Achieving Full Security for Lattice-based Group Signatures with Verifier-local Revocation

Maharage Nisansala Sevwandi Perera[1] and Takeshi Koshiba[2]

[1] Graduate School of Science and Engineering
Saitama University, Saitama, Japan
perera.m.n.s.119@ms.saitama-u.ac.jp,
[2] Faculty of Education and Integrated Arts and Sciences
Waseda University, Tokyo, Japan
tkoshiba@waseda.jp

Abstract. Even though Verifier-local revocation mechanism seems to be the most flexible revocation method that suits for any size of groups it could not reach strong security yet. Verifier-local revocation technique needs to update only the verifiers with revocation messages when a member is revoked while most of the revocation mechanisms require to re-initialize the group or track changes of the group. The first lattice-based group signature scheme with verifier-local revocability was suggested by Langlois, Ling, Nguyen, and Wang (PKC 2014). However, their scheme relies on a weaker security notion. On the other hand, Bellare, Micciancio, and Warinschi (EUROCRYPT 2003) proposed formal security definitions called full-anonymity and full-traceability for static groups. Achieving full-anonymity for schemes with verifier-local revocation is technically challenging because those schemes use a token system. This paper provides a scheme with verifier-local revocation that achieves the full-anonymity and full-traceability.

Keywords: lattice-based group signatures, verifier-local revocation, full-anonymity, full-traceability

1 Introduction

In the setting of group signatures introduced by Chaum and van Heyst [9], group members can generate signatures for the group anonymously (anonymity). On the other hand, the group manager can extract the identity of the group member who created the signature (traceability). Thus, the original group signature scheme has two core requirements, anonymity and traceability. Later more requirements such as unlinkability, unforgeability, and framing resistance have been proposed. However, the precise meaning of those requirements not always clear and sometimes their meaning overlap each other. Bellare et al. [2] (BMW03 model) proposed strong and formal definitions for the core requirements of the group signatures with two security notions called, *full-anonymity* and *full-traceability*. The full-anonymity and the full-traceability, which imply all the existing security notions provide a conceptual simplification since it requires to check only two security properties in a group signature scheme. However, the BMW03 model is for static groups, not for dynamic groups. In real-life almost all the group settings are stateless. Thus, member registration and member revocation requirements are essential when applying the group signature schemes in practice.

When a member is misbehaved, he should be punished. For instance, if a member issued a signature for an unnecessary document, he should be removed from the group. Member revocation in group signature schemes requires restricting members signing in future after revoking them. There are several member revocation methods. For instance, one revocation method is generating and distributing new keys for each member and verifiers or requesting each member to update their keys and generating the group public key newly. Since this requires to update all the unrevoked members and the verifiers, it is inconvenient to implement practically. Bresson et al. [5] suggested another revocation technique by extending the signing procedure of the scheme given in [8]. Their revocation method requires signers to proof at the zero-knowledge that his identity is not in the public list of revoked identities. However, this method causes the linear growth of the size of the group signatures with the number of revoked members. Thus it is a burden for the signers. Brickell [6] proposed a revocation method called Verifier-local revocation (VLR), which was subsequently formalized by Boneh et al. [4] in their scheme. VLR requires to pass revocation messages only to the verifiers when a member is revoked. In real-life scenarios, since the number of verifiers is much less than the number of members, passing messages only to the verifiers are efficient than any other revocation technique. Most of the group signature schemes (e.g., [16], [3]) operate in the bilinear map setting which will be insecure once quantum computers become a reality.

Lattice-based cryptography is the most prominent solution for the post-quantum cryptography. It provides provable security under worst-case hardness assumptions. Gorden et al. [11] suggested the first lattice-based group signature scheme. However, the sizes of both the group public key and the signature in their scheme increase with the number of members (N) (linear-barrier problem). Thus, it cannot apply to large groups. Then Camenisch et al. [7] presented a lattice-based group signature scheme with anonymous attribute token system, which still experiences the linear-barrier problem. Later, Languillaumie et al. [13] presented a scheme with a solution to the linear-barrier problem. However, the first three lattice-based group signature schemes follow LWE-based PKE (public-key encryption) scheme, and they are only for static groups.

Langlois et al. [14] proposed the first lattice-based group signature scheme which facilitates member revocation and free of LWE-based PKE. They have used VLR as the member revocation technique, and their scheme is more efficient while based on weaker security assumptions. In terms of security, their scheme satisfies a weaker security notion called *selfless-anonymity*. The VLR group signature schemes cannot employ the full-anonymity described in the BMW03 model directly because VLR group signature schemes use a token system to manage member revocation. Thus, each member has a token other than their secret signing key. In the full anonymity game between a challenger and an adversary as described in the BMW03 model, all the member secret signing keys are given to the adversary at the beginning. In VLR group signature schemes, revocation tokens cannot be given to the adversary because he can identify the signer of a signature using tokens. Other than that, secret signing keys cannot be given to him because he can derive the revocation token from the secret signing keys.

The present lattice-based VLR group signature schemes raise a problem, that is whether it is possible to design a VLR lattice-based group signature scheme in the BMW03 model that achieves the full-anonymity.

1.1 Our Contribution

The lattice-based VLR group signature scheme in [14] relies on the selfless-anonymity. Stronger security for VLR schemes can be achieved in two ways. One approach is by using a restricted-version of full anonymity. The other process is changing the methods in the scheme. We provide a new group signature scheme that can achieve the full-anonymity using the second method.

The previous lattice-based group signatures failed to obtain the full-anonymity because anyone possessing revocation tokens can execute signature verification algorithm and confirm whether the relevant member created the signature or not. For instance, in the anonymity game between a challenger and an adversary, if the adversary knows the revocation tokens of the challenging indices, then he can execute Verify with revocation tokens he has. If Verify returns Invalid, then he knows that the owner of the revocation token generated the signature. Thus, this leads to an assumption that the verifiers should not see the revocation tokens, especially the challenging indices' revocation tokens. Based on this assumption, new security notions were proposed ([18, 19]). However, none of them are as strong as full-anonymity because they do not provide all the revocation tokens to the adversary. Thus those security notions are restricted version of full-anonymity.

