suppose that $\mathbf{grt}_{i,j} \notin \mathrm{RL}_j$. We need to prove that with an overwhelming probability, step 5 is satisfied; i.e., Verify(Gpk,j, RL_j, Sign(Gpk,j, $\mathbf{gsk}_i$,$\boldsymbol{m}$), $\boldsymbol{m}$)=valid and $||\boldsymbol{e}_{i'}||_\infty > \beta$. For all $\mathbf{grt}_{i',j} \in \mathrm{RL}_j$, we have $\boldsymbol{B}^{\mathrm{T}}(\mathbf{grt}_{i,j} - \mathbf{grt}_{i',j}) = \boldsymbol{e}_{i'} - \boldsymbol{e}_0 \bmod q$. Let $\boldsymbol{s}_{i',j} = \mathbf{grt}_{i,j} - \mathbf{grt}_{i',j}$. We have $\|\boldsymbol{B}^{\mathrm{T}}\boldsymbol{s}_{i',j}\|_\infty \le \|\boldsymbol{e}_{i'}\|_\infty + \|\boldsymbol{e}_0\|_\infty \le \|\boldsymbol{e}_{i'}\|_\infty + \beta$. According to Lemma 4 of Ling et al. (2018), we have

$$\Pr[\|\boldsymbol{B}^{\mathrm{T}}\boldsymbol{s}_{i',j}\|_\infty \le 2\beta] \le 1/(4\beta+1)^n.$$

Thus, $\|\boldsymbol{e}_i'\|_\infty > 2\beta - \beta = \beta$ is satisfied with an overwhelming probability ($> 1 - (4\beta+1)^{-n}$).

2. Prove $\mathrm{Verify}(\cdot) = \mathrm{valid} \Rightarrow \mathbf{grt}_{i,j} \notin \mathrm{RL}_j$.

Assume $\mathrm{Verify}(\cdot) = \mathrm{valid}$; for all $\mathbf{grt}_{i',j} \in \mathrm{RL}_j$, we have $\|\boldsymbol{e}_{i'}\|_\infty > \beta$. If there is i' satisfying $\mathbf{grt}_{i,j} = \mathbf{grt}_{i',j}$, we have $\boldsymbol{e}_{i'} = \boldsymbol{e}_0$ and $\|\boldsymbol{e}_{i'}\|_\infty = \|\boldsymbol{e}_0\|_\infty \le \beta$. Thus, a contradiction exists and the above relation holds with probability 1.

Proof of Theorem 3

A list of games is established as follows:

Game 0: $\mathcal{C}$ honestly proceeds as follows:

1. Run KeyGen to obtain $(\mathrm{Gpk}, \mathrm{Gsk}, \mathrm{Grt})$. Set $\mathrm{RL}=\varnothing$, $\mathrm{Corr}=\varnothing$, and send Gpk to $\mathcal{A}$.

2. For $\mathcal{A}$'s signing queries on $\boldsymbol{m}\in\{0,1\}^*$ of member $i\le N-1$ for $j\in\{1,2,\ldots,t\}$, $\mathcal{C}$ returns σ using $\mathrm{Sign}(\mathrm{Gpk}, j, \mathbf{gsk}_i, \boldsymbol{m})$; for $\mathcal{A}$'s corruption queries in period i, $\mathcal{C}$ sets $\mathrm{Corr}=\mathrm{Corr}\cup\{\mathrm{id}, i\}$ and returns $\mathbf{gsk}_i$; for $\mathcal{A}$'s revocation queries for period j, $\mathcal{C}$ sets $\mathrm{RL}=\mathrm{RL}\cup\{\mathbf{grt}_{i,j}\}$ and returns it.

3. $\mathcal{A}$ outputs a message $\boldsymbol{m}^*\in\{0,1\}^*$, a period $j^*\in\{1,2,\ldots,t\}$, and two indices i_0, i_1, and $\forall b\in\{0,1\}$, $\mathbf{grt}_{i_b,1}, \mathbf{grt}_{i_b,2}, \ldots, \mathbf{grt}_{i_b,j^*}\notin\mathrm{RL}$.

4. $\mathcal{C}$ picks $b\xleftarrow{\$}\{0,1\}$, and generates a signature $\sigma^*=\mathrm{Sign}(\mathrm{Gpk}, j^*, \mathbf{gsk}_{i_b}, \boldsymbol{m}^*)=(\boldsymbol{m}^*, j^*, \Pi, \boldsymbol{v}, \boldsymbol{b}_{j^*})$.

