Sun *et al. Cybersecurity* (2019) 2:21
https://doi.org/10.1186/s42400-019-0037-8

Cybersecurity

RESEARCH Open Access

An efficient full dynamic group signature scheme over ring

Yiru Sun[1,2*], Yanyan Liu[1,2] and Bo Wu[1,2]

Abstract

The group signature scheme is an important primitive in cryptography, it allows members in a group to generate signatures anonymously on behalf of the whole group. In view of the practical application of such schemes, it is necessary to allow users' registration and revocation when necessary, which makes the construction of dynamic group signature schemes become a significant direction. On the basis of (Ling et al., Lattice-based group signatures: achieving full dynamicity with ease, 2017), we present the first full dynamic group signature scheme over ring, and under the premise of ensuring security, the efficiency of the scheme is improved mainly from the following three aspects: the size of keys, the dynamic construction of a Merkle hash tree that used to record the information of registered users, and the reuse of the leaves in this tree. In addition, the public and secret keys of both group manager and trace manager are generated by a trusted third party, which prevents the situation that the two managers generate their respective public key and secret key maliciously. Compared with the counterpart of the scheme in (Ling et al., Lattice-based group signatures: achieving full dynamicity with ease, 2017) over ring, the expected space complexity of the Merkle tree used in our work down almost by half, and the computational complexity of its update has been reduced by a notch because of the dynamic construction of the hash tree.

Keywords: Group signature, Dynamic, Merkle Tree, Ring-LWE

Introduction

The concept of group signature scheme was proposed by Chaum and van Heyst (1991), which allows and only allows members in a group to sign messages anonymously on behalf of the whole group, and the generated signature would reveals nothing about the identity of the signer. In other words, the verifier in the scheme can only verify that the signature was generated by one of the group members, and have no idea which member it is. However, the trace manager can use its secret key to open the signature to trace the identity of the signer, which avoids the unnecessary disputes. In view of the group signature scheme has the above properties: anonymity (Chen and Pedersen 1994) and traceability, which help the group signature scheme to be one of the cryptography primitives to realize anonymous authentication.

In the early stages, most of the constructions of group signature schemes are static (Boneh et al. 2004; Camenisch and Lysyanskaya 2004; Nguyen and Naini 2004; Furukawa and Yonezawa 2004), namely the members in a group and its size are all fixed in the setup phase, no changes about these parameters would appear during the subsequent operations in the scheme. And furthermore, they also assume that the group manager is always honest and trustworthy. After that, many other properties were considered in the construction of the group signature schemes:

(1) It is fortunately that the size of public key and generated signatures could do not depend on the size of the group (Camenisch and Stadler 1997; Camenisch and Michels 1998), this property is very important for the construction and application of group signature schemes, which avoids the over-expansion of the size of public key and signatures as the number of valid group members increases, and makes the schemes with this property

*Correspondence: sunyiru@iie.ac.cn
[1]State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Springer Open

© The Author(s). 2019 **Open Access** This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

are well suited for large groups. At the same time, the former is beneficial to improve the implementation efficiency of schemes, while the latter makes the communication complexity and cost of the scheme are independent of the group size.

(2) The power of the group manager was weakened (Bellare et al. 2005) by separating a trace manager GM_{trace} from the group manager GM_{update} and decreasing the trust level to each authority to enhance protection for honest algorithm executant, for example, their key pairs were generated by a trust third party, which improves the security of the algorithms and makes them closer to the practical application. GM_{trace} is responsible for the trace of a signature when necessary, and GM_{update} is responsible for the registration and revocation of users and the update of the group information. The tracing soundness of a group signature scheme (Stern 1996) no longer assumes that the group managers are all reliable, which means that, before the verifier outputs the final verification result, the identity of the signer traced by a trace manager and the corresponding proof are also need to be checked. This improvement makes the constructed group signature schemes have stronger security.

(3) Semi-dynamic model (Kiayias and Yung 2006), which involves the dynamic registration that allows users to apply to join the group when needed in RO model (Camenisch and Stadler 1997; Camenisch and Michels 1998; Ateniese et al. 2000; Furukawa and Imai 2005; Kawachi et al. 2008; Delerablée and Pointcheval 2006; Bichsel et al. 2010) and standard model (Practical Group Signatures Without Random Oracles; Boyen and Waters 2006; Groth 2006; 2007; Boyen and Waters 2007; Signing on Elements in Bilinear Groups for Modular Protocol Design), or the dynamic revocation that allows the group manager to remove certain group members from the group. And there are different manners to realize the latter functionality:

   (a) The group manager updates the group public key and distribute it to the users that are not revoked (Sakai et al. 2012; Camenisch and Lysyanskaya 2002).
   (b) Making use of a accumulator (Dodis et al. 2004; Nguyen 2005), which allows efficient proof of group membership and update of the group information.
   (c) The signer is required to include a proof of eligible membership when signing a message (Bresson and Stern 2001) or update its secret key (Boneh et al. 2004) according to the changes of the group.
   (d) VLR(verifier local revocation) (An Efficient Protocol for Anonymously Providing Assurance of the Container of a Private Key; Boneh and Shacham 2004; Nakanishi and Funabiki 2005; Libert and Vergnaud 2009) means that the list of revoked group members is only distributed to the verifier.

(4) Full dynamic model (Naor et al. 2001; Peikert and Rosen 2007; Camenisch and Groth 2004; Nakanishi et al. 2009; Libert et al. 2012a, b), which allows both the dynamic registration of users and the dynamic revocation of group members, which makes the algorithm has stronger security and higher practicability.

The security of schemes mentioned above are mostly based on the hardness assumption in the algebraic theory while the development of quantum computing technology makes such schemes meet serious security problems. Fortunately, the research of the post-quantum cryptography has brought new hope to cryptology. And as one important branch of it, lattice based cryptography is widely considered has potential ability to against quantum attack, because there is no efficient algorithm has been found to breaks the hardness assumptions based on lattice. However, the computational complexity and space complexity of lattice based cryptographic schemes have not been solved very well.

The first lattice based group signature scheme is given in (Gordon et al. 2010) in 2010, which was improved to obtain stronger anonymity in (Camenisch et al. 2012), and given the size of group N, the size of signatures generated by the schemes in (Gordon et al. 2010; Camenisch et al. 2012) are all polynomials in N. Subsequently, the size of the signature was lowered up to $O(\log N)$ in (Laguillaumie et al. 2013; Nguyen et al. 2015; Ling et al. 2015) by different manners. And then, an efficient lattice based static group signature scheme is presented in (Libert et al. 2016b) without using the GPV trapdoor (Gentry et al. 2008), where a Merkle tree was used as an accumulator to keep a record of the registered user and group information. In order to further satisfy the requirements of making the schemes allow users to register and to be revoked dynamically, the schemes in (Langlois et al. 2014; Libert et al. 2016a) are dependent on lattice trapdoor seriously, and contains some complex modules. By combining the static scheme in (Libert et al. 2016b) with the security model in (Bootle et al. 2016), it is possible to realize the dynamic registration and revocation of users efficiently (Ling et al. 2017). It includes an update algorithm in accumulator, and both the security and the signature size were improved.

In this paper, the first full dynamic group signature scheme over ring is presented inspired by (Ling et al.

2017), which realizes the full dynamic register and revocation of users, the dynamic construction of Merkle hash tree that is used to record the legitimate users with their witnesses and the group information, the reuse of leaves in this tree, and the honestly generation of keys of GM=(GM_{update}, GM_{trace}) by a trusted third party, which leads to a reduction in the security of the generated algorithm. And in theory, the trust third party needs to be completely trusted and not easy to be violated, however, it is impossible in practice. We can only use relatively trusted entities to partially implement the functions of a trusted third party, such as certificate authority(CA), to avoid situations where the group manager and trace manager generate their respective keys maliciously. Concretely, the scheme in this paper improves the efficiency of that in (Ling et al. 2017) from the following three aspects:

(1) To reduces the size of keys and signature, the scheme is implemented over ring, which also helps to reduce the space complexity and computational complexity of the scheme.
(2) The dynamic construction and update of the Merkle hash tree allows the size of it expanded along with the size of group gradually, and this change helps to reduce both the computational complexity of the update of group information and the space complexity of the scheme.
(3) The reuse of leaves in Merkle hash tree is realized in this scheme, which reduces the space complexity of the scheme indirectly to a certain extent.

Though we have tried a lot, there is still a large space for improvement in the use of zero-knowledge protocol to proof a legitimate membership. And the problem of the delayed verification of a signature is also not solved, the direct idea to solve this problem is to store the signature and the verification information or just store the verification result of the signature by the group manager at each time τ, and the verifier requests the corresponding information from it as needed. Unfortunately, this would increase the space complexity unlimitedly along with the extension of the time.

In the remainder of this paper, we start by reviewing some definitions, theorems used in the scheme, and the dynamic algorithm to construct the Merkle hash tree in "Preliminaries" section. And then the detailed full dynamic group signature scheme is presented in "The efficient full dynamic group signature scheme" section. To analysis the security properties of the scheme, we present the underlying zero knowledge protocol and its security analysis in "The underlying protocol" section. Finally, we discuss the properties of the scheme in "The analysis of the group signature scheme" section, and conclusion in "Conclusion" section.

Preliminaries

The background of lattice

In this section, we will review some notations, definitions and theorems used for analysing our main results. Throughout this paper, set the security parameter λ, integer $n = O(\lambda)$, prime modules $q = \tilde{O}\left(n^{1.5}\right), k = \lceil \log q \rceil, m = 2k$, and $\mathbf{R} = \mathbf{Z}[x]/f(x), f(x) = x^n + 1, \mathbf{R}_q = \mathbf{R}/q\mathbf{R}$, given vectors $\mathbf{x} = (x_1, \cdots, x_m), \mathbf{z} = (z_1, \cdots, z_m)$, integer t, then $\|\mathbf{x}\|_t = \left(\sum_{i=1}^{m} \|x_i\|^t\right)^{\frac{1}{t}}$ denotes its t-norm, $(\mathbf{x}|\mathbf{z})$ is a concatenation of the two vectors.

Definition 1 (The ring-**SVP** and ring-**SIVP**) *(Lyubashevsky et al. 2013) Given a field* $\mathbf{R}$*, let* $\gamma \geq 1$*, then the ring-*$\mathbf{SVP}_\gamma$ *problem is: given the ideal lattice* $\mathcal{I}$ *over* $\mathbf{R}$*, find out a non-zero short vector* $\mathbf{x} \in \mathcal{I}$*, such that* $\|\mathbf{x}\|_\infty \leq \gamma \cdot \lambda_1(\mathcal{I})$*. And the ring-*$\mathbf{SIVP}_\gamma$ *problem could be defined similarly: find out n independent elements* $(\mathbf{x}_1, \cdots, \mathbf{x}_n)$ *in* $\mathcal{I}$*, such that* $\|(\mathbf{x}_1, \cdots, \mathbf{x}_n)\|_\infty \leq \gamma \cdot \lambda_n(\mathcal{I})$*.*

Definition 2 (The ring-$\mathbf{SIS}^\infty_{n,m,q,\beta}$) *(Ling et al. 2015; Peikert 2016) Choose m elements* $a_j \xleftarrow{\$} \mathbf{R}_q$ *uniformly, let random vector* $\mathbf{A} = (a_1, \cdots, a_m) \in \mathbf{R}_q^m$*, positive real number* $\beta = poly(n)$*, find out a non-zero short vector* $\mathbf{z} = (z_1, \cdots, z_m) \in \mathbf{R}_q^m, \|\mathbf{z}\|_\infty \leq \beta$*, such that*

$$f_{\mathbf{A}}(\mathbf{z}) = \langle \mathbf{A}, \mathbf{z} \rangle = \mathbf{A}^\top \cdot \mathbf{z} = \sum_j a_j \cdot z_j = 0 \in \mathbf{R}_q$$

Numerous studies (Lyubashevsky and Micciancio 2006; Lyubashevsky 2008; 2012; Peikert and Rosen 2006; 2007) have shown that if $f(x)$ is irreducible polynomial with integer coefficients, $m > \frac{\log q}{\log(2\beta)}, \gamma = 16mn \log^2 n, q \geq \frac{\gamma\sqrt{n}}{4 \log n}$, then the problem ring-$\mathbf{SIS}^\infty_{n,m,q,\beta}$ is at least as difficult as the problem ring-$\mathbf{SVP}^\infty_\gamma$ over $\mathcal{I}$.