This paper suggests a scheme that can provide all the revocation tokens to the adversary even the challenged indices' revocation tokens. In original VLR schemes, when revoking a member, the group manager adds the revoking member's token into a list called revocation list (RL) and passes RL to the verifiers. Thus, Verify has an additional input RL, and the verifiers have to check whether the singer's revocation token is not in the list other than verifying the signature. We suggest a new revocation method for VLR schemes that the group manager has to sign each revocation token before adding to RL. On the other hand, at the signature verification, the verifier has to check whether the revocation tokens in RL are signed by the group manager other than validating the signer and the signature. Thus, even the adversary obtains any revocation token he cannot execute Verify because the adversary does not know the group manager's secret key. Now, we can apply the full-anonymity for our VLR group signature scheme and provide all the member secret signing keys and revocation tokens including the challenging indices' details to the adversary at the full-anonymity game.

As a result, we deliver a new lattice-based group signature scheme using VLR with new revocation and verification methods, that satisfies the full-anonymity.

2 Preliminaries

2.1 Notations

For any integer $k \geq 1$, we denote the set of integers $\{1, \ldots, k\}$ by $[k]$. We denote matrices by bold upper-case letters such as $\mathbf{A}$, and vectors by bold lower-case letters, such as $\mathbf{x}$. We assume that all vectors are in column form. While the concatenation of matrices $\mathbf{A} \in \mathbb{R}^{n\times m}$ and $\mathbf{B} \in \mathbb{R}^{n\times k}$, is denoted by $[\mathbf{A}|\mathbf{B}] \in \mathbb{R}^{n\times(m+k)}$ the concatenation of vectors $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^k$ is denoted by $(\mathbf{x}\|\mathbf{y}) \in \mathbb{R}^{m+k}$. If S is a finite set, $b \xleftarrow{\$} S$ means that b is chosen uniformly at random from S.

Throughout this paper, we present the security parameter as n and the maximum number of members in a group as $N = 2^\ell \in \mathsf{poly}(n)$. We select prime modulus $q{=}\omega(n^2 \log n)$, dimension $m{\geq}\ 2n \log q$, Gaussian parameter $\sigma{=}\omega(\sqrt{n \log q \log n})$. Moreover we select the integer norm bound β=$\lceil\sigma \cdot \log m\rceil$ s.t. $(4\beta + 1)^2 \leq q$, the number of decomposition p=$\lfloor\log\beta\rfloor + 1$, the sequence of integers $\beta_1, \beta_2, \beta_3, \ldots, \beta_p$=$\beta_1 = \lceil\beta/2\rceil; \beta_2 = \lceil(\beta - \beta_1)/2\rceil; \beta_3 = \lceil(\beta - \beta_1 - \beta_2)/2\rceil; \ldots; \beta_p = 1$. We choose the number of protocol repetitions t=$\omega(\log n)$. Let $\mathcal{H}$: $\{0,1\}^* \to \{1,2,3\}^t$, and $\mathcal{G}$: $\{0,1\}^* \to \mathbb{Z}_q^{n\times m}$ be hash functions, modeled as random oracles. We use one-time signature scheme $\mathcal{OTS}$ = (OGen, OSign, OVer), where OGen is the key generation algorithm of $\mathcal{OTS}$ key pair (**ovk**, **osk**), OSign is signature generation and OVer is signature verification functions.

2.2 Lattices

Let q be a prime and $\mathbf{B} = [\mathbf{b}_1|\cdots|\mathbf{b}_m] \in \mathbb{Z}_q^{r\times m}$ be linearly independent vectors in $\mathbb{Z}_q^r$. The r-dimensional lattice $\Lambda(\mathbf{B})$ for $\mathbf{B}$ is defined as

$$\Lambda(\mathbf{B}) = \{\mathbf{y} \in \mathbb{Z}^r \mid \mathbf{y} \equiv \mathbf{B}\mathbf{x} \bmod q \text{ for some } \mathbf{x} \in \mathbb{Z}_q^m\},$$

which is the set of all linear combinations of columns of $\mathbf{B}$ and m is the rank of $\mathbf{B}$.

We consider a discrete Gaussian distribution for a lattice. The Gaussian function centered in a vector $\mathbf{c}$ with parameter $s > 0$ is defined as $\rho_{s,\mathbf{c}}(\mathbf{x}) = e^{-\pi\|(\mathbf{x}-\mathbf{c})/s\|^2}$ and the corresponding probability density function proportional to $\rho_{s,\mathbf{c}}$ is defined as $D_{s,\mathbf{c}}(\mathbf{x}) = \rho_{s,\mathbf{c}}(\mathbf{x})/s^n$ for all $\mathbf{x} \in \mathbb{R}^n$. The discrete Gaussian distribution with respect to a lattice Λ is defined as $D_{\Lambda,s,\mathbf{c}}(\mathbf{x}) = D_{s,\mathbf{c}}(\mathbf{x})/D_{s,\mathbf{c}}(\Lambda) = \rho_{s,\mathbf{c}}(\mathbf{x})/\rho_{s,\mathbf{c}}(\Lambda)$ for all $\mathbf{x} \in \Lambda$. Since $\mathbb{Z}^m$ is also a lattice, we can define a discrete Gaussian distribution for $\mathbb{Z}^m$. By $D_{\mathbb{Z}^m,\sigma}$, we denote the discrete Gaussian distribution for $\mathbb{Z}^m$ around the origin with the standard deviation σ.

2.3 Lattice-Related Properties

The security of our scheme depends on the hardness of Learning With Errors (**LWE**) and two homogeneous and Inhomogeneous Short Integer Solution Problems (**SIS** and **ISIS**).

Definition 1 (LWE [17]). *LWE is parametrized by $n, m \geq 1, q \geq 2$, and χ. For $\boldsymbol{s} \in \mathbb{Z}_q^n$, the distribution $\mathrm{A}_{\boldsymbol{s},\chi}$ is obtained by sampling $\boldsymbol{a} \in \mathbb{Z}_q^n$ uniformly at random and $\mathrm{e} \leftarrow \chi$, and outputting the pair $(\boldsymbol{a}, \boldsymbol{a}^T \cdot \boldsymbol{s} + \mathrm{e})$.*

There are two versions of LWE problem, *Search-LWE* and *Decision-LWE*. While Search-LWE requires to find the secret $\mathbf{s}$, Decision-LWE requires to distinguish LWE samples and samples chosen according to the uniform distribution. We use the hardness of Decision-LWE problem.

For a prime power q, $b \geq \sqrt{n}\omega(\log n)$, and distribution χ, solving $LWE_{n,q,\chi}$ problem is at least as hard as solving $SIVP_\gamma$ (*Shortest Independent Vector Problem*), where $\gamma = \tilde{O}(nq/b)$ [21].