5. $\mathcal{A}$ makes queries as before without the right to ask for $\mathbf{gsk}_{i_b}$ or $\mathbf{grt}_{i_b,j}$ $\forall b\in\{0,1\}$ and each $j\in\{1,2,\ldots,j^*\}$.

6. $\mathcal{A}$ outputs $b'\in\{0,1\}$.

Game 1: $\mathcal{C}$ simulates step 4 of Game 0 by programming the oracle:

1. Choose $\boldsymbol{v}\xleftarrow{\$}\{0,1\}^n$ and $\boldsymbol{e}_0\xleftarrow{\$}\chi^m$.

2. Define $\boldsymbol{B}=\mathcal{G}(\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{u}, \boldsymbol{m}^*, \boldsymbol{v})$ and $\boldsymbol{b}_{j^*}=\boldsymbol{B}^{\mathrm{T}}\mathbf{grt}_{i_b,j^*}+\boldsymbol{e}_0 \bmod q$.

3. Program $\mathcal{H}_2$, and Π^* is statistically close to Π.

4. Output $\widehat{\sigma}^*=(\boldsymbol{m}^*, j^*, \Pi^*, \boldsymbol{v}, \boldsymbol{b}_{j^*})$.

Game 2: $\mathcal{C}$ defines $\boldsymbol{b}_{j^*}=\boldsymbol{B}^{\mathrm{T}}\boldsymbol{r}+\boldsymbol{e}_0$, so $\boldsymbol{b}_{j^*}$ is close statistically to the one in Game 1, and thus Game 2 is statistically indistinguishable from Game 1.

Game 3: $\mathcal{C}$ gets $(\boldsymbol{B}, \boldsymbol{b}_{j^*})\xleftarrow{\$}\mathcal{U}$, so $(\boldsymbol{B}, \boldsymbol{b}_{j^*})$ is close to the one in Game 2. Games 3 and 2 are computationally indistinguishable. Furthermore, the advantage $\mathrm{Adv}_{\mathcal{A}}^{\text{BU-anon}}$ is 0.

According to the indistinguishability of Games 1–3, the advantage $\mathrm{Adv}_{\mathcal{A}}^{\text{BU-anon}}$ in Game 1 is negligible; i.e., our new scheme is BU-anonymous.

Proof of Theorem 4

Suppose that a forger $\mathcal{F}^*$ breaks the scheme with advantage ϵ; using $\mathcal{F}^*$, we design an efficient $\mathcal{A}$ to solve the $\mathrm{SIS}^{\infty}_{n,m,q,2\beta(1+\omega(\sqrt{\log m}))}$ problem.

Setup: $\mathcal{A}$ proceeds as follows:

1. Choose $i^*\in\{0,1,\ldots,N-1\}$, $\boldsymbol{e}^*_{i^*,0}, \boldsymbol{e}^*_{i^*,1}\xleftarrow{\$}\mathcal{D}_{\mathbb{Z}^m,s}$, and $\boldsymbol{R}\xleftarrow{\$}\{-1,1\}^{m\times m}$.

2. Run TrapGen to obtain $\boldsymbol{A}_1\in\mathbb{Z}_q^{n\times m}$ and $\boldsymbol{R}_{\boldsymbol{A}_1}$.

3. Define $\boldsymbol{A}=\hat{\boldsymbol{A}}$ and $\boldsymbol{A}_0=\boldsymbol{A}\boldsymbol{R}-i^*\boldsymbol{A}_1 \bmod q$.

4. Sample $\boldsymbol{B}_0, \boldsymbol{B}_1\xleftarrow{\$}\mathbb{Z}_q^{n\times m}$ and define $\boldsymbol{u}=\boldsymbol{A}(\boldsymbol{e}^*_{i^*,0}+\boldsymbol{R}\boldsymbol{e}^*_{i^*,1}) \bmod q$.

5. For period $\mathrm{TP}_{j\in\{1,2,\ldots,t\}}$, define $\mathbf{grt}_{i^*,j}=(\boldsymbol{B}_0+\mathcal{H}_1(\mathrm{TP}_j)\boldsymbol{B}_1)\boldsymbol{e}^*_{i^*,0} \bmod q$.

6. For $i=i^*$, let $\mathbf{gsk}_{i^*}=(\boldsymbol{e}^*_{i^*,0}, \boldsymbol{e}^*_{i^*,1})$ and $\mathbf{grt}_{i^*}=\{\mathbf{grt}_{i^*,1}, \mathbf{grt}_{i^*,2}, \ldots, \mathbf{grt}_{i^*,t}\}$.