Definition 3 (The ring-**LWE** distribution) *(Peikert 2016) For secret element* $s \in \mathbf{R}_q$*,* $\mathcal{X}$ *is the noise distribution in* $\mathbf{R}_q$ *with bound* β*, choose a* $\xleftarrow{\$} \mathbf{R}_q$*,* $e \xleftarrow{\$} \mathcal{X}$ *uniformly, then* $A_{s,\mathcal{X}} = (a, b = s \cdot a + e \mod q)$ *is called the ring-*$\mathbf{LWE}$ *distribution in* $\mathbf{R}_q \times \mathbf{R}_q$*.*

Definition 4 (The decision ring-$\mathbf{LWE}_{n,m,q,\mathcal{X}}$) *(Lyubashevsky et al. 2010; Peikert 2016) Let* $n, m \geq 1, q \geq 2$*, given m samples* $(a_j, b_j) \in \mathbf{R}_q \times \mathbf{R}_q$*, which are sampled from one of the two distributions:* $A_{s,\mathcal{X}}$ *and the uniform distribution in* $\mathbf{R}_q \times \mathbf{R}_q$*, then the decision ring-*$\mathbf{LWE}_{n,m,q,\mathcal{X}}$ *is to distinguish which one the samples are from.*

Theorem 1 *(Lyubashevsky et al. 2010) Let* $q = 1 \mod 2n, \beta \geq \omega\left(\sqrt{n \log n}\right), \gamma = n^2 \left(\frac{q}{\beta}\right) \left(\frac{nm}{\log(nm)}\right)^{1/4}$*, then there is an error distribution* $\mathcal{X}$ *with bound* β*, such that*

the problem ring-$\mathbf{LWE}_{n,m,q,\mathcal{X}}$ *is at least as difficult as the problem ring-*$\mathbf{SVP}_{\gamma}^{\infty}$ *over* $\mathcal{I}$.

The Merkle hash tree and its dynamic comstruction

The construction of Merkle tree used in the group signature scheme is based on the collision-resistant hash functions. For arbitrary positive integer t, let $\mathbf{G} = \left(1, 2, 4, \cdots, 2^{k-1}\right), \mathbf{bin}(t)$ is the binary representation of t, then $t = \mathbf{G} \cdot \mathbf{bin}(t)$. let $\mathcal{H} = \left\{h_{\mathbf{A}} | \mathbf{A} \xleftarrow{\$} \mathbf{R}_q^m\right\}$, $h_{\mathbf{A}} : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^k$ is collision-resistant hash functions based on the problem ring-$\mathbf{SIS}_{n,m,q,\beta}$, where $\mathbf{A} = [\mathbf{A}_0 | \mathbf{A}_1] \in \mathbf{R}_q^m$, $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathbf{R}_q^k$, for arbitrary $(\mathbf{u}_0, \mathbf{u}_1) \in \{0,1\}^k \times \{0,1\}^k$, we have

$$h_{\mathbf{A}}(\mathbf{u}_0, \mathbf{u}_1) = \mathbf{bin}(\mathbf{A}_0 \cdot \mathbf{u}_0 + \mathbf{A}_1 \cdot \mathbf{u}_1 \mod q) \in \{0,1\}^k$$

so the following equivalent relationship is true,

$$h_{\mathbf{A}}(\mathbf{u}_0, \mathbf{u}_1) = \mathbf{u} \Leftrightarrow \mathbf{A}_0 \cdot \mathbf{u}_0 + \mathbf{A}_1 \cdot \mathbf{u}_1 = \mathbf{G} \cdot \mathbf{u} \mod q$$

Let $\mathcal{H} = \{h_{\mathbf{A}} | \mathbf{A} \in \mathbf{R}_q^m\}$, then we give the following specific description of the dynamic updating algorithm $\mathbf{TDA}(t, \mathbf{d}^*)$ to construct and update the Merkle tree that is used to record the registered users and partial group information in this paper:

> **TSetup**: Initialize the Merkle tree as an empty tree with depth 1, and its root is $\mathbf{u}$. Let t denote the number of legal members in the group.
> **TJoin**: Search for the first non-zero leaf in all leaves, and assume that its index is $i \le t$. Include an empty tree with depth $j = \lceil \log t \rceil$ into the original one if there is not a such leaf. And take its root $\mathbf{u}_{t,1}$ and the root $\mathbf{u}_{t,0}$ of the original tree as two inputs of the hash function to compute a new root $\mathbf{u} = h_{\mathbf{A}}(\mathbf{u}_{t,0}, \mathbf{u}_{t,1})$ of the new Merkle tree. In other words, the original tree and the empty tree are two children of the new Merkle tree with depth $j+1$. And for any $i \in \left[2^{j+1}\right]$, we have $|\mathbf{bin}(i)| = j+1$.
> **TUpdate**: Let $\mathbf{u}_{j+1} = \mathbf{d}^*$ denote the value of the leaf corresponding to the ith user, $\mathbf{bin}(i) = (i_1, \cdots, i_{j+1})$ is the binary description of integer i, its witness is $w = (\mathbf{bin}(i), (\mathbf{w}_{j+1}, \cdots, \mathbf{w}_1))$. Update the value of notes recursively in the path $\mathbf{u}_j, \cdots, \mathbf{u}_0$ from the leaf $\mathbf{u}_{j+1}$ to root $\mathbf{u}$, then output the witness w, a new root $\mathbf{u}_{new}$, where $\mathbf{w}_{j+1}, \cdots, \mathbf{w}_1$ and $\mathbf{u}_j, \cdots, \mathbf{u}_0$ satisfy the following relationship
>
> $$\forall l \in \{j, \cdots, 1, 0\}, \mathbf{u}_l = \begin{cases} h_{\mathbf{A}}(\mathbf{u}_{l+1}, \mathbf{w}_{l+1}), \ if\ i_{l+1} = 0 \\ h_{\mathbf{A}}(\mathbf{w}_{l+1}, \mathbf{u}_{l+1}), \ if\ i_{l+1} = 1 \end{cases}$$
>
> Let $\mathbf{u}_{new} = \mathbf{u}_0$ be the new root of the Merkle tree.

Given the variable t, the computational complexity of algorithm $\mathbf{TUpdate}(t, \mathbf{d}^*)$ is $O(\log t)$, and it satisfies the following property

Theorem 2 *Suppose that the ring-*$\mathbf{SIS}_{m,q,\beta}^{\infty}$ *is difficult,* $R = \{\mathbf{d}_0, \cdots, \mathbf{d}_t\}$ *be the set of the leaves related to users who have been registered, then the algorithm* $\mathbf{TDA}(t, \mathbf{d}^*)$ *is secure. And given a negligible function* $negl(\lambda)$*, for any* PPT adversary $\mathcal{A}$, the following inequality is true

$$\Pr[(\mathbf{d}^*, \mathbf{w}^*) \leftarrow \mathcal{A}(R, t) : \mathbf{d}^* \notin R, \mathbf{u} = \mathbf{u}_0] \le negl(\lambda)$$

The full dynamic group signature scheme and its security

Generally, there are four participants in a group signature scheme: the trusted third party(TTP): who generates the public parameters and the public-private key of the group manager and the trace manager. The group manager GM_{update}: who is responsible to update the group information and the registration and revocation of users. The trace manager GM_{trace}: given a signature, GM_{trace} is responsible to trace the identity of signer when there is a dispute. The users: who are usually appeared as a signer to sign messages or a verifier to verify signatures. Here, we give some changes of the full dynamic group signature scheme in (Ling et al. 2017), and a revised definition is given as follows:

> $\mathbf{GKeyGen}(\lambda) \to (pp, (\mathbf{mpk}, \mathbf{msk}), (opk, osk))$: On input the security parameter λ, this algorithm outputs the public parameter pp, group public key $gpk = (pp, \mathbf{mpk}, opk)$, and distribute the group secret key $\mathbf{msk}$ to GM_{update}, the tracing secret key osk to GM_{trace}. Initialize the registration list $\mathbf{reg}$ and the group information $\mathbf{info}$ as $\emptyset$, and we assume that they can only be edited by a party knowing $\mathbf{msk}$.
> $\mathbf{UKeyGen}(pp) \to (\mathbf{upk}, \mathbf{usk})$: Given the public parameter pp, this algorithm outputs a user's key pair $(\mathbf{upk}, \mathbf{usk})$.
> $\langle \mathbf{Join}(gpk, \mathbf{upk}), \mathbf{Issue}(gpk, \mathbf{msk}, \mathbf{reg}, \mathbf{info}) \rangle$: This algorithm is an interactive protocol between a user and the group manager GM_{update}. Assume that the new registered user is the tth member in the group, the user become a legitimate member of the group if the algorithm goes well, and the $\mathbf{Join}$ algorithm sets its signing secret key $gsk = (\mathbf{bin}(t), \mathbf{upk}_t, \mathbf{usk}_t)$. For the $\mathbf{Issue}$ algorithm, GM_{update} runs the algorithm $TDA(t, \mathbf{upk}_t)$ to update the Merkle hash tree, the group information $\mathbf{info}_\tau$, and the registered user list $\mathbf{reg}$.
> $\mathbf{Revoke}(gpk, S, \mathbf{msk}, \mathbf{reg}, \mathbf{info}_\tau) \to \mathbf{info}_{\tau_{new}}$: Given the revocation list S, for any $i \in S$, the group manager GM_{update} runs algorithm $\mathbf{TUpdate}\left(\mathbf{bin}(i), 0^k\right)$ to update the Merkle hash tree, the registered user list $\mathbf{reg}$ and the group information $\mathbf{info}_{\tau_{new}}$.
> $\mathbf{Sign}(gpk, \mathbf{gsk}_i, \mathbf{info}_\tau, M) \to \Sigma$: On input group public key gpk, group information $\mathbf{info}_\tau$, this algorithm outputs a signature Σ to a message M signed by the user corresponding to ith leaf at τ or an error symbol

$\perp$ if the user is illicit at τ, i.e. the user has not been registered or has been revoked at τ.

Verify$(gpk, \Pi_{sign}, \mathbf{info}_\tau, M) \rightarrow 0/1$: Verify the signature Σ and output 1 if it is valid, otherwise output 0.

Trace$(gpk, osk, M, \Sigma, \mathbf{reg}, \mathbf{info}_\tau) \rightarrow (\mathbf{b}6', \Pi_{trace})$: This algorithm is operated by the trace manager GM_{trace}, it outputs the public key $\mathbf{b}'$ of the signer who signed the message M at τ and generate a proof for this fact if the signature Σ is valid. Otherwise output $\perp$.

Judge$(gpk, \mathbf{b}', M, \Pi_{trace}, \Sigma, \mathbf{info}_\tau) \rightarrow 0/1$: Verify the proof Π_{trace} generated by the trace manager GM_{trace}, and output 1 if it is valid, otherwise output 0.

To verify that whether the signer is legitimate or not, i.e. the signer has registered and not be revoked when he signs a message M at τ, the group manager verifies that whether the value of the leaf corresponding to this signer is non-zero. And to avoid leaking any information about the signer's identity, we bring to the extension-permutation technology to hide it. In other words, suppose that the binary representation of the value of the leaf that corresponding to the signer is $\mathbf{bin}(\mathbf{d}_i) = (d_{i1}, d_{i2}, \cdots, d_{ik}), i \in [t]$, choose a vector $\mathbf{a} \xleftarrow{\$} \{0,1\}^{k-1}$ uniformly such that the Hamming weight of $\mathbf{d}_i' = (\mathbf{bin}(\mathbf{d}_i)|\mathbf{a}) \in \{0,1\}^{2k-1}$ is k. Given $\mathcal{S}_{2k-1} = \{\pi_{2k-1} | \pi_{2k-1}$ *is a random permutation of elements in* $\{0,1\}^{2k-1}\}, \pi_{2k-1} \in \mathcal{S}_{2k-1}$, we have

$$\textit{the Hamming weigth of } \pi_{2k-1}\left(\mathbf{d}_i'\right) \textit{ is } k \Leftrightarrow \mathbf{d}_i \neq 0$$

Moreover, the full dynamic group signature scheme needs to satisfies the following properties: correctness, anonymity, non-frameability, traceability, and tracking soundness.

Correctness: This property means that if the signer signs a message honestly, the algorithm **Verify** can always output 1, the trace manager GM_{trace} can trace the identity of the signer by the algorithm **Trace**, and generates a proof Π_{trace} accepted by the algorithm **Judge**.

Anonymity: For any PPT adversary $\mathcal{A}$, this property means that it is impossible to distinguish signatures generated by two legitimate users with a non-negligible probability, even though the adversary $\mathcal{A}$ could learn the secret key *msk* of GM_{update}, corrupt some of the users, and is given the access to the oracle **Trace**.

Non-frameability: For any PPT adversary $\mathcal{A}$, the probability to generate a valid signature that traced to a legitimate user is negligible, even though the adversary $\mathcal{A}$ could learn the secret keys of GM_{update} and GM_{trace}, and corrupt some of the users.

Traceability: For any PPT adversary $\mathcal{A}$, the probability to generate a valid signature that traced to a illicit user is negligible, even though the adversary $\mathcal{A}$ could learn the secret key of GM_{trace} and corrupt some of the users.

Tracing soundness: For any PPT adversary $\mathcal{A}$, the probability to generate a valid signature that traced to two different users is negligible, even though the adversary $\mathcal{A}$ could learn the secret keys of GM_{update} and GM_{trace}, and corrupt some of the users.