Definition 2 (SIS [17, 21]). *Given m uniformly random vectors $\boldsymbol{a}_i \in \mathbb{Z}_q^n$, forming the columns of a matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, find a nonzero vector $\boldsymbol{x} \in \Lambda^\perp(\boldsymbol{A})$ such that $||\boldsymbol{x}|| \leq \beta$ and $\boldsymbol{A}\boldsymbol{x} = 0 \mod q$.*

Definition 3 (ISIS [14]). *Given m uniformly random vectors $\boldsymbol{a}_i \in \mathbb{Z}_q^n$, forming the columns of a matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, find a vector $\boldsymbol{x} \in \Lambda_{\boldsymbol{u}}^\perp(\boldsymbol{A})$ such that $||\boldsymbol{x}|| \leq \beta$.*

For any m, $\beta = \mathsf{poly}(n)$, and $q \geq \beta \cdot \omega(\sqrt{n \log n})$, solving $SIS_{n,m,q,\beta}$ problem or $ISIS_{n,m,q,\beta}$ problem with non-negligible probability is at least as hard as solving $SIVP_\gamma$ problem, for some $\gamma = \tilde{O}(\beta\sqrt{n})$ [10].

2.4 Lattice-Related Algorithms

We use a randomized nearest-plane algorithm, called, $\mathsf{SampleD}$ [10, 15] and preimage sampleable trapdoor functions (PSTFs) $\mathsf{GenTrap}$ [1,10,15].

- $\mathsf{SampleD}(\mathbf{R}, \mathbf{A}, \mathbf{u}, \sigma)$ outputs $\mathbf{x} \in \mathbb{Z}^m$ sampled from the distribution $D_{\mathbb{Z}^m,\sigma}$ for any vector $\mathbf{u}$ in the image of $\mathbf{A}$, a trapdoor $\mathbf{R}$ and $\sigma = \omega(\sqrt{n \log q \log n})$. The output $\mathbf{x}$ should satisfy the condition $\mathbf{A} \cdot \mathbf{x} = \mathbf{u} \bmod q$.
- $\mathsf{GenTrap}(n, m, q)$ is an efficient randomized algorithm that outputs a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a trapdoor matrix $\mathbf{R}$ for given any integers $n \geq 1, q \geq 2$, and sufficiently large $m = 2n \log q$. The distribution of the output $\mathbf{A}$ is negl(n)-far from the uniform distribution.

2.5 VLR Group Signature

The VLR group signature scheme consists of three PPT algorithms [4] since the *implicit tracing algorithm* is used to trace the misbehaved users.

- $\mathsf{KeyGen}(n, N)$: This randomized PPT algorithm takes as inputs the security parameter n and the maximum number of group members N, and outputs a group public key $\mathbf{gpk}$, a vector of user secret keys $\mathbf{gsk} = (\mathbf{gsk}[0], \mathbf{gsk}[1], ..., \mathbf{gsk}[N-1])$, and a vector of user revocation tokens $\mathbf{grt} = (\mathbf{grt}[0], \mathbf{grt}[1], ..., \mathbf{grt}[N-1])$.

- $\mathsf{Sign}(\mathbf{gpk}, \mathbf{gsk}[d], M)$: This randomized algorithm takes a secret signing key $\mathbf{gsk}[d]$ and a message $M \in \{0,1\}^*$ as inputs and returns a signature Σ.
- $\mathsf{Verify}(\mathbf{gpk}, \mathrm{RL}, \Sigma, M)$: This deterministic algorithm verifies whether the given Σ is valid on M using $\mathbf{gpk}$. Then it validates that the signer not being revoked using RL.

Implicit Tracing Algorithm: Any VLR group signature scheme has an *implicit tracing algorithm* that uses **grt** as the tracing key and traces a signature to at least one group user who generated it. For an input valid signature Σ on a message M run $\mathsf{Verify}(\mathbf{gpk}, \mathrm{RL}, \Sigma, M)$ for each $i = 0, \ldots, N-1$. It outputs the index of the first user for the verification algorithm returns invalid. The tracing algorithm fails if this algorithm verifies properly for all users on the given signature.

3 Definitions of the Security Notations

In this section, first, we define some existing security notions related to our contribution. Then we discuss the difficulties of employing the full-anonymity given in the BMW03 model directly to the existing VLR group signature schemes.

- *Anonymity* requires that no adversary without group manager's key recovers the identity of the user from its signature, which is generated by one of the indices from two indistinguishable indices.
- *Traceability* requires that no adversary forges a signature that cannot be traced.

3.1 Full Anonymity and Full Traceability

Bellare et al. [2] delivered a standard security model (BMW03 model) for group signatures with two strong security properties, *full anonymity* and *full traceability.*

3.2 Selfless-anonymity

Selfless-anonymity is a relaxed anonymity, and it differs from the full-anonymity by the limitations it has. The selfless-anonymity provides none of the member secret keys to the adversary, but only the group public key is given. However, even with these weaknesses, the selfless-anonymity facilitates any member to determine whether his secret signing key is used to generate a particular signature if he forgets whether he signed the message.

The full-anonymity and the selfless-anonymity games between a challenger and an adversary are given in Appendix A

3.3 Difficulties of achieving the full-anonymity for VLR schemes

The full-anonymity is suggested for static groups. Thus, members have only secret signing keys. Even the secret signing key is used to generate signatures, by using the secret signing keys nobody can guess the signer. But the members in VLR schemes have another secret attribute called revocation token. Revealing revocation tokens to the outsiders makes the scheme insecure. For instance, if an adversary knows the user i_0's revocation token $\mathbf{grt}[i_0]$, then the adversary can confirm whether the user i_0 generated the given signature or not by executing Verify by replacing RL with $\mathbf{grt}[i_0]$ as depicted in Figure 1. According to the full-anonymity game in Figure 1 if Σ_b is generated by user i_0, then Verify return Res as Invalid for i_0. Thus it confirms that user i_0 generated the signature. Moreover, since VLR group signatures derive the revocation tokens from the secret signing keys, the selfless-anonymity also restricts revealing the secret signing keys.