7. For $i\ne i^*$, define $\boldsymbol{A}_{\mathrm{id}}=[\boldsymbol{A}|\boldsymbol{A}_0+i\boldsymbol{A}_1]$ and run $\mathrm{SampleRight}(\boldsymbol{A}, (i-i^*)\boldsymbol{A}_1, \boldsymbol{R}, \boldsymbol{R}_{\boldsymbol{A}_1}, \boldsymbol{u}, s)$ to obtain $\boldsymbol{e}_i=(\boldsymbol{e}_{i,0}, \boldsymbol{e}_{i,1})\in\mathbb{Z}^{2m}$. Then let $\mathbf{gsk}_i=\boldsymbol{e}_i$ and $\mathbf{grt}_i=\{\mathbf{grt}_{i,1}, \mathbf{grt}_{i,2}, \ldots, \mathbf{grt}_{i,t}\}$, where $\mathbf{grt}_{i,j}=(\boldsymbol{B}_0+\mathcal{H}_1(\mathrm{TP}_j)\cdot\boldsymbol{B}_1)\boldsymbol{e}_{i,0} \bmod q$.

8. Let $\mathcal{H}_1: \mathbb{Z}_q^n\to\mathbb{Z}_q^{n\times n}$ be an FRD function, and $\mathcal{H}_2: \{0,1\}^*\to\{1,2,3\}^{\kappa=\omega(\log n)}$ and $\mathcal{G}: \{0,1\}^*\to\mathbb{Z}_q^{n\times m}$ be two hash functions.

9. Let $\mathrm{Gpk}=(\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{u}, \mathcal{H}_1, \mathcal{H}_2, \mathcal{G})$, $\mathrm{Gsk}=(\mathbf{gsk}_0, \mathbf{gsk}_1, \ldots, \mathbf{gsk}_{N-1})$, and $\mathrm{Grt}=(\mathbf{grt}_0, \mathbf{grt}_1, \ldots, \mathbf{grt}_{N-1})$, and send $(\mathrm{Gpk}, \mathrm{Grt})$ to $\mathcal{F}^*$.

Queries: $\mathcal{F}^*$ proceeds as follows:

1. Corrupting: take id with an index i as input; $\mathcal{A}$ outputs $\mathbf{gsk}_i$ and adds (id, i) to Corr.

2. Signing: take $\boldsymbol{m}\in\{0,1\}^*$ of i at period j as input; $\mathcal{A}$ outputs σ using $\mathrm{Sign}(\mathrm{Gpk}, j, \mathbf{gsk}_i, \boldsymbol{m})$. In particular, the values in $\{1,2,3\}^{\kappa=\omega(\log n)}$ are sampled as responses to $\mathcal{H}_2$. Let r_d be a reply to the d^{th} $(d\le q_{\mathcal{H}_2})$ query; here, $q_{\mathcal{H}_2}$ is the whole number of queries to $\mathcal{H}_2$.

Forgery: $\mathcal{F}^*$ returns $\boldsymbol{m}^*\in\{0,1\}^*$, $\mathrm{RL}^*_{j^*}\subseteq\mathrm{Grt}$ for period j^*, and a forged $\sigma^*=(\boldsymbol{m}^*, j^*, \Pi^*, \boldsymbol{v}^*, \boldsymbol{b}^*_{j^*})$, which satisfies the following:

1. $\mathrm{Verify}(\mathrm{Gpk}, j^*, \mathrm{RL}^*_{j^*}, \sigma^*, \boldsymbol{m}^*)=\mathrm{valid}$.

2. The implicit-tracing does not succeed, or returns a member not included in $\mathrm{Corr}\backslash\mathrm{RL}^*_{j^*}$.

3. $\mathcal{A}$ has not obtained σ^* by a signing query.