The efficient full dynamic group signature scheme

By using the dynamic algorithm to construct the Merkle hash tree and the formal definition of the full dynamic group signature scheme, the specific construction of the scheme in this paper could be defined as follows:

GKeyGen(λ): Given the security parameter λ, this algorithm is operated by a trusted third party, let $t > 0$ denote the number of registered users, $l = \lceil \log t \rceil$, $n = O(\lambda)$, prime modules $q = \tilde{O}(n^{1.5})$, $k = \lceil \log q \rceil$, $m = 2k$, real integer $\beta > 0$, $\mathcal{X}$ is the noise distribution bounded by β in $\mathbf{R}$, $k' = \omega(\log \lambda)$. $H : \{0,1\}^* \rightarrow \{0,1\}^{k'}$ is a hash function for FS transformation, and $Com : \{0,1\}^* \times \{0,1\}^m \rightarrow \mathbf{Z}_q^n$ is a string commitment scheme with properties of statistical hiding and computational binding (Kawachi et al. 2008). Choose a matrix $\mathbf{A} \xleftarrow{\$} \mathbf{R}_q^m$ uniformly, for any $j \in \{1,2\}$, TTP chooses $\mathbf{S}_j \xleftarrow{\$} \mathcal{X}^k$, $E_j \xleftarrow{\$} \mathcal{X}$, $\mathbf{B} \xleftarrow{\$} \mathbf{R}_q^k$, $\mathbf{msk} \xleftarrow{\$} \mathbf{R}^m$ uniformly, and computes the public keys $P_i = \mathbf{S}_i^\top \mathbf{B} + E_i \in \mathbf{R}_q$, $\mathbf{mpk} = \mathbf{A} \times \mathbf{msk}$. Output the public parameter $pp = (\lambda, n, q, k, m, \beta, \mathcal{X}, k', H, Com, \mathbf{A})$, the tracing public key $opk = (\mathbf{B}, P_1, P_2)$, the group public key $gpk = (pp, \mathbf{mpk}, opk)$. And distribute the tracing secret key $osk = (\mathbf{S}_1, E_1)$ to GM_{trace}, the group secret key **msk** to GM_{update}. Initialize the registration list **reg** and the group information **info** as $\emptyset$, and we assume that they can only be edited by a party knowing **msk**.

UKeyGen(pp): The user chooses $\mathbf{usk} \xleftarrow{\$} \mathbf{R}^m$ uniformly as its secret key, and computes the related public key $\mathbf{upk} = \mathbf{bin}(\mathbf{A} \cdot \mathbf{usk}) \mod q \in \{0,1\}^k$.

$\langle$**Join**$(gpk, \mathbf{upk})$, **Issue**$(gpk, \mathbf{msk}, \mathbf{reg}, \mathbf{info})\rangle$: Assume that the new registered user is the tth member in the group, and the user runs algorithm **Join**, sends its public key **upk** to the group manager GM_{update}, and if this algorithm goes well, the algorithm **Issue** searches and denotes the first non-zero leaf as t' if he approves the user's application. Let $\mathbf{upk}_{t'} = \mathbf{upk}$, $\mathbf{reg}_{t'} = \mathbf{reg}_{t'}[\mathbf{upk}_{t'}][\tau]$, τ is the time the user registered, the algorithm **Issue** includes $\mathbf{reg}_{t'}$ into the registration list $\mathbf{reg} := (\mathbf{reg}_1[\mathbf{upk}_1][\tau], \cdots, \mathbf{reg}_{t'}[\mathbf{upk}_{t'}][\tau], \cdots,$

$\mathbf{reg}_t[\mathbf{upk}_t][\tau])$. Then the group manager GM_{update} runs the algorithm $TDA(\mathbf{bin}(t'), \mathbf{upk}_{t'})$ to update the Merkle tree, outputs the group information $\mathbf{info}_\tau = (\mathbf{u}, \{\mathbf{w}_j\}_{i_j})$ where $\mathbf{u}$ is the root and $\{\mathbf{w}_j\}_{i_j}$ are witnesses of all legal users, and updates the counter of registered users $t = t + 1$. Let $\mathbf{usk}_{t'} = \mathbf{usk}$, the user sets $gsk_{t'} = (\mathbf{bin}(t'), \mathbf{upk}_{t'}, \mathbf{usk}_{t'})$ as its signing secret key.

Revoke$(gpk, S, \mathbf{msk}, \mathbf{reg}, \mathbf{info}_\tau)$: Given the revocation list S that is the set of public keys of group members who would be revoked, if $S = \{\mathbf{upk}_{i_1}, \cdots, \mathbf{upk}_{i_r}\}$ is not an empty set, $i_j \in [t]$, $j \in [r]$, for every $j \in [r]$, $\mathbf{upk}_{i_j} \in S$, GM_{update} runs the algorithm **TUpdate** in $TDA(\mathbf{bin}(i_j), 0^k)$ to update the Merkle hash tree, then updates the registration list **reg**: changes $\mathbf{reg}_{i_j}[\mathbf{upk}_{i_j}][\tau]$ to $\mathbf{reg}_{i_j}[0^k][\tau_{new}]$ if $\mathbf{upk}_{i_j} \in S$, otherwise changes $\mathbf{reg}_{i_j}[\mathbf{upk}_{i_j}][\tau]$ to $\mathbf{reg}_{i_j}[\mathbf{upk}_{i_j}][\tau_{new}]$, outputs the new group information $\mathbf{info}_{\tau_{new}} = (\mathbf{u}_{new}, \{\mathbf{w}_j\}_{i_j})$ that consists of a new root $\mathbf{u}_{new}$ and witnesses $\{\mathbf{w}_j\}_{i_j}$ of $\mathbf{upk}_{i_j}$, updates the counter of legitimate users $t = t - r$. So, the leaves with value 0^k in the Merkle tree corresponding to the potential users who have not been registered or those have been revoked.

Sign$(gpk, gsk_i, \mathbf{info}_\tau, M)$: To sign a message M at τ by using the group information $\mathbf{info}_\tau$, the user related to the ith leaf verifies that whether there is a witness of $\mathbf{bin}(i)$ in $\mathbf{info}_\tau$ firstly, if not, return $\perp$. Otherwise, the user obtains $(\mathbf{bin}(i), (\mathbf{w}_l, \cdots, \mathbf{w}_1))$ from $\mathbf{info}_\tau$ to do the follows: For each $j \in \{1, 2\}$, random string $\mathbf{r}_j \xleftarrow{\$} \{0,1\}^k$, the user encrypts vector $\mathbf{upk}_i$ by making use of the double-encryption paradigm (Naor and Yung 1990) and the RLWE-based encryption scheme (Regev 2009; Lyubashevsky et al. 2013) to obtain the ciphertext

$$\mathbf{c}_j = (c_{j,1}, \mathbf{c}_{j,2}) = \Big(\mathbf{B}\cdot\mathbf{r}_j \mod q, P_j\cdot\mathbf{r}_j + \Big\lceil\frac{q}{2}\Big\rfloor\cdot\mathbf{upk}_i \mod q\Big) \in \mathbf{R}_q \times \mathbf{R}_q^k$$

Then the user generates a non-interactive zero-knowledge argument of knowledge(NIZKAoK) Π_{sign} for:

(1) it has legitimate witness $\zeta = (\mathbf{usk}_i, \mathbf{upk}_i, \mathbf{bin}(i), \mathbf{w}_l, \cdots, \mathbf{w}_1, \mathbf{r}_1, \mathbf{r}_2)$ such that the signer is a legitimate member in the group, i.e. $\mathbf{upk}_i \neq 0^k$, and the values of nodes in the path that from the leaf corresponding to the user to the root are all correct.
(2) $(\mathbf{usk}_i, \mathbf{upk}_i)$ is a valid public-private key-pair.
(3) $(\mathbf{c}_1, \mathbf{c}_2)$ are two legitimate ciphertext of $\mathbf{upk}_i$.

Finally, the signer outputs the signature $\Sigma = ((\mathbf{c}_1, \mathbf{c}_2), \Pi_{sign})$. The NIZK argument of knowledge mentioned above is obtained from the Stern's three-round interactive protocol (Song 2001) by FS transformation, i.e. runs the Stern protocol k' times sequentially to obtain a negligible soundness error, and the transcript is $\Pi_{sign} = \Big(\{CMT_j\}_{j=1}^{k'}, CH, \{RSP_j\}_{j=1}^{k'}\Big)$, where

$$CH = H\Big(M, \{CMT_j\}_{j=1}^{k'}, \mathbf{A}, \mathbf{u}_\tau, \mathbf{B}, P_1, P_2, \mathbf{c}_1, \mathbf{c}_2\Big) \in \{1,2,3\}^{k'}$$

Verify$(gpk, \Pi_{sign}, \mathbf{info}_\tau, M)$: The verifier obtains the root $\mathbf{u}_\tau$ of the Merkle hash tree at τ from the group information $\mathbf{info}_\tau$, and verifies that whether the predicted challenge CH is true, outputs 0 if not, otherwise verifies the respond RSP_j that corresponding to CMT_j and CH_j for each $j \in [k']$, and outputs 1 if everything is correct, otherwise outputs 0.

Trace$(gpk, osk, M, \Sigma, \mathbf{reg}, \mathbf{info}_\tau)$: The trace manager GM_{trace} uses its tracing secret key osk to decrypt the ciphertext $\mathbf{c}_1 = (c_{1,1}, \mathbf{c}_{1,2})$ and compute $\mathbf{b}' = \left\lfloor \frac{(\mathbf{c}_{1,2} - S_1^\top \cdot c_{1,1})}{q/2} \right\rceil \in \{0,1\}^k$. If there is not a witness of $\mathbf{b}'$ in $\mathbf{info}_\tau$ or $\mathbf{b}' = 0^k$, output $\perp$. Then GM_{trace} generates a non-interactive zero-knowledge argument of knowledge(NIZKAoK) Π_{trace} for the fact that the user corresponding to $\mathbf{b}'$ really generated a signature Σ to message M at τ. In other words, the trace manager GM_{trace} should proof that he has $\mathbf{S}_1 \in \mathbf{R}_q^k, E_1 \in \mathbf{R}_q, \mathbf{y} \in \mathbf{R}_q^k$, such that

$$\|\mathbf{S}_1\|_\infty \leq \beta, |E_1| \leq \beta, \|\mathbf{y}\|_\infty \leq \Big\lceil\frac{q}{5}\Big\rceil$$
$$\mathbf{S}_1^\top \cdot \mathbf{B} + E_1 = P_1 \mod q$$
$$\mathbf{c}_{1,2} - \mathbf{S}_1^\top \cdot c_{1,1} = \mathbf{y} + \Big\lfloor\frac{q}{2}\Big\rfloor \cdot \mathbf{b}' \mod q$$

Similarly, the NIZKAoK mentioned above is obtained from the Stern's three-round interactive protocol (Song 2001) by FS transformation, i.e. GM_{trace} runs the Stern protocol k' times sequentially to obtain a negligible soundness error, and the transcript is $\Pi_{trace} = \Big(\{CMT_j\}_{j=1}^{k'}, CH, \{RSP_j\}_{j=1}^{k'}\Big)$, where

$$CH = H(M, \{CMT_j\}_{j=1}^{k'}, gpk, \Sigma, \mathbf{info}_\tau, \mathbf{b}') \in \{1,2,3\}^{k'}$$

Finally, this algorithm outputs $(\mathbf{b}', \Pi_{trace})$.

Judge$(gpk, \mathbf{b}', M, \Pi_{trace}, \Sigma, \mathbf{info}_\tau)$: Verify the proof Π_{trace} and output 1 if it is true, otherwise output 0.

In this scheme, the public parameter and the public-private key pair are all generated by a trusted third party, which can avoid the problem that the illegitimate group managers generate their keys maliciously, but not the malpractices of the legitimate group man-

agers. This is one possible attack on this type of scenario that we can think of, such as group members can be added or withdrawn according to a group manager's personal preference or interest relationship. To this problem, we can consider to set up the group manager a trust value TV, a confidence threshold CT, and a reduction coefficient RC, where the value of TV is initialized to $tv = 1, 0 < CT$, and $RC < 1$. The value of TV is reduced to $TV_s = tv - s \cdot RC$ if the group manager has s times malpractices, and it would be revoked if $TV_s < CT$.

Furthermore, it is not necessary to prepare a large storage space for a large empty tree standby before a signature is generated, namely we only need to extend or update the Merkle hash tree when a user needs a registration or be revoked. Compared with the scheme in (Ling et al. 2017), our work could realizes the truly dynamic of the group signature scheme, which helps to economize considerable storage space, and there is also no limits on the upper bound of the size of the group as long as the storage space is allowed. In addition, the fact that the scheme is implemented based on ring could help to reduce the computational complexity and space complexity of it.

Finally, a timestamp τ is given to each member in the group, the group manager GM_{update} updates the group information $\mathbf{info}_\tau$ once a new user registered or a legitimate member has been revoked, which indicates that the user can not sign a message M before a registration or after a revocation. Given a group information $\mathbf{info}_\tau$, we can confirm the timestamp τ uniquely, and vice versa. For any two timestamps $\tau_1 < \tau_2$, the group information $\mathbf{info}_{\tau_1}$ is published earlier than $\mathbf{info}_{\tau_2}$.