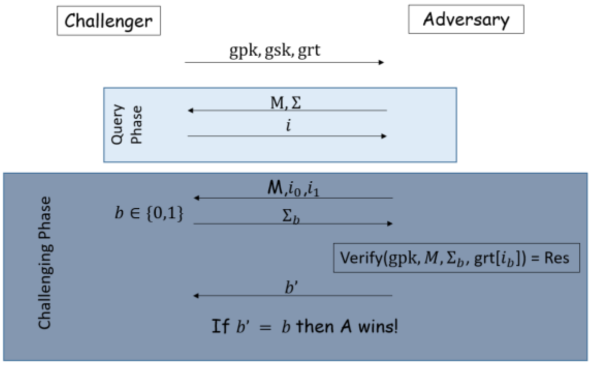

Fig. 1. Full Anonymity for VLR schemes

Because of these reasons, to obtain stronger security for VLR group signature schemes, we need a restricted version of full anonymity or new scheme with different methods.

4 New lattice-based VLR Scheme

The new scheme requests the group manager to sign revoking member's token before adding to the revocation list RL. Thus the group manager signs the revoking member's revocation token **grt** using the group manager secret key **gmsk**. Accordingly, at the signature verification, the verifier has to check whether the revocation tokens in RL are signed by the group manager. For this, the verifier executes Verify with the group manager's public key. Because of this reason an adversary who knows the revocation token of any member i cannot replace RL in Verify(**gpk**, M, Σ, RL) with the i's revocation token $\mathbf{grt}[i]$ and check whether the user i generated the signature or not. The signature verification algorithm rejects verifying the given signature because the adversary is providing a revocation token which is not signed by the group manager.

In the full-anonymity game depicted in Figure 1 when the adversary tries to execute Verify with the revocation token of i_0 and i_1 he gets Invalid as the response in both cases because he fails to provide tokens with the group manager's signature. Thus, the adversary cannot understand the signer of the given signature. Therefore, the new scheme can employ the full-anonymity by giving all the members' secret signing keys and tokens to the adversary.

4.1 Description of the Scheme

We use the scheme in [14] as the base and construct our new scheme as follows.

Key Generation: This randomized algorithm KeyGen(n, N) works as below.

1. Run PPT algorithm GenTrap(n, m, q) to get $\mathbf{A}_0 \in \mathbb{Z}_q^{n\times m}$ and a trapdoor $\mathbf{T_A}$.
2. Sample $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^n$ and $\mathbf{A}_i^b \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$ for each $b \in \{0,1\}$ and $i \in [\ell]$.
3. Set the matrix $\mathbf{A} = [\mathbf{A}_0|\mathbf{A}_1^0|\mathbf{A}_1^1|\ldots|\mathbf{A}_\ell^0|\mathbf{A}_\ell^1] \in \mathbb{Z}_q^{n\times(2\ell+1)m}$.
4. Run GenTrap(n,m,q) to obtain $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$ and a trapdoor $\mathbf{T_B}$.
5. For each group member select a ℓ-bit string as the index d and generate secret signing keys and revocation tokens as below.

(a) Let $d = d[1] \dots d[\ell] \in \{0,1\}^\ell$ be the binary representation of d.
(b) Sample vectors $\mathbf{x}_1^{d[1]}, \dots, \mathbf{x}_\ell^{d[\ell]} \hookleftarrow D_{\mathbb{Z}^m,\sigma}$.
(c) Compute $\mathbf{z} = \sum_{i=1}^{\ell} \mathbf{A}_i^{d[i]} \cdot \mathbf{x}_i^{d[i]} \bmod q$.
(d) Get $\mathbf{x}_0 \in \mathbb{Z}^m \leftarrow \mathsf{SampleD}(\mathbf{T_A}, \mathbf{A}_0, \mathbf{u} - \mathbf{z}, \sigma)$.
(e) Let $\mathbf{x}_1^{1-d[1]}, \dots, \mathbf{x}_\ell^{1-d[\ell]}$ be zero vectors $\mathbf{0}^m$.
(f) Define $\mathbf{x} = (\mathbf{x}_0 || \mathbf{x}_1^0 || \mathbf{x}_1^1 || \dots || \mathbf{x}_\ell^0 || \mathbf{x}_\ell^1) \in \mathbb{Z}^{(2\ell+1)m}$.
If $||\mathbf{x}||_\infty \leq \beta$ then proceed else repeat from (b).
(g) Let the user secret signing key be $\mathbf{gsk}[d] = \mathbf{x}^{(d)}$ and revocation token be $\mathbf{grt}[d] = \mathbf{A}_0 \cdot \mathbf{x_0} \in \mathbb{Z}_q^n$.

Now we have, the group public key $\mathbf{gpk} = (\mathbf{A}, \mathbf{B}, \mathbf{u})$, the group manager's secret key **gmsk** $=\mathbf{T_B}$, the group manager's public key $\mathbf{gmpk} = \mathbf{B}$, group members' secret signing keys $\mathbf{gsk} = (\mathbf{gsk}[0], \mathbf{gsk}[1], \dots, \mathbf{gsk}[N-1])$, and their revocation tokens $\mathbf{grt} = (\mathbf{grt}[0], \mathbf{grt}[1], \dots, \mathbf{grt}[N-1])$.

Signing: The randomized algorithm $\mathsf{Sign}(\mathbf{gpk}, \mathbf{gsk}, M)$ generates Σ on a message M as follows.

1. Generate a one-time-signature $\mathcal{OTS}$ key pair (**ovk, osk**) using OGen.
2. Sample $\rho \xleftarrow{\$} \{0,1\}^n$, let $\mathbf{V} = \mathcal{G}(\mathbf{A}, \mathbf{u}, M, \rho) \in \mathbb{Z}_q^{m \times n}$.
3. Sample $\mathbf{e} \leftarrow \chi^m$.
4. Compute $\mathbf{v} = \mathbf{V} \cdot (\mathbf{A}_0 \cdot \mathbf{x_0}) + \mathbf{e} \mod q$ ($||\mathbf{e}||_\infty \leq \beta$ with overwhelming probability and $(\mathbf{A}_0 \cdot \mathbf{x_0})$ is the revocation token **grt** of user i).
5. Repeat the zero knowledge interactive protocol of the commitment described in Section 4.2 $t = \omega(\log n)$ times with the public parameter ($\mathbf{A}$, $\mathbf{u}$, $\mathbf{V}$, $\mathbf{v}$) and prover's witness ($\mathbf{x}$, $\mathbf{e}$) to make the soundness error negligible and proof that user is certified. Then make it non-interactive using the Fiat-Shamir heuristic as a triple, $\Pi = (\{CMT^{(k)}\}_{k=1}^t, CH, \{RSP^{(k)}\}_{k=1}^t)$, where CH $= (\{Ch^{(k)}\}_{k=1}^t) = \mathcal{H}(M, \mathbf{A}, \mathbf{u}, \mathbf{V}, \mathbf{v}, \{CMT^{(k)}\}_{k=1}^t) \in \{1,2,3\}^t$.
6. Compute $\mathcal{OTS}$; $sig = \mathsf{OSig}(\mathbf{osk}, \Pi)$.
7. Output signature $\Sigma = (\mathbf{ovk}, M, \rho, \mathbf{v}, \Pi, sig)$.