$\mathcal{F}^*$ proceeds as in Zhang et al. (2019b). Let $\boldsymbol{B}^*=\mathcal{G}(\boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{B}_0, \boldsymbol{B}_1, \boldsymbol{u}, \boldsymbol{m}^*, \boldsymbol{v}^*)$. $\mathcal{A}$ obtains a 3-fork involving

$$(\boldsymbol{m}^*, \boldsymbol{A}, \boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{u}, \boldsymbol{B}^*, \boldsymbol{B}_0, \boldsymbol{B}_1, j^*, \boldsymbol{b}^*_{j^*}, \mathrm{CMT}_r\}_{r=1}^{\kappa}),$$

after at most $32q_{\mathcal{H}_2}/(\epsilon-3^{-\kappa})$ executions of $\mathcal{F}^*$. With the help of an extractor $\mathcal{K}$ as described in the argument of knowledge, we obtain a valid witness $=(\mathrm{id}=\mathrm{Bin}(i)\in\{0,1\}^{\ell}, \boldsymbol{e}_i=(\boldsymbol{e}_{i,0}, \boldsymbol{e}_{i,1})\in\mathbb{Z}^{2m}, \boldsymbol{e}_0\in\mathbb{Z}^m)$ such that:

1. $[\boldsymbol{A}|\boldsymbol{A}_0+i\boldsymbol{A}_1]\boldsymbol{e}_i=\boldsymbol{u} \bmod q$, $\boldsymbol{e}_i\in\mathbf{Sec}_{\beta}(\mathrm{id})$.

2. $\boldsymbol{b}^*_{j^*}=((\boldsymbol{B}^*)^{\mathrm{T}}\hat{\boldsymbol{B}_{j^*}})\boldsymbol{e}_{i,0}+\boldsymbol{e}^*_0$, $0<\|\boldsymbol{e}_0\|_{\infty}\le\beta$.

Thus, we show two cases:

1. If $i \neq i^*$ (the probability is at most $1-1/N$), $\mathcal{A}$ aborts.

2. If $i = i^*$, $\mathcal{A}$ returns $\hat{\boldsymbol{e}} = (\boldsymbol{e}^*_{i^*,0} - \boldsymbol{e}_{i^*,0}) + \boldsymbol{R} \cdot (\boldsymbol{e}^*_{i^*,1} - \boldsymbol{e}_{i^*,1})$. Thus, we have

$$\begin{aligned}\hat{\boldsymbol{A}}\hat{\boldsymbol{e}} &= \boldsymbol{A}(\boldsymbol{e}^*_{i^*,0} - \boldsymbol{e}_{i^*,0} + \boldsymbol{R}(\boldsymbol{e}^*_{i^*,1} - \boldsymbol{e}_{i^*,1})) \\ &= \underbrace{\boldsymbol{A}(\boldsymbol{e}^*_{i^*,0} + \boldsymbol{R}\boldsymbol{e}^*_{i^*,1})}_{\boldsymbol{u}} - \underbrace{\boldsymbol{A}(\boldsymbol{e}_{i^*,0} + \boldsymbol{R}\boldsymbol{e}_{i^*,1})}_{\boldsymbol{u}} \\ &= \boldsymbol{0} \bmod q.\end{aligned}$$

We now show that with a high probability, $\hat{\boldsymbol{e}} \neq \boldsymbol{0} \bmod q$ and $\|\hat{\boldsymbol{e}}\| \leq \mathrm{poly}(m)$:

1. $\|\hat{\boldsymbol{e}}\|_\infty \leq \mathrm{poly}(m)$. For $b \in \{0,1\}$, $\|\boldsymbol{e}^*_{i^*,b}\|_\infty \leq \beta$, $\|\boldsymbol{e}_{i^*,b}\|_\infty \leq \beta$, $\boldsymbol{R} \xleftarrow{\$} \{1,-1\}^{m\times m}$; thus, we have $\|\hat{\boldsymbol{e}}\|_\infty \leq (1+\omega(\sqrt{\log m}))2\beta = \mathrm{poly}(m)$.

2. $\hat{\boldsymbol{e}} \neq \boldsymbol{0} \bmod q$. Since $\sigma^* = (\boldsymbol{m}^*, j^*, \Pi^*, \boldsymbol{v}^*, \boldsymbol{b}^*_{j^*})$ is a forged signature, the implicit-tracing does not succeed, or returns a member not included in $\mathrm{Corr}\backslash\mathrm{RL}^*_{j^*}$.

(1) If the implicit-tracing will not succeed, then $\mathrm{Verify}(\mathrm{Gpk}, j^*, \mathbf{grt}_{i^*,j^*}, \sigma^*, \boldsymbol{m}^*) = \text{valid}$ implies that $\hat{\boldsymbol{B}_{j^*}}\boldsymbol{e}_{i^*,0} \bmod q \neq \mathbf{grt}_{i^*,j^*} = \hat{\boldsymbol{B}_{j^*}}\boldsymbol{e}^*_{i^*,0} \bmod q$; thus, $\boldsymbol{e}_{i^*,0} \neq \boldsymbol{e}^*_{i^*,0}$.