The underlying protocol

The definition of the underlying protocol

Suppose that the size of the legitimate members in the group is $t \geq 1$ at time τ, for any $b \in \{1,2\}$, $i \in [t]$, $\forall j \in [l-1]$, the underlying zero-knowledge protocol is used to proof the following relationship by utilizing the Stern's protocol (Song 2001)

$$\begin{cases} \mathbf{upk}_i \neq 0 \\ \mathbf{u}_j = \begin{cases} h_{\mathbf{A}}(\mathbf{u}_{j+1}, \mathbf{w}_{j+1}), \ if\ i_{j+1} = 0 \\ h_{\mathbf{A}}(\mathbf{w}_{j+1}, \mathbf{u}_{j+1}), \ if\ i_{j+1} = 1 \end{cases} (\star) \\ \mathbf{upk}_i = \mathbf{bin}(\mathbf{A} \cdot \mathbf{usk}_i) \\ \mathbf{c}_b = (c_{b,1}, \mathbf{c}_{b,2}) = \left(\mathbf{B} \cdot \mathbf{r}_b, P_b \cdot \mathbf{r}_b + \left\lfloor \frac{q}{2} \right\rceil \cdot \mathbf{upk}_i\right) \end{cases} \tag{1}$$

Given a bit b, a vector $\mathbf{a}$, let $\mathbf{ext}(b, \mathbf{a}) = (\bar{b} \cdot \mathbf{a}, b \cdot \mathbf{a})^\top$, $\mathbf{ext}_2(b) = (\bar{b}, b)^\top$, then we have the following equivalence relationship:

$$\begin{aligned} (\star) &\Leftrightarrow \bar{i}_{j+1} \cdot h_{\mathbf{A}}(\mathbf{u}_{j+1}, \mathbf{w}_{j+1}) + i_{j+1} \cdot h_{\mathbf{A}}(\mathbf{w}_{j+1}, \mathbf{u}_{j+1}) = \mathbf{u}_j \\ &\Leftrightarrow \bar{i}_{j+1}(\mathbf{A}_0\mathbf{u}_{j+1} + \mathbf{A}_1\mathbf{w}_{j+1}) + i_{j+1}(\mathbf{A}_0\mathbf{w}_{j+1} + \mathbf{A}_1\mathbf{u}_{j+1}) = \mathbf{G} \cdot \mathbf{u}_j \mod q \\ &\Leftrightarrow \mathbf{A} \cdot \begin{pmatrix} \bar{i}_{j+1} \cdot \mathbf{u}_{j+1} \\ i_{j+1} \cdot \mathbf{u}_{j+1} \end{pmatrix} + \mathbf{A} \cdot \begin{pmatrix} i_{j+1} \cdot \mathbf{w}_{j+1} \\ \bar{i}_{j+1} \cdot \mathbf{w}_{j+1} \end{pmatrix} = \mathbf{G} \cdot \mathbf{u}_j \mod q \\ &\Leftrightarrow \mathbf{A} \cdot \mathbf{ext}(i_{j+1}, \mathbf{u}_{j+1}) + \mathbf{A} \cdot \mathbf{ext}(\bar{i}_{j+1}, \mathbf{w}_{j+1}) = \mathbf{G} \cdot \mathbf{u}_j \mod q \end{aligned}$$

Then for any $b \in \{1,2\}, i \in [t], \mathbf{bin}(i) = (i_1, \cdots, i_l)$, the Eq. 1 is equal to the following form

$$\begin{cases} \mathbf{A} \cdot \mathbf{ext}(i_1, \mathbf{u}_1) + \mathbf{A} \cdot \mathbf{ext}(\bar{i}_1, \mathbf{w}_1) - \mathbf{G} \cdot \mathbf{u} = 0 \mod q \\ \mathbf{A} \cdot \mathbf{ext}(i_2, \mathbf{u}_2) + \mathbf{A} \cdot \mathbf{ext}(\bar{i}_2, \mathbf{w}_2) - \mathbf{G} \cdot \mathbf{u}_1 = 0 \mod q \\ \qquad \cdots \\ \mathbf{A} \cdot \mathbf{ext}(i_l, \mathbf{upk}_i) + \mathbf{A} \cdot \mathbf{ext}(\bar{i}_l, \mathbf{w}_l) - \mathbf{G} \cdot \mathbf{u}_{l-1} = 0 \mod q \\ \mathbf{A} \cdot \mathbf{usk}_i - \mathbf{G} \cdot \mathbf{upk}_i = 0 \mod q \\ c_{b,1} = \mathbf{B} \cdot \mathbf{r}_b \mod q \\ \mathbf{c}_{b,2} = P_b \cdot \mathbf{r}_b + \left\lfloor \frac{q}{2} \right\rceil \cdot \mathbf{upk}_i \mod q \end{cases}$$

Let $\mathbf{B}_n^{2n}$ be the set of strings with length $2n$, where the Hamming weight of each string is n, to illustrate the fact that the user's public key $\mathbf{upk}_i \neq 0^k$, we pad $\mathbf{upk}_i$ with a random string with length $k-1$ to obtain a new string $\mathbf{upk}_i^*$, such that $\mathbf{upk}_i^* \in \mathbf{B}_k^{2k-1}$, then for any permutation $\pi_{\mathbf{upk}_i} \in \mathcal{S}_{2k-1}$, we have

$$\mathbf{upk}_i \neq 0^k \Leftrightarrow \mathbf{upk}_i^* \in \mathbf{B}_k^{2k-1} \Leftrightarrow \pi_{\mathbf{upk}_i}\left(\mathbf{upk}_i^*\right) \in \mathbf{B}_k^{2k-1}$$

We make similar operations for each $\mathbf{usk}_i$ to obtain $\mathbf{usk}_i^* \in \mathbf{B}_m^{2m}$, for any $\pi_{\mathbf{upk}_i} \in \mathcal{S}_{2m}$, we have $\mathbf{usk}_i^* \in \mathbf{B}_m^{2m} \Leftrightarrow \pi_{\mathbf{usk}_i}(\mathbf{usk}_i^*) \in \mathbf{B}_m^{2m}$. Similarly, extend the vectors $\mathbf{u}_1, \cdots, \mathbf{u}_{l-1}, \mathbf{w}_1, \cdots, \mathbf{w}_l, \mathbf{r}_1, \mathbf{r}_2$ to obtain $\mathbf{u}_1^* \cdots, \mathbf{u}_{l-1}^*$, $\mathbf{w}_1^* \cdots, \mathbf{w}_l^* \in \mathbf{B}_k^{2k}$, $\mathbf{r}_1^*, \mathbf{r}_2^* \in \mathbf{B}_k^{2k}$. And then let $\hat{\mathbf{u}}_1 = \mathbf{ext}(i_1, \mathbf{u}_1^*), \cdots, \hat{\mathbf{u}}_{l-1} = \mathbf{ext}\left(i_{l-1}, \mathbf{u}_{l-1}^*\right) \in \{0,1\}^{4k}$, $\hat{\mathbf{upk}}_i = \mathbf{ext}\left(i_l, \mathbf{upk}_i^*\right) \in \{0,1\}^{4k-2}$, $\hat{\mathbf{w}}_1 = \mathbf{ext}\left(\bar{i}_1, \mathbf{w}_1^*\right), \cdots, \hat{\mathbf{w}}_l = \mathbf{ext}\left(\bar{i}_l, \mathbf{w}_l^*\right) \in \{0,1\}^{4k}$.

Given $\mathbf{upk}_i = (upk_{i1}, \cdots, upk_{ik})$, for any $j \in [k]$, let $\mathbf{upk}'_{ij} = \mathbf{ext}_2(upk_{ij})$. For any $b \in \{0,1\}$, $\mathbf{t} = (t_0, t_1) \in \mathbf{Z}^2$, let $T_b(\mathbf{t}) = \left(t_b, t_{\bar{b}}\right)$. Then for any $b_j \in \{0,1\}$, we have $\mathbf{upk}'_{ij} = \mathbf{ext}_2(upk_{ij}) \Leftrightarrow T_{b_j}\left(\mathbf{upk}'_{ij}\right) = \mathbf{ext}_2(upk_{ij} \oplus b_j)$. Because b_j is chosen randomly, so the operations above are equal to carry out a one-time pad to the user's upk_{ij} by b_j to hide it perfectly.

Let $r \in \{2k-1, 2k\}, b \in \{0,1\}, \pi \in \mathcal{S}_r, \mathbf{t} = (t_0, t_1)^T \in \mathbf{Z}^{2r}$, we define the permutation $F_{b,\pi}(\mathbf{t}) = (\pi(t_b), \pi(t_{\bar{b}}))$. Then for all $b_1, \cdots, b_l \in \{0,1\}, \phi_{u,1}, \cdots, \phi_{u,l-1}, \phi_{w,1}, \cdots, \phi_{w,l} \in \mathcal{S}_{2k}, \pi_{upk_i} \in \mathcal{S}_{2k-1}$, the following relationship is true,

$$\begin{cases} \forall j \in [l-1], \hat{\mathbf{u}}_j = \mathbf{ext}\left(i_j, \mathbf{u}_j^*\right) \Leftrightarrow F_{b_j,\phi_{u,j}}(\hat{\mathbf{u}}_j) = \mathbf{ext}\left(i_j \oplus b_j, \phi_{u,j}\left(\mathbf{u}_j^*\right)\right) \\ \forall j \in [l], \hat{\mathbf{w}}_j = \mathbf{ext}\left(i_j, \mathbf{w}_j^*\right) \Leftrightarrow F_{b_j,\phi_{w,j}}(\hat{\mathbf{w}}_j) = \mathbf{ext}\left(i_j \oplus b_j, \phi_{w,j}\left(\mathbf{w}_j^*\right)\right) \\ \hat{\mathbf{upk}}_i = \mathbf{ext}\left(i_l, \mathbf{upk}_i^*\right) \Leftrightarrow F_{b_l,\pi_{upk_i}}\left(\hat{\mathbf{upk}}_i\right) = \mathbf{ext}\left(i_l \oplus b_l, \pi_{upk_i}\left(\mathbf{upk}_i^*\right)\right) \end{cases} \tag{2}$$

Let

$$\mathbf{z} = \Big(\mathbf{u}_1^* | \hat{\mathbf{u}}_1 | \hat{\mathbf{w}}_1 | \cdots | \mathbf{u}_{l-1}^* | \hat{\mathbf{u}}_{l-1} | \hat{\mathbf{w}}_{l-1} | \mathbf{upk}_i^* | \hat{\mathbf{upk}}_i | \hat{\mathbf{w}}_l | \mathbf{usk}_i^* | \mathbf{r}_1^* | \mathbf{r}_2^* | \mathbf{upk}_{i1}' | \cdots | \mathbf{upk}_{ik}' \Big)$$

then $\mathbf{z} \in \{0,1\}^{10kl+2m+6k-3}$, the Eq. 2 can be unified into one equation $\mathbf{A}' \cdot \mathbf{z} = \mathbf{U} \mod q$, where $\mathbf{A}', \mathbf{U}$ could be obtained from the public parameters. Let **VALID** be the set of vectors in $\{0,1\}^{10kl+2m+6k-3}$ that satisfy the relationship above, let

$$\bar{\mathcal{S}} = \mathcal{S}_{2k}^{2l-1} \times \mathcal{S}_{2k-1} \times \mathcal{S}_{2m} \times \mathcal{S}_{2l}^2 \times \{0,1\}^l$$

for any

$$\eta = \big((\phi_{u,1}, \cdots, \phi_{u,l-1}, \phi_{w,1}, \cdots, \phi_{w,l}), \pi_{upk_i}, \pi_{usk_i}, (\pi_{r,1}, \pi_{r,2}), (b_1, \cdots, b_l)\big) \in \bar{\mathcal{S}}$$

let Γ_η be the permutation for strings in $\{0,1\}^{10kl+2m+6k-3}$, then we have

$$\mathbf{z} \in \mathbf{VALID} \Leftrightarrow \Gamma_\eta(\mathbf{z}) \in \mathbf{VALID}$$

After that, we could utilize the Stern's protocol and the equal relationship above to proof that $\mathbf{z} \in \mathbf{VALID}$, and $\mathbf{A}' \cdot \mathbf{z} = \mathbf{U} \mod q$. Let $D = 10kl + 2m + 6k - 3$, the underlying zero-knowledge argument of knowledge is as follows,

Algorithm 1: The underlying zero knowledge argument of knowledge

Commitment: The prover P chooses $\mathbf{r}_z \in \mathbf{Z}_q^D, \eta \in \bar{\mathcal{S}}$, and $\rho_1, \rho_2, \rho_3 \in \{0,1\}^m$ uniformly and randomly, and computes the commitments $C_1 = Com(\eta, \mathbf{A}' \cdot \mathbf{r}_z; \rho_1), C_2 = Com(\Gamma_\eta(\mathbf{r}_z); \rho_2)$, $C_3 = Com(\Gamma_\eta(\mathbf{z} + \mathbf{r}_z); \rho_3)$ respectively. Finally, sends the commitment $CMT = (C_1, C_2, C_3)$ to the verifier V.
Challenge: V chooses a challenge $CH \in \{1,2,3\}$ uniformly and randomly, and sends it to P.
Response: P sends a respond RSP to V depend on the challenge CH,

1. If $CH = 1$, set $\mathbf{t}_z = \Gamma_\rho(\mathbf{z}), \mathbf{t}_r = \Gamma_\rho(\mathbf{r}_z)$, $RSP = (\mathbf{t}_z, \mathbf{t}_r, \rho_2, \rho_3)$.
2. If $CH = 2$, set $\eta_2 = \eta, \mathbf{z}_2 = \mathbf{z} + \mathbf{r}_z$, $RSP = (\eta_2, \mathbf{z}_2, \rho_1, \rho_3)$.
3. If $CH = 3$, set $\eta_3 = \eta, \mathbf{z}_3 = \mathbf{r}_z$, $RSP = (\eta_3, \mathbf{z}_3, \rho_1, \rho_2)$.