Verification: $\mathsf{Verify}(\mathbf{gpk}, M, \Sigma, RL = \{\{\mathbf{u}_i\}_i\})$ verifies the given signature Σ is valid on the given message M and signer is a valid member as follows.

1. Parse the signature Σ as $(\mathbf{ovk}, M, \rho, \mathbf{v}, \Pi, sig)$.
2. If $\mathsf{OVer}(\mathbf{ovk}, \Pi, sig) = 0$ then return 0.
3. Get $\mathbf{V} = \mathcal{G}(\mathbf{A}, \mathbf{u}, M, \rho) \in \mathbb{Z}_q^{m \times n}$.
4. Parse Π as $(\{CMT^{(k)}\}_{k=1}^t, \{Ch^{(k)}\}_{k=1}^t, \{RSP^{(k)}\}_{k=1}^t)$.

5. If $(Ch^{(1)}, \dots, Ch^{(t)}) \neq \mathcal{H}(M, \mathbf{A}, \mathbf{u}, \mathbf{V}, \mathbf{v}, \{CMT^{(k)}\}_{k=1}^{t})$ then return 0.
6. For $k = 1$ to t run the verification steps of the commitment scheme to validate $RSP^{(k)}$ with respect to $CMT^{(k)}$ and $Ch^{(k)}$. If any of the conditions fails then output invalid and hold.
7. For each $\mathbf{u}_i \in RL$,
   (a) Parse $\mathbf{u}_i$ as $(\mathbf{grt}_i, \Sigma_{rt_i})$.
   (b) Check whether $\mathbf{grt}_i$ is signed by the group manager by executing $\mathsf{Verify}(\mathbf{gmpk}, \mathbf{grt}_i, \Sigma_{rt_i})$, where $\mathbf{gmpk}$ is the group manager's public key. If $\mathsf{Verify}(\mathbf{gmpk}, \mathbf{grt}_i, \Sigma_{rt_i})$, returns $\mathsf{Invalid}$ then return $\mathsf{Invalid}$.
   (c) Compute $\mathbf{e}_i' = \mathbf{v} - \mathbf{V} \cdot \mathbf{grt}_i \mod q$ to check whether there exists an index i such that $||\mathbf{e}_i'||_\infty \leq \beta$. If so return invalid.
8. Return valid.

Revoke: The algorithm $\mathsf{Revoke}(\mathbf{gpk}, \mathbf{gmsk}, \mathbf{grt}[i], \mathrm{RL})$ takes the group manager's secret key $\mathbf{gmsk}$, revoking member's revocation token $\mathbf{grt}[i]$, and latest revocation list RL and proceeds as follows.

1. Generate a signature for the revoking token as $\Sigma_{rt_i} = \mathsf{Sign}(\mathbf{gmsk}, \mathbf{grt}[i])$.
2. Add revoking token and generated signature to RL, $RL \leftarrow RL \cup (\mathbf{grt}_i, \Sigma_{rt_i})$.
3. Return RL.

4.2 The Underlying ZKAoK for the Group Signature Scheme

Zero-Knowledge Interactive Protocol is the main building block of the scheme as it allows a signer to argue that he is a certified group member who has a valid secret key and who has not been revoked.

Let COM be the statistically hiding and computationally binding commitment scheme described in [12].

Our scheme can be seen as an adaptation of [14]. Thus we can use the protocol described in [14]. We use matrix $\mathbf{A} = [\mathbf{A}_0|\mathbf{A}_1^0|\mathbf{A}_1^1|\dots|\mathbf{A}_\ell^0|\mathbf{A}_\ell^1] \in \mathbb{Z}_q^{n\times(2\ell+1)m}$, $\mathbf{V} \in \mathbb{Z}_q^{m\times n}$, $\mathbf{u} \in \mathbb{Z}_q^n$, and $\mathbf{v} \in \mathbb{Z}_q^m$ as the public parameters. The witness of the prover is the vector $\mathbf{x}^{(d)} = (\mathbf{x}_0||\mathbf{x}_1^0||\mathbf{x}_1^1||...||\mathbf{x}_\ell^0||\mathbf{x}_\ell^1) \in \Sigma^{(2\ell+1)m}$ for some $d \in \{0,1\}^\ell$ and vector $\mathbf{e} \in \mathbb{Z}^m$. While keeping prover's identity d in secret he has to convince the verifier that,

1. $\mathbf{A} \cdot \mathbf{x} = \mathbf{u} \bmod q$ and d is hidden in $\mathbf{x}^{(d)}$.
2. $||\mathbf{e}||_\infty \leq \beta$ and $\mathbf{V} \cdot (\mathbf{A}_0 \cdot \mathbf{x}_0) + \mathbf{e} = \mathbf{v} \mod q$.

5 Analysis of the Scheme

This paper provides a new scheme that satisfies the full-anonymity. However, the restricted versions of full-anonymity called almost-full anonymity [19] and dynamical-almost-full anonymity [18] are efficient than the proposed scheme because those schemes do not require the group manager to sign revoking tokens. Moreover, in the selfless-anonymity, any user can check whether he created a particular signature or not. But in the proposed scheme this is not possible since the users do not know the group manager's secret key. However, in terms of security, the new scheme is much stronger than any other security applied for VLR schemes.

5.1 Correctness

For all **gpk**, **gmsk**, **gmpk**, **gsk**, and **grt**,
$\mathsf{Verify}(\mathbf{gpk}, M, \mathsf{Sign}(\mathbf{gpk}, \mathbf{gsk}[i], M), RL) = \text{Valid} \iff \mathbf{grt}[i] \notin RL$ and
For all $(\mathbf{grt}_i, \Sigma_{rt_i})$ in RL, $\mathsf{Verify}(\mathbf{gmpk}, \mathbf{grt}_i, \Sigma_{rt_i})$= Valid.

Verify in the proposed scheme only accepts signatures generated on given messages and which are only generated by active members. If the revocation token of the signer is in RL, then his signature is not accepted by Verify. Similarly Sign also checks whether the signer can satisfy those requirements. The underlying interactive protocol confirms that only active members can generate signatures and signers have to possess valid secret signing key.