(2) If the implicit-tracing algorithm traces to a member $\hat{i}^* \notin \mathrm{Corr}\backslash\mathrm{RL}^*_{j^*}$, clearly we have the following facts:

$$\begin{cases}\mathrm{Verify}(\mathrm{Gpk}, j^*, \mathbf{grt}_{\hat{i}^*,j^*}, \sigma^*, \boldsymbol{m}^*) = \text{invalid}, \\ \mathrm{Verify}(\mathrm{Gpk}, j^*, \mathrm{RL}^*_{j^*}, \sigma^*, \boldsymbol{m}^*) = \text{valid}.\end{cases}$$

Thus, we have:

(a) $\mathbf{grt}_{\hat{i}^*,j^*} \notin \mathrm{RL}^*_{j^*}$; thus, $\hat{i}^* \notin \mathrm{Corr}$.

(b) Since $\|\boldsymbol{b}^*_{j^*} - (\boldsymbol{B}^*)^{\mathrm{T}}\mathbf{grt}_{\hat{i}^*,j^*}\|_\infty = \|(\boldsymbol{B}^*)^{\mathrm{T}}(\hat{\boldsymbol{B}_{j^*}}\boldsymbol{e}_{i^*,0} - \mathbf{grt}_{\hat{i}^*,j^*}) + \boldsymbol{e}_0\|_\infty \leq \beta$, $\|\boldsymbol{e}_0\|_\infty \leq \beta$, $\|(\boldsymbol{B}^*)^{\mathrm{T}}(\hat{\boldsymbol{B}_{j^*}}\boldsymbol{e}_{i^*,0} - \mathbf{grt}_{\hat{i}^*,j^*})\|_\infty \leq 2\beta$. Further, according to Lemma 4 of Ling et al. (2018), we have $\mathbf{grt}_{\hat{i}^*,j^*} = \hat{\boldsymbol{B}_{j^*}}\boldsymbol{e}_{i^*,0} \bmod q$ with an overwhelming probability.

Now, consider the following two cases:

(c) If $\mathcal{F}^*$ does not request $\mathbf{gsk}_{i^*}$, then vector $(\boldsymbol{e}^*_{i^*,0}, \boldsymbol{e}^*_{i^*,1})$ cannot be known to $\mathcal{F}^*$; thus, according to Lemma 1, we have $(\boldsymbol{e}^*_{i^*,0}, \boldsymbol{e}^*_{i^*,1}) \neq (\boldsymbol{e}_{i^*,0}, \boldsymbol{e}_{i^*,1})$ with an overwhelming probability.

(d) If $\mathcal{F}^*$ requests $\mathbf{gsk}_{i^*}$, then $i^* \in \mathrm{Corr}$, and thus $i^* \neq \hat{i}^*$; therefore, $\mathbf{grt}_{i^*,j^*} \neq \mathbf{grt}_{\hat{i}^*,,j^*}$, which means $\boldsymbol{e}_{i^*,0} \neq \boldsymbol{e}^*_{i^*,0}$.

The following analysis is the same as those in Zhang et al. (2019b). For the different cases in 2(1) and 2(2)(d) (assume that $\boldsymbol{e}^*_{i^*,1} = \boldsymbol{e}_{i^*,1}$), and in 2(1), 2(2)(c), and 2(2)(d) (assume that $\boldsymbol{e}^*_{i^*,1} \neq \boldsymbol{e}_{i^*,1}$), we conclude that with probability $1 - \exp^{-\widetilde{\mathcal{O}}(n)}$, $\hat{\boldsymbol{e}} = (\boldsymbol{e}^*_{i^*,0} - \boldsymbol{e}_{i^*,0}) + \boldsymbol{R}\cdot(\boldsymbol{e}^*_{i^*,1} - \boldsymbol{e}_{i^*,1}) \neq \boldsymbol{0} \bmod q$. Therefore, based on the above analysis, we conclude that with a probability $\epsilon' \geq \epsilon/(2N)(1-(7/9)^\kappa)(1-\exp^{-\widetilde{\mathcal{O}}(n)})$, $\hat{\boldsymbol{e}}$ will satisfy $\hat{\boldsymbol{A}}\hat{\boldsymbol{e}} = \boldsymbol{0} \bmod q$, $0 \neq \|\hat{\boldsymbol{e}}\|_\infty \leq 2\beta(1+\omega(\sqrt{\log m})) = \mathrm{poly}(m)$.