Verification: V verifies the proof generated by P depend on the challenge CH and the respond RSP,

1. If $CH = 1$, verify that $\mathbf{t}_z \in \mathbf{VALID}$, $C_2 = Com(\mathbf{t}_r; \rho_2), C_3 = Com(\mathbf{t}_z + \mathbf{t}_r; \rho_3)$.
2. If $CH = 2$, verify that $C_1 = Com(\eta_2, \mathbf{A}' \cdot \mathbf{z}_2 - \mathbf{u}; \rho_1)$, $C_3 = Com(\Gamma_{\eta_2}(\mathbf{z}_2); \rho_3)$.
3. If $CH = 3$, verify that $C_1 = Com(\eta_3, \mathbf{A}' \cdot \mathbf{z}_3; \rho_1)$, $C_2 = Com(\Gamma_{\eta_3}(\mathbf{z}_3); \rho_2)$.

Finally, V outputs 1 if and only if the verification is true.

The security analysis of the underlying protocol

Theorem 3 *Suppose that the problem ring-*$\mathbf{SVP}_{\tilde{O}(n)}$ *is difficult, then the protocol in the previous section satisfies the following properties: perfect completeness, statistical zero knowledge, argument of knowledge, and the soundness error is* $\frac{2}{3}$*, the communication complexity is* $\tilde{O}(D \log q)$.

Proof As to the property of perfect completeness, if participants in the protocol run each step honestly, then V would accepts the proof generated by P with probability 1. Owing to $\mathbf{r}_z \in \mathbf{Z}_q^D, \mathbf{z} \in \{0,1\}^D, \|\mathbf{r}_z\| = \|\mathbf{z}\| = D$, it is easy to verify that the communication complexity is $\tilde{O}(D \log q)$. And next, we will present a detailed description of the property of zero knowledge.

We construct a PPT simulator *Sim* firstly to simulate the real interactions between a honest prover P and a malicious verifier V^*, such that the distribution of the transcript outputted simulator *Sim* is statistical close to that of the real interactions. *Sim* chooses $\bar{CH} \in \{1,2,3\}$ randomly as a prediction of the challenge that the verifier V^* would not choose.

If $\bar{CH} = 1$, *Sim* computes a vector $\mathbf{z}' \in \mathbf{Z}_q^D$ by using the algebraic method , such that $\mathbf{A}' \cdot \mathbf{z}' = \mathbf{u} \mod q$. Then chooses $\mathbf{r}_z \in \mathbf{Z}_q^D, \eta \in \bar{\mathcal{S}}$, and strings $\rho_1, \rho_2, \rho_3 \in \{0,1\}^m$ uniformly and randomly to compute the commitments $C_1' = Com(\eta, \mathbf{A}' \cdot \mathbf{r}_z; \rho_1), C_2' = Com(\Gamma_\eta(\mathbf{r}_z); \rho_2)$, $C_3' = Com(\Gamma_\eta(\mathbf{z}' + \mathbf{r}_z); \rho_3)$, and sends the commitment $CMT = \big(C_1', C_2', C_3'\big)$ to V^*. Depend on the challenge CH that received from V^*, the simulator responds as follows:

1. If $CH = 1$, output $\perp$ and break.
2. If $CH = 2$, let $RSP = (\eta, \mathbf{z}' + \mathbf{r}_z, \rho_1, \rho_3)$ and send it to V^*.
3. If $CH = 3$, let $RSP = (\eta, \mathbf{r}_z, \rho_1, \rho_2)$ and send it to V^*.

If $\bar{CH} = 2$, *Sim* chooses $\mathbf{z}' \in \mathbf{VALID}, \mathbf{r}_z \in \mathbf{Z}_q^D, \eta \in \bar{\mathcal{S}}$, and strings $\rho_1, \rho_2, \rho_3 \in \{0,1\}^m$ uniformly and randomly to compute the commitments $C_1' = Com(\eta, \mathbf{A}' \cdot \mathbf{r}_z; \rho_1), C_2' = Com(\Gamma_\eta(\mathbf{r}_z); \rho_2), C_3' = Com(\Gamma_\eta(\mathbf{z}' + \mathbf{r}_z); \rho_3)$, and sends the commitment $CMT = (C_1', C_2', C_3')$ to the verifier V^*. Depend on the challenge CH that received from V^*, the simulator responds as follows:

1. If $CH = 1$, let $RSP = (\Gamma_\eta(\mathbf{z}'), \Gamma_\eta(\mathbf{r}_z), \rho_2, \rho_3)$ and send it to V^*.
2. If $CH = 2$, output $\perp$ and break.
3. If $CH = 3$, let $RSP = (\eta, \mathbf{r}_z, \rho_1, \rho_2)$ and send it to V^*.

If $\bar{CH} = 3$, *Sim* chooses $\mathbf{z}' \in \mathbf{VALID}, \mathbf{r}_z \in \mathbf{Z}_q^D, \eta \in \bar{\mathcal{S}}$, and strings $\rho_1, \rho_2, \rho_3 \in \{0,1\}^m$ uniformly and randomly, and computes the commitments $C_1' = Com(\eta, \mathbf{A}' \cdot (\mathbf{z}' + \mathbf{r}_z) - \mathbf{u}; \rho_1), C_2' = Com(\Gamma_\eta(\mathbf{r}_z); \rho_2), C_3' = Com(\Gamma_\eta(\mathbf{z}' + \mathbf{r}_z); \rho_3)$, and sends the commitment $CMT = \left(C_1', C_2', C_3'\right)$ to the verifier V^*. Depend on the challenge $\bar{CH}$ that received from V^*, the simulator responds as follows:

1. If $\bar{CH} = 1$, compute *RSP* as in the case $(\bar{CH} = 2, CH = 1)$, and send it to V^*.
2. If $\bar{CH} = 2$, compute *RSP* as in the case $(\bar{CH} = 1, CH = 2)$, and send it to V^*.
3. If $CH = 3$, output $\perp$ and break.

For the commitment scheme is statistical indistinguishable, the distribution of the output of *Sim* and that of the real interactions are statistical indistinguishable. i.e. there is a negligible function $negl(n)$ such that $\Pr[\perp \leftarrow Sim] = \frac{1}{3} \pm negl(n)$. So the simulator would outputs an acceptable transcript as long as no error symbol $\perp$ is outputted, in other words, *Sim* would outputs a transcript that is indistinguishable from that of a real interactions with probability almost $\frac{2}{3}$.

Finally, we would like to give a concrete explanation of the property of argument of knowledge. Suppose that there are three different valid responds $RSP_1 = (\mathbf{t}_z, \mathbf{t}_r, \rho_2, \rho_3), RSP_2 = (\eta_2, \mathbf{z}_2, \rho_1, \rho_3), RSP_3 = (\eta_3, \mathbf{z}_3, \rho_1, \rho_2)$ corresponding to three different challenges of one commitment *CMT*, then the validity of responds indicates the following relationship:

$$\begin{cases} \mathbf{t}_z \in \mathbf{VALID}; C_1 = Com(\eta_2, \mathbf{A}' \cdot \mathbf{z}_2 - \mathbf{u}; \rho_1) = Com(\eta_3, \mathbf{A}' \cdot \mathbf{z}_3; \rho_1); \\ C_2 = Com(\mathbf{t}_r; \rho_2) = Com(\Gamma_{\eta_3}(\mathbf{z}_2); \rho_2); \\ C_3 = Com(\mathbf{t}_z + \mathbf{t}_r; \rho_3) = Com(\Gamma_{\eta_2}(\mathbf{z}_2); \rho_3) \end{cases}$$

Because of the computational binding of the commitment scheme *Com*, we have

$$\begin{cases} \mathbf{t}_z \in \mathbf{VALID}; \eta_2 = \eta_3; \mathbf{t}_r = \Gamma_{\eta_3}(\mathbf{z}_3); \mathbf{t}_z + \mathbf{t}_r = \Gamma_{\eta_2}(\mathbf{z}_2) \mod q; \\ \mathbf{A}' \cdot \mathbf{z}_2 - \mathbf{u} = \mathbf{A}' \cdot \mathbf{z}_3 \mod q \end{cases}$$

For $\mathbf{t}_z \in \mathbf{VALID}$, let $\mathbf{z}' = \Gamma_{\eta_2}^{-1}(\mathbf{t}_z)$, then $\mathbf{z}' \in \mathbf{VALID}$, $\Gamma_{\eta_2}(\mathbf{z}') + \Gamma_{\eta_2}(\mathbf{z}_3) = \Gamma_{\phi_2}(\mathbf{z}_2) \mod q$, and we could learn that $\mathbf{z}' + \mathbf{z}_3 = \mathbf{z}_2, \mathbf{A}' \cdot \mathbf{z}' + \mathbf{A}' \cdot \mathbf{z}_3 = \mathbf{A}' \cdot \mathbf{z}_2 \mod q$, Finally, we obtain a solution $\mathbf{z}'$ to a instance of the problem ring-**SIS**, which satisfies $\mathbf{A}' \cdot \mathbf{z}' = \mathbf{u} \mod q$. □

The analysis of the group signature scheme

Notation

The security of the full dynamic group signature scheme presented in this paper satisfies the strong security definition given in (Bootle et al. 2016): correctness, anonymity, non-frameability, traceability, and tracing soundness. Before the specific description, we would like to give a brief description of oracles and special symbols used in the proof firstly. *HUL* is the set of honest users whose secret keys are generated honesty. *BUL* is the set of users whose signing secret keys are sent to the adversary. *CUL* is the set of users whose public keys are chosen by the adversary. *SL* is the set of signatures generated by oracle **sign**. *CL* is the set of signatures generated by oracle $\mathbf{Chal}_b$. And oracles used in the proof are as follows:

AddU(i): Add an honest user i into the set *HUL* at time τ.
CreU$(i, \mathbf{upk}_i)$: Create a new user i whose public key $\mathbf{upk}_i$ is chosen by the adversary, which is invoked in the oracle **SenToM**.
SenToM(i, M_{in}): It is used to run the algorithm **Join**, on behalf of a corrupt user, together with the honest group manager GM_{update}.
SenToU(i, M_{in}): It is used to run the algorithm **Join**, on behalf of the corrupt group manager GM_{update}, together with a legitimate user i.
RReg(i): Return the registration information $\mathbf{reg}_i$ of user i.
MReg(i, ρ): Change the registration information $\mathbf{reg}_i$ of user i into ρ.
RevealU(i): Return the signing secret key gsk_i of user i to the adversary, and add i to the set *BUL*.
Sign(i, M, τ): Return a signature to a message M signed by user i at time τ, and add this signature to the set *SL*.
$\mathbf{Chal_b}(\mathbf{info}_\tau, i_0, i_1, M)$: For any $b \in \{0,1\}$, Return the signature to a message M signed by user i_b at time τ, and add this signature to the set *CL*. This requires that the users i_0, i_1 are all legitimate at time τ, and this oracle could be revoked only once.
$\mathbf{Trace}(\mathbf{info}_\tau, \Sigma, M)$: Return the signer of a signature Σ signed at time τ and a proof of this fact, which requires that the signature $\Sigma \notin CL$.
UpdateG(S, τ): It allows the adversary to update some information about the group at time τ, which requires that each element in S is legitimate user's public key at time τ.
$\mathbf{IsActive}(\mathbf{info}_\tau, \mathbf{reg}, i)$: Return 1 if and only if the user i is a legitimate member in the group at time τ, otherwise return 0.

The security analysis

Complexity: Given a security parameter λ, the size of legitimate users t, $l = \lceil \log t \rceil, n = O(\lambda), q = \tilde{O}\left(n^{1.5}\right) = \tilde{O}\left(c\lambda^{1.5}\right)$ with a constant c, $k = O\left(\log\left(\lambda^{1.5}\right)\right)$ (Table 1). Then the size of group public key $gpk = (pp, \mathbf{mpk}, opk)$ is $|gpk| = \tilde{O}\left(\lambda^{1.5}\right) + l \cdot O(\log \lambda)$, the size of signing secret key $gsk_i = (\mathbf{bin}(i), \mathbf{upk}_i, \mathbf{usk}_i)$ is $|gsk_i| = l + 3k = l + O(\log \lambda)$, and the size of signature $\Sigma = (\Pi_{sign}, \mathbf{c}_1, \mathbf{c}_2)$ is

Table 1 Comparison of main parameters in (Ling et al. 2017) and our work

Indicators / Schemes	$\|gpk\|$	$\|gsk[i]\|$	$\|\Sigma\|$
(Ling et al. 2017)	$\tilde{O}\left(\lambda^2+\lambda\cdot l\right)$	$\tilde{O}(\lambda)+l$	$\tilde{O}(\lambda\cdot l)$
Our work	$\tilde{O}(\lambda^{1.5})+l\cdot O(\log\lambda)$	$l+O(\log\lambda)$	$\tilde{O}(\lambda)+l\cdot O\left(\log\lambda^{1.5}\right)$

$$
\begin{aligned}
|\Sigma| =& |\Pi_{sign}|+|\mathbf{c}_1|+|\mathbf{c}_2| \\
=& k'\cdot|CMT|+k'+k'\cdot|RSP|+2(k+1)\log q \\
=& k'\cdot(20kl+6m+12k+3n\log q-5) \\
&+2(k+1)\log q \\
=& k'\cdot(20kl+6m+12k+(3n+2k+2)\log q-5) \\
=& \tilde{O}(\lambda)+l\cdot O\left(\log\lambda^{1.5}\right)
\end{aligned}
$$

Suppose that the upper bounds of the size of the group in (Ling et al. 2017) and that in our work are the same and denoted as N, let $l=\log N$, then the expected computational complexity of realizing the dynamic registration and revocation of the counterpart of the scheme in (Ling et al. 2017) over ring is $O(l)$, and that of our work is

$$
\begin{aligned}
& O\left(\frac{1}{2}\cdot l+\frac{1}{2^2}\cdot(l-1)+\cdots+\frac{1}{2^{l-1}}\cdot 2+\frac{1}{2^l}\right) \\
=& O\left(l\cdot\left(\left(\sum_{i=1}^{l-1}\frac{1}{2^i}\right)+\frac{1}{2^{l-1}}\right)-\sum_{i=2}^{l-1}\frac{i}{2^{i+1}}\right) \\
=& O\left(l-\left(1-\frac{l}{2^{l-1}}\right)\right) \\
=& O(l-1)
\end{aligned}
$$

Correspondingly, the expected space complexity of Merkle tree used in (Ling et al. 2017) is $O(2N-1)$ (Table 2), and that of our work is

$$
\begin{aligned}
& O\left(\frac{1}{2}\cdot(2N-1)+\frac{1}{2^2}\cdot(N-1)+\cdots\right. \\
& \left.+\frac{1}{2^{l-1}}\cdot\left(\frac{N}{2^{l-3}}-1\right)+\frac{1}{2^l}\cdot\left(\frac{N}{2^{l-3}}-1\right)\right) \\
=& O\left(\sum_{i=l}^{3-l}2^i-\sum_{i=1}^{l}\frac{1}{2^i}\right) \\
=& O\left(\frac{1}{3}\cdot\left(2^{l+2}+\frac{1}{2^{l-3}}\right)-\left(1-\frac{1}{2^l}\right)\right) \\
=& O\left(\frac{4}{3}\cdot N-1+\frac{11}{3N}\right) \\
=& O\left(\frac{4}{3}\cdot N-1\right)
\end{aligned}
$$

.