5.2 Anonymity

Theorem 1. *In the random oracle model, the proposed scheme is full anonymous based on the hardness of $Decision - LWE_{n,q,\chi}$ problem.*

Proof. We define a sequence of games conducted between a challenger C and an adversary A, where the advantage of the adversary is negligible in the last game. Game 0 is the original full-anonymity game which provides all the members' secret signing keys and revocation tokens to the adversary at the beginning. The adversary can request the index of the signer by giving a signature. We prove that the games are indistinguishable, based on $\mathcal{OTS}$, the zero-knowledge property of the underlying argument system, and the hardness of the $Decision - LWE_{n,q,\chi}$ problem. Game 4 is the last game which is independent of the bit $b \in \{0, 1\}$.

Game 0: The challenger C runs $\mathsf{KeyGen}(1^n, 1^N)$ to obtain the group public key $\mathbf{gpk} = (\mathbf{A}, \mathbf{B}, \mathbf{u})$, the group manager's secret key $\mathbf{gmsk} = \mathbf{T_B}$, the group manager's public key $\mathbf{gmpk} = \mathbf{B}$, group members' secret signing keys $\mathbf{gsk} = (\mathbf{gsk}[0], \mathbf{gsk}[1], \ldots, \mathbf{gsk}[N-1])$, and their revocation tokens $\mathbf{grt} = (\mathbf{grt}[0], \mathbf{grt}[1], \ldots, \mathbf{grt}[N-1])$. The challenger C gives the group public key $\mathbf{gpk}$ and all the group members' secret keys $\mathbf{gsk}$ and revocation tokens $\mathbf{grt}$ to the adversary A. In the query phase, A can request to reveal index of the signer for any signature. In the challenge phase, A sends two indices (i_0, i_1) together with a message M^* and C generates and sends back the challenging signature $\Sigma^* = (\mathbf{ovk}, M^*, \rho, \mathbf{v}, \Pi, sig)$ for a random bit $b \leftarrow \{0,1\}$. The adversary's goal is to identify which index is used to generate the challenging signature. A returns b'. If $b' = b$ then the experiment returns 1 or 0 otherwise.

Game 1: In this game, the challenger C makes a slight modification with respect to Game 0. In Game the one-time key pair ($\mathbf{ovk}$, $\mathbf{osk}$) is generated at the signature generation. In this game, C generates the one-time key pair $(\mathbf{ovk}^*, \mathbf{osk}^*)$ at the beginning of the game. If A accesses the opening oracle with a valid signature $\Sigma = (\mathbf{ovk}, M, \rho, \mathbf{v}, \Pi, sig)$, where $\mathbf{ovk}{=}\mathbf{ovk}^*$, C returns a random bit and aborts. However, A comes up with a signature Σ, where $\mathbf{ovk}{=}\mathbf{ovk}^*$ contradicts the strong unforgeability of $\mathcal{OTS}$, and since $\mathbf{ovk}^*$ is independent of the adversary's view, the probability of $\mathbf{ovk}{=}\mathbf{ovk}^*$ is negligible. Even after seeing the challenging signature if A comes up with a valid signature Σ where $\mathbf{ovk}{=}\mathbf{ovk}^*$, then sig is a forged one-time signature, which defeats the strong unforgeability of $\mathcal{OTS}$. Thus, we assume that A does not request for opening of a valid signature with $\mathbf{ovk}^*$ and aborting the game is negligible.

Game 2: In this game, without honestly generating the legitimate non-interactive proof Π, the challenger C simulates the proof Π^* without using the witness. C invokes the simulator for each $k \in [t]$ and then programs the random oracle $\mathcal{H}$ accordingly. The challenging signature $\Sigma^* = (\mathbf{ovk}^*, M^*, \rho, \mathbf{v}, \Pi^*, sig)$ is statistically close to the challenging signature in the previous game because the argument system is statistically zero-knowledge. Thus Game 2 is indistinguishable from Game 1.

Game 3: In this game, the challenger C replaces the original revocation token by a vector sampled uniformly random. The original game has $\mathbf{v} = \mathbf{V} \cdot \mathbf{grt}[i_b] + \mathbf{e} \mod q$, where $\mathbf{V}$ is uniformly random over $\mathbb{Z}_q^{m \times n}$ and $\mathbf{e}$ is sampled from the error distribution χ. In this game C samples a vector $\mathbf{t} \xleftarrow{\$} \mathbb{Z}_q^n$ uniformly and computes $\mathbf{v} = \mathbf{V} \cdot \mathbf{t} + \mathbf{e} \mod q$. The challenger C replaces only the revocation token $\mathbf{grt}[i_b]$ with $\mathbf{t}$. The rest of the game is same as Game 2. Thus, the two games are statistically indistinguishable.

Game 4: Game 3 has $\mathbf{v} = \mathbf{V} \cdot \mathbf{t} + \mathbf{e}_1 \mod q$. In this game the challenger C makes $\mathbf{v}$ truly uniform by sampling $\mathbf{y} \xleftarrow{\$} \mathbb{Z}_q^m$ and setting $\mathbf{v} = \mathbf{y}$. Thus, C

makes revocation token totally independent of the bit b. In Game 3, ($\mathbf{V}$, $\mathbf{v}$) pair is a proper $LWE_{n,q,\chi}$ instance. Thus, the distribution of the pair ($\mathbf{V}$, $\mathbf{v}$) is computationally close to the uniform distribution over $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$. Game 3 and Game 4 are indistinguishable, under the assumption of the hardness of $LWE_{n,q,\chi}$ problem. If the adversary can distinguish $\mathbf{v}$ from $\mathbf{y}$, then he can solve Decision-LWE problem.

Hence, these games prove that the new scheme is secure with full anonymity.

5.3 Traceability

Theorem 2. *Based on the hardness of* $\boldsymbol{SIS}^{\infty}_{n,(\ell+1)\cdot m,q,2\beta}$ *problem, the proposed scheme is traceable, in the random oracle model.*

We construct a PPT algorithm $\mathcal{F}$ that solves SIS problem with non-negligible probability. The forgery $\mathcal{F}$ is given the verification key ($\mathbf{A}$, $\mathbf{u}$) and then he generates the key pair ($\mathbf{B}$, $\mathbf{T_B}$). $\mathcal{F}$ passes $\mathbf{gpk}$=($\mathbf{A}$, $\mathbf{u}$, $\mathbf{B}$) and $\mathbf{gmsk}$ = $\mathbf{T_B}$ and responds to the A's queries as follow.