Table 2 Comparison of the expect complexity of Merkle trees used in (Ling et al. 2017) and our work

Indicators / Schemes	The update complexity	The space complexity
(Ling et al. 2017)	$O(\log N)$	$O(2N-1)$
Our work	$O(\log\frac{N}{2})$	$O(\frac{4}{3}\cdot N-1)$

Theorem 4 *The full dynamic group signature scheme based on ring in this paper is correct.*

Proof Now, we give a specific description of the correctness of our scheme according to the perfect completeness of the underlying protocol and the correctness of the encryption scheme. If the signature $\Sigma=(\Pi_{sign},\mathbf{c}_1,\mathbf{c}_2)$ is generated by a legitimate user, then the perfect completeness of the underlying protocol could help the signature Σ to pass the verification of the algorithm **Verify**, and the algorithm **Trace** will outputs the user public key $\mathbf{upk}_i$ with a probability approximate to 1 together with a proof Π_{trace} accepted by **Judge**. We need to compute $\mathbf{e}=\mathbf{c}_{1,2}-\mathbf{S}_1^{\top}c_{1,1}=E_1\cdot\mathbf{r}_1+\left\lfloor\frac{q}{2}\right\rfloor\cdot\mathbf{upk}_i \mod q$ when to decrypt a ciphertext, and let $\mathbf{b}'=\left(b'_1,\cdots,b'_l\right)$, $\mathbf{e}=(e_1,\cdots,e_l)$, for any $j\in[l]$,

$$
b'_j=\begin{cases}0, \text{ if } |e_j-0|<|e_j-\frac{q}{2}| \\ 1, \text{ if } |e_j-0|\geq|e_j-\frac{q}{2}|\end{cases}
$$

Note that $\|E_1\cdot\mathbf{r}_1\|_\infty<\frac{q}{5}$, so $\mathbf{b}'=\mathbf{upk}_i$ with overwhelming probability. Furthermore, because the user corresponding to $\mathbf{upk}_i$ is legitimate, then the witness $w=(\mathbf{bin}(i),\mathbf{w}_l,\cdots,\mathbf{w}_1)$ is included in the group information $\mathbf{info}_\tau$, and the value of the related leaf is not 0^k. So, the algorithm **Trace** could always obtain a tuple $(\mathbf{S}_1,E_1,\mathbf{y})$ that satisfies requirement. And finally, for the fact that the proof Π_{trace} is perfect completeness, so the algorithm **Judge** outputs 1 with probability 1. □

Theorem 5 *Suppose that the problem ring-*$\mathbf{LWE}_{n,m,q,\mathcal{X}}$ *is difficult, then the scheme in this paper is anonymous in RO model.*

Proof Assume that the size of legitimate users is t, the adversary $\mathcal{A}$ and challenger $\mathcal{C}$ are all PPT algorithms. For two different users $i_0\neq i_1\in[t]$ given by $\mathcal{A}$, we give the following game before the concrete proof:

We say that the scheme has a property of anonymity if there is a negligible function $negl(\lambda)$, such that $\Pr\left[\mathbf{Exp}^{anon-b}_{FDGS,\mathcal{A}}(\lambda)\right]=1\leq negl(\lambda)$. Given a negligible function $negl(\lambda)$, we will finish this proof by hybrid games. Let the output of each game is OP_l, $l\in[0,9]$.

Algorithm 2: $\mathbf{Exp}^{anon-b}_{FDGS,\mathcal{A}}(\lambda)$

$(pp,(opk,osk))\leftarrow$
GKeyGen(λ), $HUL,CUL,BUL,SL,CL=\emptyset$.
$(\mathbf{info},(\mathbf{mpk},\mathbf{msk}))\leftarrow\mathcal{A}(pp)$.
Return 0 if $\mathcal{A}$'s output is not well-formed, let
$gpk=(pp,\mathbf{mpk},opk)$.
$b^*\leftarrow\mathcal{A}^{\mathbf{AddU},\mathbf{CreU},\mathbf{RevealU},\mathbf{SenToU},\mathbf{Trace},\mathbf{MReg},\mathbf{Chal_b}}(gpk)$,
return b^*.

Game0: Given two different legitimate users $i_0 \neq i_1 \in [t]$ by $\mathcal{A}$, let $b = 0$, the challenger $\mathcal{C}$ runs the experiment above honestly by using i_0.

Game1: This game is completely consistent with **Game0** except that include $(\mathbf{S}_2, E_2)$ to *osk*, i.e. let $osk = ((\mathbf{S}_1, E_1), (\mathbf{S}_2, E_2))$. And this change, to the view of the adversary $\mathcal{A}$, makes no difference, $\Pr[OP_1 = 1] = \Pr[OP_0 = 1]$.

Game2: This game is completely consistent with **Game1** except that use a simulator Sim_{trace} to simulate the real interactions of the protocol that generates Π_{trace}, i.e. replace the real transcript Π_{trace} with a simulated transcript of Sim_{trace}. And the two transcripts are statistical indistinguishable because of the statistical zero-knowledge of Π_{trace}, $\Pr[OP_2 = 1] - \Pr[OP_1 = 1] \leq negl(\lambda)$.

Game3: This game is completely consistent with **Game2** except that replace $(\mathbf{S}_1, E_1)$ with $(\mathbf{S}_2, E_2)$ when Sim_{trace} simulates the oracle **Trace**. For a legitimate signature $(M, \Pi_{sign}, \mathbf{c}_1, \mathbf{c}_2)$, where $\mathbf{c}_1, \mathbf{c}_2$ are encryptions to different strings respectively. Let F_1 be the signature inquiry initiated by $\mathcal{A}$ to the oracle **Trace**, and the view of $\mathcal{A}$ may changing if F_1 appears, however, it violates the soundness of the protocol that generates Π_{sign}. And the change in this game, to the view of $\mathcal{A}$, is indistinguishable except the incident F_1, i.e. $\Pr[OP_3 = 1] - \Pr[OP_2 = 1] \leq \Pr[F_1] \leq negl(\lambda)$.

Game4: This game is completely consistent with **Game3** except that use a simulator Sim_{sign} to simulate the real interactions of the protocol that generates Π_{sign}, i.e. replace the real transcript Π_{sign} with a simulated transcript of Sim_{sign}. And the two transcripts are statistical indistinguishable because of the statistical zero-knowledge of Π_{sign}, $\Pr[OP_4 = 1] - \Pr[OP_3 = 1] \leq negl(\lambda)$.

Game5: This game is completely consistent with **Game4** except that change the ciphertext $\mathbf{c}_1$ into the encryption to $\mathbf{upk}_{i_1}$ when initiate an inquiry to the oracle $\mathbf{Chal}_b$. And the difference of the view of $\mathcal{A}$ caused by this change is negligible for the semantic security of the encryption scheme. The challenger responds with $(\mathbf{S}_2, E_2)$ during the inquiry to the oracle **Trace**, which makes no difference by substitute the ciphertext $\mathbf{c}_1$, so, $\Pr[OP_5 = 1] - \Pr[OP_4 = 1] = negl(\lambda)$.

Game6: This game is completely consistent with **Game5** except that replace $(\mathbf{S}_2, E_2)$ with $(\mathbf{S}_1, E_1)$ when Sim_{trace} simulates the oracle **Trace**. For a legitimate signature $(M, \Pi_{sign}, \mathbf{c}_1, \mathbf{c}_2)$, where $\mathbf{c}_1, \mathbf{c}_2$ are encryptions to different strings respectively, let F_2 be the signature inquiry initiated by $\mathcal{A}$ to the oracle **Trace**, which violates the soundness of the protocol that generates Π_{sign}. And the change in this game, to the view of $\mathcal{A}$, is indistinguishable except the incident F_2, $\Pr[OP_6 = 1] - \Pr[OP_5 = 1] \leq \Pr[F_2] \leq negl(\lambda)$.

Game7: This game is completely consistent with **Game6** except that change the ciphertext $\mathbf{c}_2$ into the encryption to $\mathbf{upk}_{i_1}$. And the difference of the view of $\mathcal{A}$ caused by this change is negligible for the semantic security of the encryption scheme. The challenger responds with $(\mathbf{S}_1, E_1)$ during the inquiry to the oracle **Trace**, which makes no difference to the view of the adversary, $\Pr[OP_7 = 1] - \Pr[OP_6 = 1] = negl(\lambda)$.

Game8: This game is completely consistent with **Game7** except that replace the simulator Sim_{sign} with a real protocol that generates Π_{sign}, i.e. replace the simulated transcript of Sim_{sign} by a real transcript Π_{sign}. And the two transcripts are statistical indistinguishable because of the statistical zero knowledge of the protocol Π_{sign}, $\Pr[OP_8 = 1] - \Pr[OP_7 = 1] \leq negl(\lambda)$.

Game9: This game is completely consistent with **Game8** except that replace the simulator Sim_{trace} with a real protocol that generates Π_{trace}, i.e. replace the simulated transcript of Sim_{trace} by a real transcript Π_{trace}. And the two transcripts are statistical indistinguishable because of the statistical zero knowledge of the protocol Π_{trace}, $\Pr[OP_9 = 1] - \Pr[OP_8 = 1] \leq negl(\lambda)$.

Finally, we could learn from the games above that the probability:

$$\begin{aligned}&\Pr[OP_9 = 1] - \Pr[OP_0 = 1]\\ =&\Pr\left[\mathbf{Exp}^{anon-1}_{FDGS,\mathcal{A}}(\lambda)\right] - \Pr\left[\mathbf{Exp}^{anon-0}_{FDGS,\mathcal{A}}(\lambda)\right]\\ \leq& c \cdot negl(\lambda)\end{aligned}$$

where c is constant. So, the scheme in this paper satisfies the property of anonymity. □

Theorem 6 *Suppose that the ring-*$\mathbf{SIS}^{\infty}_{n,m,q,1}$ *is difficult, then the scheme in this paper is unforgeable in the RO model.*

Proof Suppose that there ia a PPT adversary $\mathcal{A}$ could forge a valid signature with a non-negligible probability ϵ, then there is a PPT algorithm $\mathcal{B}$ could break the security of Merkle hash tree or solve the problem ring-$\mathbf{SIS}^{\infty}_{n,m,q,1}$ with a non-negligible probability by invoking $\mathcal{A}$ as a black box. And to complete the proof, we give the following game:

If there is a negligible function $negl(\lambda)$, such that $\Pr\left[\mathbf{Exp}^{unforge}_{FDGS,\mathcal{A}}(\lambda)\right] = 1 \leq negl(\lambda)$, then we say that the scheme is unforgeable. Given a random matrix $\mathbf{A}$, the challenger computes the public parameter pp honestly, then invokes the algorithm of $\mathcal{A}$, runs the operations in the game above, during this process, $\mathcal{B}$ responds the inquiries of $\mathcal{A}$ honestly. If the adversary $\mathcal{A}$ wins the game and outputs $\left(M^*, \Sigma^*, i^*, \Pi^*_{trace}, \mathbf{info}_\tau\right)$

Algorithm 3: $\mathbf{Exp}^{unforge}_{FDGS,\mathcal{A}}(\lambda)$ firstly

$pp \leftarrow \mathbf{GKeyGen}(\lambda)$, $HUL, CUL, BUL, SL = \emptyset$.
$(\mathbf{info}, (\mathbf{mpk}, \mathbf{msk}), (opk, osk)) \leftarrow \mathcal{A}(pp)$.
Return 0 if $\mathcal{A}$'s output is not well-formed, let $gpk = (pp, \mathbf{mpk}, opk)$.
$(M, \Sigma, i, \Pi_{trace}, \mathbf{info}_\tau) \leftarrow \mathcal{A}^{\mathbf{CreU},\mathbf{RevealU},\mathbf{SenToU},\mathbf{Sign}}(gpk)$.
Return 1 if $\mathbf{Verify}(gpk, \mathbf{info}_\tau, M, \Sigma) = 1 \wedge \mathbf{Judge}(gpk, \mathbf{upk}_i, \mathbf{info}_\tau, \Pi_{trace}, M, \Sigma) = 1 \wedge i \in HUL \setminus BUL \wedge (M, \Sigma, \tau) \notin SL$.

finally, then there is a non-negligible function ϵ, such that $\Pr\left[\mathbf{Exp}^{unforge}_{FDGS,\mathcal{A}}(\lambda)\right] = 1 \geq \epsilon$, and the algorithm $\mathcal{B}$ could operate as follows: Decompose the signature Σ^* into $\left(\Pi^*_{sign}, \mathbf{c}^*_1, \mathbf{c}^*_2\right)$, where $\Pi_{sign} = \left(\left\{CMT^*_i\right\}_{i=1}^{k'}, CH^*, \left\{RSP^*_i\right\}_{i=1}^{k'}\right)$, because the adversary $\mathcal{A}$ wins the game above, so $\left\{RSP^*_i\right\}_{i=1}^{k'}$ is legitimate responds to $\left\{CMT^*_i\right\}_{i=1}^{k'}, CH^*$. Let $\xi^* = \left(M^*, \left\{CMT^*_i\right\}_{i=1}^{k'}, \mathbf{A}, \mathbf{u}_\tau, \mathbf{B}, P_1, P_2, \mathbf{c}^*_1, \mathbf{c}^*_2\right)$, for the successful probability to guess the value $H(\xi^*)$ is $3^{-k'}$, so the adversary uses the ξ^* to initiate queries to the oracle H with overwhelming probability, and ξ^* is the preimage of H with probability $\epsilon' = \epsilon - 3^{-k'}$, let $t^* \in \{1, 2, \cdots, Q_H\}$ be the index of one inquiry, where Q_H is the number of inquiries that the adversary $\mathcal{A}$ made to the oracle H. The inputs of the hash queries from 1th to t^*th are all ξ^*, and $\mathcal{B}$ runs the operations of $\mathcal{A}$ for t^* times. And the inputs of other hash queries from $t^* + 1$th to Q_Hth are something else, $\mathcal{B}$ responds by independent values respectively. By the Forking lemma in (Brickell et al. 2000; Pointcheval and Stern 2000), the probability of $\mathcal{B}$ gets three different hash values $CH^1_{t^*}, CH^2_{t^*}, CH^3_{t^*} \in \{1, 2, 3\}^{k'}$ to the same input ξ^* is $\geq \frac{1}{2}$, then for any $j \in \{1, 2, \cdots, k'\}$, we have $\Pr\left[\left(CH^1_{t^*,j}, CH^2_{t^*,j}, CH^3_{t^*,j}\right) = (1, 2, 3)\right] = 1 - \left(\frac{7}{9}\right)^{k'}$. Given three different legitimate responds $\left(RSP^1_{t^*,j}, RSP^2_{t^*,j}, RSP^3_{t^*,j}\right)$, what we could learn from the protocol that generates Π_{sign} is that we could extract a witness $\zeta' = \left(\mathbf{usk}_{i'}, \mathbf{upk}_{i'}, w'_\tau, \mathbf{r}'_1, \mathbf{r}'_2\right)$, where $w'_\tau = (\mathbf{bin}(i'), \mathbf{w}'_{l,\tau}, \cdots, \mathbf{w}'_{1,\tau}) \in \{0,1\}^l \times \left(\{0,1\}^k\right)^l$, such that for $\forall b \in \{1, 2\}, \forall j \in \{0, l-1\}$, we have

$$\begin{cases} \mathbf{u}_{j,\tau} = \begin{cases} h_{\mathbf{A}}(\mathbf{u}_{j+1,\tau}, \mathbf{w}_{j+1,\tau}), \ if \ i'_{j+1} = 0 \\ h_{\mathbf{A}}(\mathbf{w}_{j+1,\tau}, \mathbf{u}_{j+1,\tau}), \ if \ i'_{j+1} = 1 \end{cases} \\ \mathbf{A} \cdot \mathbf{usk}_{i'} = \mathbf{G} \cdot \mathbf{upk}_{i'} \\ \mathbf{c}^*_b = \left(c^*_{b,1}, \mathbf{c}^*_{b,2}\right) = \left(\mathbf{B} \cdot \mathbf{r}'_b, P_b \cdot \mathbf{r}'_b + \left\lfloor \frac{q}{2} \right\rceil \cdot \mathbf{upk}_{i'}\right) \end{cases}$$

We can learn from the correctness of the encryption scheme that $\mathbf{c}^*_1$ is the encryption to $\mathbf{upk}_{i'}$. The algorithm **Judge** outputs 1 because of the fact that $\mathcal{A}$ wins the game, and what we can learn from the soundness of the protocol that generates Π_{trace} is that $\mathbf{c}^*_1$ is the encryption to $\mathbf{upk}_{i^*}$, then $\mathbf{upk}_{i'} = \mathbf{upk}_{i^*}$ with overwhelming probability. By the correctness of the Merkle hash tree, the user i^* is legitimate. $i^* \in HUL \setminus BUL$ indicates that the adversary $\mathcal{A}$ doesn't know $gsk_{i^*} = (\mathbf{bin}(i^*), \mathbf{upk}_{i'}, \mathbf{usk}_{i^*})$. $\mathbf{usk}_{i^*}$ was chosen by $\mathcal{B}$ and $\mathbf{A} \cdot \mathbf{usk}_{i^*} = \mathbf{G} \cdot \mathbf{upk}_{i'}$, so we have $\Pr[\mathbf{usk}_{i^*} \neq \mathbf{usk}_{i'}] \geq \frac{1}{2}$. Let $\mathbf{z} = \mathbf{usk}_{i^*} - \mathbf{usk}_{i'}$, then $\mathbf{z} \neq \mathbf{0}$ and $\mathbf{Az} = 0 \mod q$, so, the algorithm $\mathcal{B}$ could solve the problem ring-$\mathbf{SIS}^{\infty}_{n,m,q,1}$ with non-negligible probability. □

Theorem 7 *Suppose that the ring-*$\mathbf{SIS}^{\infty}_{n,m,q,1}$ *is difficult, then the scheme in this paper is traceable in RO model.*

Proof To finish the proof, we give the following game firstly:

If there is a negligible function $negl(\lambda)$, such that $\Pr\left[\mathbf{Exp}^{trace}_{FDGS,\mathcal{A}}(\lambda)\right] = 1 \leq negl(\lambda)$, then we say that the scheme is traceable. In other words, If the adversary $\mathcal{A}$ wins the game above, the signature generated by $\mathcal{A}$ is legitimate and it was traced to a revoked user or a legitimate user without a valid proof Π_{trace} to it, and next, we will explain that the probability of the fact that the adversary $\mathcal{A}$ wins the game above is negligible.

Let $(\mathbf{info}_\tau, M, \Sigma)$ be a forged information by the adversary $\mathcal{A}$ in the game $\mathbf{Exp}^{trace}_{FDGS,\mathcal{A}}(\lambda)$, then the challenger could extract the identity $(\mathbf{bin}(i), \Pi_{trace})$ by running the algorithm **Trace**. Decompose the signature Σ into $(\Pi_{sign}, \mathbf{c}'_1, \mathbf{c}'_2)$, where $\Pi_{sign} = \left(\{CMT_j\}_{j=1}^{k'}, CH, \{RSP_j\}_{j=1}^{k'}\right)$, for $(\mathbf{info}_\tau, M, \Sigma)$ is a legitimate signature, so $\{RSP_j\}_{j=1}^{k'}$ are valid responds to $\{CMT_j\}_{j=1}^{k'}, CH$. Then we could extract a witness $\zeta' = \left(\mathbf{usk}_{i'}, \mathbf{upk}_{i'}, w'_\tau, \mathbf{r}'_1, \mathbf{r}'_2\right)$, which is similar to

Algorithm 4: $\mathbf{Exp}^{trace}_{FDGS,\mathcal{A}}(\lambda)$

$(pp, (\mathbf{mpk}, \mathbf{msk})) \leftarrow \mathbf{GKeyGen}(\lambda)$,
$HUL, CUL, BUL, SL = \emptyset$.
$(\mathbf{info}, (opk, osk)) \leftarrow \mathcal{A}(pp)$.
Return 0 if $\mathcal{A}$'s output is not well-formed, let $gpk = (pp, \mathbf{mpk}, opk)$.
$(M, \Sigma, \tau) \leftarrow \mathcal{A}^{\mathbf{AddU},\mathbf{CreU},\mathbf{SenToM},\mathbf{RevealU},\mathbf{MReg},\mathbf{Sign},\mathbf{UpdateG}}(gpk)$.
Return 0 if $\mathbf{Verify}(gpk, \mathbf{info}_\tau, M, \Sigma) = 0$.
$(i, \Pi_{trace}) \leftarrow \mathbf{Trace}(gpk, osk, \mathbf{info}_\tau, \mathbf{reg}, M, \Sigma)$.
Return 1 if $\mathbf{IsActive}(\mathbf{info}_\tau, \mathbf{reg}, i) = \perp \vee \mathbf{Judge}(gpk, \mathbf{upk}_i, \mathbf{info}_\tau, \Pi_{trace}, M, \Sigma) = 0 \vee i = 0$.

the property of unforgeability, where $w'_\tau = \left(\mathbf{bin}(i'), \mathbf{w}'_{l,\tau}, \cdots, \mathbf{w}'_{1,\tau}\right) \in \{0,1\}^l \times (\{0,1\}^k)^l$, such that for $\forall b \in \{1,2\}, \forall j \in \{0, l-1\}$, we have

$$\begin{cases} \mathbf{upk}_{i'} \neq 0 \\ \mathbf{u}_{j,\tau} = \begin{cases} h_{\mathbf{A}}(\mathbf{u}_{j+1,\tau}, \mathbf{w}_{j+1,\tau}), \ if \ i'_{i+1} = 0 \\ h_{\mathbf{A}}(\mathbf{w}_{j+1,\tau}, \mathbf{u}_{j+1,\tau}), \ if \ i'_{i+1} = 1 \end{cases} \\ \mathbf{A} \cdot \mathbf{usk}_{i'} = \mathbf{G} \cdot \mathbf{upk}_{i'} \\ \mathbf{c}'_b = \left(c'_{b,1}, \mathbf{c}'_{b,2}\right) = \left(\mathbf{B} \cdot \mathbf{r}'_b, P_b \cdot \mathbf{r}'_b + \lfloor \frac{q}{2} \rceil \cdot \mathbf{upk}_{i'}\right) \end{cases}$$

What we can learn from the correctness of the encryption scheme is that the ciphertext $\mathbf{c}'_1$ could be decrypted to $\mathbf{upk}_{i'}$, and we can learn from the correctness of the algorithm **Trace** that $\mathbf{upk}_i$ is the plaintext obtained from the ciphertext $\mathbf{c}'_1$, so $\mathbf{upk}_i = \mathbf{upk}_{i'}$ with overwhelming probability, and the probability that a valid signature be traced to a revoked user is negligible. In fact, we can learn from the security of Merkle hash tree that the probability that the valid signature above be traced to a revoked user with a valid proof Π_{trace} is negligible. Because of the fact that the challenger has the legitimate witness to generate a valid proof Π_{trace}, and we can learn from the perfect completeness of the protocol that generates Π_{trace} that the algorithm **Judge** would accepts Π_{trace} with probability 1. In conclusion, the scheme in this paper is traceable. □

Theorem 8 *The scheme in this paper satisfies the property of tracing soundness in RO model.*

Proof To finish the proof, we give the following game firstly:

Algorithm 5: $\mathbf{Exp}^{trace-sound}_{FDGS,\mathcal{A}}(\lambda)$

$pp \leftarrow \mathbf{GKeyGen}(\lambda), CUL = \emptyset.$
$(\mathbf{info}, (\mathbf{mpk}, \mathbf{msk}), (opk, osk)) \leftarrow \mathcal{A}(pp).$
Return 0 if $\mathcal{A}$'s output is not well-formed, let
$gpk = (pp, \mathbf{mpk}, opk).$
$(M, \Sigma, i_0, \Pi_{trace,i_0}, i_1, \Pi_{trace,i_1}, \mathbf{info}_\tau) \leftarrow$
$\mathcal{A}^{\mathbf{CreU},\mathbf{MReg}}(gpk).$
Return 1 if for
$b \in \{0,1\}, \mathbf{Verify}(gpk, \mathbf{info}_\tau, M, \Sigma) = 1 \wedge i_0 \neq i_1 \neq \perp$
$\wedge \mathbf{Judge}(gpk, \mathbf{upk}_{i_b}, \mathbf{info}_\tau, \Pi_{trace}, M, \Sigma) = 1.$