- **Signatures queries**: If A queries signature of user d on a random message M, then $\mathcal{F}$ returns simulated $\Sigma = \mathsf{Sign}(\mathbf{gpk}, \mathbf{gsk}[d], M)$.
- **Corruption queries**: The corruption set CU is initially set to be empty. If A queries the secret key of any user d, then $\mathcal{F}$ adds d to the set CU and returns $\mathbf{gsk}[d]$.
- Queries to the random oracles $\mathcal{H}, \mathcal{G}$ are handled by consistently returning uniformly random values in $\{1, 2, 3\}^t$. For each $k \le q_{\mathcal{H}}$, we let r_k denote the answer to the k-th query.

Finally, A outputs a message M^*, revocation data RL^* and a non-trivial forged signature Σ^*, which satisfies the requirements of the traceability game, where Σ^* such that $\mathsf{Verify}(\mathbf{gpk}, M^*, \Sigma^*, RL^*) = \mathsf{Valid}$ and implicit tracing algorithm fails, or returns a user index j^* outside of the coalition $CU \setminus RL^*$.
$\mathcal{F}$ exploits the forgery as below.

We require that A always queries $\mathcal{H}$ on input $(M^*, \mathbf{A}, \mathbf{u}, \mathbf{V}^*, \mathbf{v}^*, \{CMT^{(k)}\}_{k=1}^t)$. As a result, with probability at least $\epsilon - 3^{-t}$, there exists certain $\kappa^* \le q_{\mathcal{H}}$ such that the κ^*-th oracle queries involve the tuple $(M^*, \mathbf{A}, \mathbf{u}, \mathbf{V}^*, \mathbf{v}^*, \{CMT^{(k)}\}_{k=1}^t)$. For any fixed κ^* run A many times and input as in the original run. For each repeated run, A returns same output $r'_{\kappa^*}, \dots, r'_{\kappa^*-1}$ for the first κ^*-1 queries as in initial run and from the κ^*-th query onwards return fresh random values $r'_{\kappa^*}, \ldots, r'_{q\mathcal{H}} \xleftarrow{\$} \{1,2,3\}^t$. The forking lemma [[20], Lemma 7] implies that, with probability larger than 1/2, $\mathcal{F}$ can obtain a 3-fork involving tuple $(M^*, \mathbf{A}, \mathbf{u}, \mathbf{V}^*, \mathbf{v}^*, \{CMT^{(k)}\}_{k=1}^t)$ after less than $32 \cdot q\mathcal{H}/(\epsilon - 3^{-t})$ executions of A. Let the responses of $\mathcal{F}$ with respect to the 3-fork branches be $r_{\kappa^*}^{(1)} = (Ch_1^{(1)}, \ldots, Ch_t^{(1)}); r_{\kappa^*}^{(2)} = (Ch_1^{(2)}, \ldots, Ch_t^{(2)}); r_{\kappa^*}^{(3)} = (Ch_1^{(3)}, \ldots, Ch_t^{(3)})$. A simple calculation shows that $Pr[\exists j \in \{1, \ldots, t\} : \{Ch_i^{(1)}, Ch_i^{(2)}, Ch_i^{(3)}\}] = \{1,2,3\}1 - (7/9)^t$. Under the condition of the existence of such index i, one parses the 3 forgeries corresponding to the fork branches to obtain $(RSP_i^{(1)}, RSP_i^{(2)}, RSP_i^{(3)})$.

Then by using the knowledge extractor ζ of the underlying argument system, we can extract vectors $(\mathbf{y}, \mathbf{e})$. These vectors satisfy the followings.

1. $\mathbf{y} = (\mathbf{y}_0 || \mathbf{y}_1^0 || \mathbf{y}_1^1 || \ldots || \mathbf{y}_\ell^0 || \mathbf{y}_\ell^1)$ for some $d \in \{0,1\}^\ell$, and $\mathbf{A} \cdot \mathbf{y} = \mathbf{u} \mod q$.
2. $||\mathbf{e}^*||_\infty \leq \beta$ and $\mathbf{V}^* \cdot (\mathbf{A}_0 \cdot \mathbf{y}_0) + \mathbf{e}^* = \mathbf{v}^* \mod q$.

Remaining proof is same as the proof given in [14]. Thus finally, we can obtain a vector, which is a valid solution to the SIS problem. This concludes the proof of traceability.

6 Conclusion

This paper provides a new scheme with new methods for member revocation and signature verifications. As a result, the proposed scheme was able to achieve the full-anonymity becoming the first lattice-based group signature scheme with VLR that achieves the full-anonymity in comparison with known lattice-based group signature schemes. However, the group manager has to sign every revoking members' s token. This leads to an open problem because the security of the scheme depends on the trust of the group manager. If the group manager's information is revealed, then the scheme is not secure.

Acknowledgments. This work is supported in part by JSPS Grant-in-Aids for Scientic Research (A) JP16H01705 and for Scientic Research (B) JP17H01695.

References

1. Agrawal, S., Boyen, X., Vaikuntanathan, V., Voulgaris, P., Wee, H.: Functional encryption for threshold functions (or fuzzy ibe) from lattices. In: PKC 2012, LNCS. vol. 7293, pp. 280–297. Springer Berlin Heidelberg (2012)

2. Bellare, M., Micciancio, D., Warinschi, B.: Foundations of group signatures: Formal definitions, simplified requirements, and a construction based on general assumptions. In: EUROCRYPT 2003, LNCS. vol. 2656, pp. 614–629. Springer Berlin Heidelberg (2003)
3. Bichsel, P., Camenisch, J., Neven, G., Smart, N.P., Warinschi, B.: Get shorty via group signatures without encryption. In: SCN 2010. vol. 10, pp. 381–398. LNCS (2010)
4. Boneh, D., Shacham, H.: Group signatures with verifier-local revocation. In: ACM-CCS 2004. pp. 168–177. ACM (2004)
5. Bresson, E., Stern, J.: Efficient revocation in group signatures. In: PKC 2001, LNCS. vol. 1992, pp. 190–206. Springer Berlin Heidelberg (2001)
6. Brickell, E.: An efficient protocol for anonymously providing assurance of the container of the private key. *Submitted to the Trusted Comp. Group (April 2003)* (2003)
7. Camenisch, J., Neven, G., Rückert, M.: Fully anonymous attribute tokens from lattices. In: SCN 2012, LNCS. vol. 12, pp. 57–75. Springer Berlin Heidelberg (2012)
8. Camenisch, J., Stadler, M.: Efficient group signature schemes for large groups. In: CRYPTO 1997. pp. 410–424. Springer (1997)
9. Chaum, D., Van Heyst, E.: Group signatures. In: EUROCRYPT 1991, LNCS. vol. 547, pp. 257–265. Springer Berlin Heidelberg (1991)
10. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: ACM 2008. pp. 197–206. ACM (2008)
11. Gordon, S.D., Katz, J., Vaikuntanathan, V.: A group signature scheme from lattice assumptions. In: ASIACRYPT 2010, LNCS. vol. 6477, pp. 395–412. Springer Berlin Heidelberg (2010)
12. Kawachi, A., Tanaka, K., Xagawa, K.: Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: ASIACRYPT 2008, LNCS. vol. 5350, pp. 372–389. Springer Berlin Heidelberg (2008)
13. Laguillaumie, F., Langlois, A., Libert, B., Stehlé, D.: Lattice-based group signatures with logarithmic signature size. In: ASIACRYPT 2013, LNCS. vol. 8270, pp. 41–61. Springer Berlin Heidelberg (2013)
14. Langlois, A., Ling, S., Nguyen, K., Wang, H.: Lattice-based group signature scheme with verifier-local revocation. In: PKC 2014, LNCS. vol. 8383, pp. 345–361. Springer Berlin Heidelberg (2014)
15. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: EUROCRYPT 2012. vol. 7237, pp. 700–718. Springer Berlin Heidelberg (2012)
16. Nakanishi, T., Funabiki, N.: Verifier-local revocation group signature schemes with backward unlinkability from bilinear maps. In: Advances in Cryptology - ASIACRYPT 2005. pp. 533–548. Springer (2005)
17. Peikert, C.: A decade of lattice cryptography. Foundations and Trends in Theoretical Computer Science 10(4), 283–424 (2016), https://doi.org/10.1561/0400000074
18. Perera, M.N.S., Koshiba, T.: Achieving almost-full security for lattice-based fully dynamic group signatures with verifier-local revocation. In: ISPEC 2018, LNCS (to appear)
19. Perera, M.N.S., Koshiba, T.: Fully dynamic group signature scheme with member registration and verifier-local revocation. In: ICMC 2018, Mathematics and Computing (to appear)
20. Pointcheval, D., Vaudenay, S.: On Provable Security for Digital Signature Algorithms. Ecole Normale Supérieure (Paris). Laboratoire d'Informatique (1996)
21. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: STOC 2005. pp. 84–93. ACM Press (2005)

A Security Notions

A.1 Full Anonymity

The full-anonymity game between a challenger and an adversary is as follows. The adversary is strong as he has given all the member secret keys. At the beginning of the game, all the user secret keys **gsk** and the public key **gpk** are given to the adversary, and he can see the outcome of the tracing algorithm.

- **Initial Phase:** The challenger C runs KeyGen to obtain (**gpk**, **gmsk**,**gsk**). Then gives (**gpk**,**gsk**) to the adversary A.
- **Query Phase:** The adversary A can access the opening oracle, which results with $\mathsf{Open}(\mathbf{gmsk}, M, \varSigma)$ when queried with a message M and a signature $\varSigma$.
- **Challenge Phase:** The adversary A outputs a message M and two distinct identities i_0, i_1. The challenger C selects a bit $b \overset{\$}{\leftarrow} \{0,1\}$, generates a signature $\varSigma^*$, and sends to the adversary A. The adversary still can query the opening oracle except the signature challenged.
- **Guessing Phase:** Finally, A outputs a bit b', the guess of b. If $b' = b$, then the adversary A wins.

Definition 4. *Let A be an adversary against the anonymity of a group signature scheme GS. The advantage of A in the above full-anonymity game is*

$$\boldsymbol{Adv}^{anon}_{\mathrm{GS,A}}(n, N) = |\Pr[\boldsymbol{Exp}^{anon}_{\mathrm{GS,A}}(n, N) = 1] - 1/2|.$$

A group signature scheme is full-anonymous if $\boldsymbol{Adv}^{anon}_{\mathrm{GS,A}}$ is negligible.

A.2 Selfless-anonymity

The adversary in the selfless-anonymity game is weaker than the adversary in the full anonymity game since the adversary has not given any secret key in the selfless-anonymity game. The adversary has to determine which of the two adaptively chosen keys generated the challenging signature.

- **Initial Phase:** The challenger C runs KeyGen to obtain (**gpk**, **gsk**, **grt**). Then gives **gpk** to the adversary A.
- **Query Phase:** The adversary A can make the following queries.

1. Signing: The adversary A requests a signature for any message $M \in \{0,1\}^*$ with any user index i, and C returns $\Sigma = \mathsf{Sign}(\mathbf{gpk}, \mathbf{gsk}[i], M)$.
2. Corruption: The adversary A queries for the secret key of any user i, and the challenger C returns $\mathbf{gsk}[i]$.
3. Revocation: The adversary A queries for the revocation token of any user i, and the challenger C returns $\mathbf{grt}[i]$.

- **Challenge Phase:** The adversary A outputs a message M^* and two distinct identities i_0, i_1, such that A did not make the corruption or revocation queries for i_0, i_1. The challenger C selects a bit $b \xleftarrow{\$} \{0,1\}$, computes signature $\Sigma^*{=}\mathsf{Sign}(\mathbf{gpk}, \mathbf{gsk}[i_b], M^*)$ for i_b, and sends the challenging signature Σ^* to the adversary A.
- **Restricted Queries:** Even after the challenge phase the adversary A can make queries but with following restrictions.
  - Signing: The adversary A can query as before.
  - Corruption: The adversary A cannot query for i_0 or i_1.
  - Revocation: The adversary A cannot query for i_0 or i_1.
- **Guessing Phase:** Finally, the adversary A outputs a bit b', the guess of b. If $b' = b$, then A wins.

Definition 5. *Let A be an adversary against the anonymity of a VLR group signature scheme DGS. The advantage of A in the above selfless-anonymity game is*

$$\boldsymbol{Adv}^{anon}_{\mathrm{DGS,A}}(n, N) = |\Pr[\boldsymbol{Exp}^{anon}_{\mathrm{DGS,A}}(n, N) = 1] - 1/2|.$$

A VLR group signature scheme is selfless-anonymous if $\boldsymbol{Adv}^{anon}_{\mathrm{DGS,A}}$ *is negligible.*