Suppose that the information $(M, \Sigma, i_0, \Pi_{trace,i_0}, i_1, \Pi_{trace,i_1}, \mathbf{info}_\tau)$ is the output of the adversary $\mathcal{A}$ in this game, if the game $\mathbf{Exp}^{trace-sound}_{FDGS,\mathcal{A}}(\lambda)$ outputs 1 finally, i.e. $\mathbf{Judge}(gpk, \mathbf{upk}_{i_b}, \mathbf{info}_\tau, \Pi_{trace}, M, \Sigma) = 1$, $i_0 \neq i_1 \neq \perp$, $\mathbf{Verify}(gpk, \mathbf{info}_\tau, M, \Sigma) = 1$, then we say that $\mathcal{A}$ wins. Given a transcript $\Pi_{trace} = \left(\{CMT_j\}_{j=1}^{k'}, CH, \{RSP_j\}_{j=1}^{k'}\right)$, the fact that the algorithm **Judge** outputs 1 indicates that $\{RSP_j\}_{j=1}^{k'}$ are legitimate responds to $\{CMT_j\}_{j=1}^{k'}, CH$. For any $b \in \{0,1\}$, it is similarly to the property of unforgeability, we could extract $\mathbf{S}_{1,b}, E_{1,b}, \mathbf{y}_b$, such that

$$\|\mathbf{S}_{1,b}\|_\infty \leq \beta, |E_{1,b}| \leq \beta, \|\mathbf{y}_b\|_\infty \leq \lceil \frac{q}{5} \rceil$$
$$\mathbf{S}^\top_{1,b} \cdot \mathbf{B} + E_{1,b} = P_{1,b} \mod q$$
$$\mathbf{c}_{1,2} - \mathbf{S}^\top_{1,b} \cdot c_{1,1} = \mathbf{y}_b + \lfloor \frac{q}{2} \rfloor \cdot \mathbf{upk}_{i_b} \mod q$$

then we have

$$\left(\mathbf{S}^\top_{1,0} - \mathbf{S}^\top_{1,1}\right) \cdot c_{1,1} = (\mathbf{y}_1 - \mathbf{y}_0) + \lfloor \frac{q}{2} \rfloor \cdot (\mathbf{upk}_{i_1} - \mathbf{upk}_{i_0}) \mod q$$

Suppose that $\mathbf{upk}_{i_1} \neq \mathbf{upk}_{i_0}$, so $\| \lfloor \frac{q}{2} \rfloor \cdot (\mathbf{upk}_{i_1} - \mathbf{upk}_{i_0}) \|_\infty = \lfloor \frac{q}{2} \rfloor$, $\|\mathbf{y}_1 - \mathbf{y}_0\|_\infty \leq 2 \cdot \lceil \frac{q}{5} \rceil$, and

$$\|(\mathbf{y}_1 - \mathbf{y}_0) + \lfloor \frac{q}{2} \rfloor \cdot (\mathbf{upk}_{i_1} - \mathbf{upk}_{i_0})\|_\infty > 0$$

then $\mathbf{S}^\top_{1,0} \neq \mathbf{S}^\top_{1,1}$, we obtained two different solutions of the function $\mathbf{S}^\top_1 \cdot \mathbf{B} + E_1 = P_1 \mod q$, which is contradictory to the fact that there is at most one solution to the ring-$\mathbf{LWE}_{n,m,q,\mathcal{X}}$ sample $(\mathbf{B}, P_1)$. So, $\mathbf{upk}_{i_1} = \mathbf{upk}_{i_0}$ with overwhelming probability. In other words, the probability of the fact that $\mathcal{A}$ wins is negligible, so the scheme in this paper satisfies the property of tracing soundness. □

Conclusion

In this paper, we give the first ring based full dynamic group signature scheme, and improve the efficiency of it mainly from the following three aspects: the size of public/secret keys, the dynamic construction of the Merkle hash tree, and the reuse of its leaves. These changes help to reduce the computational complexity and space complexity by leaps and bounds. In addition, we avoid the adverse condition where the group managers generate their keys maliciously. Though we have tried a lot, there is still a large space for improvement in the use of zero-knowledge proof, and the problem of the delayed verification of a signature is also not solved. Next, we would like to focus on the two problems and do some correlative works.

Acknowledgements
Not applicable.

Authors' contributions
The first author conceived the idea of the study and wrote the paper; all authors discussed the results and revised the final manuscript. All authors read and approved the final manuscript.

Funding
This work was supported by National Natural Science Foundation of China (Grant No. 61379141 and No. 61772521), Key Research Program of Frontier Sciences, CAS (Grant No. QYZDB-SSW-SYS035), and the Open Project Program of the State Key Laboratory of Cryptology.

Availability of data and materials
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 20 March 2019 Accepted: 27 May 2019
Published online: 17 July 2019

References

An Efficient Protocol for Anonymously Providing Assurance of the Container of a Private Key. https://www.researchgate.net/publication/243775241. Accessed 2004

Ateniese G, Camenisch J, Joye M, Tsudik G (2000) A practical and provably secure coalition-resistant group signature scheme. In: Bellare M (ed). Proceedings of Conference CRYPTO: 20-24 August 2000; California. Springer, Beilin Heidelberg. pp 255–270

Bellare M, Shi HX, Zhang C (2005) Foundations of group signatures: the case of dynamic groups. In: Menezes A (ed). Proceedings of Conference CT-RSA: 14-18 February 2005; San Francisco. Springer, Beilin Heidelberg. pp 136–153

Bichsel P, Camenisch J, Neven G, Smart NP, Warinschi B (2010) Get shorty via group signatures without encryption. In: Garay JA, Prisco RD (eds). Proceedings of Conference SCN: 13-15 September 2010; Amalfi. Springer, Beilin Heidelberg. pp 381–398

Boneh D, Boyen X, Shacham H (2004) Short group signatures. In: Franklin M (ed). Proceedings of Conference CRYPTO: 15-19 August 2004; California. Springer, Beilin Heidelberg. pp 41–55

Boneh D, Shacham H (2004) Group signatures with verifier-local revocation. In: Proceedings of Conference CCS: 25-29 October 2004; Washington DC. ACM DL. pp 168–177

Bootle J, Cerulli A, Chaidos P, Ghadafi E, Groth J (2016) Foundations of fully dynamic group signatures. In: Manulis M, Sadeghi AR, Schneider S (eds). Proceedings of Conference ACNS: 19-22 June 2016; Guildford. Springer, Beilin Heidelberg. pp 117–136

Boyen X, Waters B (2006) Compact group signatures without random oracles. In: Vaudenay S (ed). Proceedings of Conference EUROCRYPT: 28 May-1 June 2006; St.Petersburg. Springer, Beilin Heidelberg. pp 427–444

Boyen X, Waters B (2007) Full-domain subgroup hiding and constant-size group signatures. In: Okamoto T, Wang XT (eds). Proceedings of Conference PKC: 16-20 April 2007; Beijing. Springer, Beilin Heidelberg. pp 1–15

Bresson E, Stern J (2001) Efficient revocation in group signatures. In: Kim K (ed). Proceedings of Conference PKC: 13-15 February 2001; Cheju Island. Springer, Beilin Heidelberg. pp 190–206

Brickell E, Pointcheval D, Vaudenay S, Yung M (2000) Design validations for discrete logarithm based signature schemes. In: Imai H, Zheng YL (eds). Proceedings of Conference PKC: 18-20 January 2000; Melbourne. Springer, Beilin Heidelberg. pp 276–292

Camenisch J, Groth J (2004) Group signatures: better efficiency and new theoretical aspects. In: Blundo C, Cimato S (eds). Proceedings of Conference SCN: 8-10 September 2004; Amalfi. Springer, Beilin Heidelberg. pp 120–133

Camenisch J, Lysyanskaya A (2002) Dynamic accumulators and application to efficient revocation of anonymous credentials. In: Yung M (ed). Proceedings of Conference CRYPTO: 18-22 August 2002; California. Springer, Beilin Heidelberg. pp 61–76

Camenisch J, Lysyanskaya A (2004) Signature schemes and anonymous credentials from bilinear maps. In: Franklin M (ed). Proceedings of Conference CRYPTO: 15-19 August 2004; California. Springer, Beilin Heidelberg. pp 56–72

Camenisch J, Michels M (1998) A group signature scheme with improved efficiency. In: Ohta K, Pei DY (eds). Proceedings of Conference ASIACRYPT: 18-22 October 1998; Beijing. Springer, Beilin Heidelberg. pp 160–174

Camenisch J, Stadler M (1997) Efficient group signature schemes for large groups (extended abstract). In: KaliskiJr BS (ed). Proceedings of Conference CRYPTO: 17-21 August 1997; California. Springer, Beilin Heidelberg. pp 410–424

Camenisch J, Neven G, Rückert M (2012) Fully anonymous attribute tokens from lattices. In: Visconti I, Prisco RD (eds). Proceedings of Conference SCN: 5-7 September 2012; Amalfi. Springer, Beilin Heidelberg. pp 57–75

Chaum D, van Heyst EV (1991) Group signatures. In: Davies DW (ed). Proceedings of Conference EUROCRYPT: 8-11 April 1991; Brighton. Springer, Beilin Heidelberg. pp 257–265

Chen L, Pedersen TP (1994) New group signature schemes. In: Santis AD (ed). Proceedings of Conference EUROCRYPT: 9-12 May 1994; Perugia. Springer, Beilin Heidelberg. pp 171–181

Delerablée C, Pointcheval D (2006) Dynamic fully anonymous short group signatures. In: Nguyen PQ (ed). Proceedings of Conference VIETCRYPT: 25-28 September 2006; Hanoi. Springer, Beilin Heidelberg. pp 193–210

Dodis Y, Kiayias A, Nicolosi A, Shoup V (2004) Anonymous identification in ad hoc groups. In: Cachin C, Camenisch JL (eds). Proceedings of Conference EUROCRYPT: 2-6 May 2004; Interlaken. Springer, Beilin Heidelberg. pp 609–626

Furukawa J, Imai H (2005) An efficient group signature scheme from bilinear maps. IEICE Trans Fundam Electron Commun Comput Sci E89-A:1328–1338

Furukawa J, Yonezawa S (2004) Group signatures with separate and distributed authorities. In: Blundo C, Cimato S (eds). Proceedings of Conference SCN: 8-10 September 2004; Amalfi. Springer, Beilin Heidelberg. pp 77–90

Gentry C, Peikert C, Vaikuntanathan V (2008) Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of Conference STOC: 17-20 May 2008; Victoria. ACM DL. pp 197–206

Gordon SD, Katz J, Vaikuntanathan V (2010) A group signature scheme from lattice assumptions. In: Abe M (ed). Proceedings of Conference ASIACRYPT: 5-9 December 2010; Singapore. Springer, Beilin Heidelberg. pp 395–412

Groth J (2006) Simulation-sound nizk proofs for a practical language and constant size group signatures. In: Lai XJ, Chen KF (eds). Proceedings of Conference ASIACRYPT: 3-7 December 2006; Shanghai. Springer, Beilin Heidelberg. pp 444–459

Groth J (2007) Fully anonymous group signatures without random oracles. In: Kurosawa K (ed). Proceedings of Conference ASIACRYPT: 2-6 December 2007; Kuching. Springer, Beilin Heidelberg. pp 164–180

Kawachi A, Tanaka K, Xagawa K (2008) Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In: Pieprzyk J (ed). Proceedings of Conference ASIACRYPT: 7-11 December 2008; Singapore. Springer, Beilin Heidelberg. pp 372–389

Kiayias A, Yung M (2006) Secure scalable group signature with dynamic joins and separable authorities. Secur Netw 1:24–45

Laguillaumie F, Langlois A, Libert B, Stehlé D (2013) Lattice-based group signatures with logarithmic signature size. In: Sako K, Sarkar P (eds). Proceedings of Conference ASIACRYPT: 1-5 December 2013; Bengaluru. Springer, Beilin Heidelberg. pp 41–61

Langlois A, Ling S, Nguyen K, Wang H (2014) Lattice-based group signature scheme with verifier-local revocation. In: Krawczyk H (ed). Proceedings of Conference PKC: 26-28 March 2014; Buenos. Springer, Beilin Heidelberg. pp 345–361

Libert B, Ling S, Mouhartem F, Nguyen K, Wang H (2016a) Signature schemes with efficient protocols and dynamic group signatures from lattice assumptions. In: Cheon JH, Takagi T (eds). Proceedings of Conference ASIACRYPT: 4-8 December 2016; Hanoi. Springer, Beilin Heidelberg. pp 373–403

Libert B, Ling S, Nguyen K, Wang H (2016b) Zero-knowledge arguments for lattice-based accumulators: logarithmic-size ring signatures and group signatures without trapdoors. In: Fischlin M, Coron JS (eds). Proceedings of Conference EUROCRYPT: 8-12 May 2016; Vienna. Springer, Beilin Heidelberg. pp 1–31

Libert B, Peters T, Yung M (2012a) Group signatures with almost-for-free revocation. In: Naini RS, Canetti R (eds). Proceedings of Conference CRYPTO: 19-23 August 2012; Santa Barbara. Springer, Beilin Heidelberg. pp 571–589

Libert B, Peters T, Yung M (2012b) Scalable group signatures with revocation. In: Pointcheval D, Johansson T (eds). Proceedings of Conference EUROCRYPT: 15-19 April 2012; Cambridge. Springer, Beilin Heidelberg. pp 609–627

Libert B, Vergnaud D (2009) Group signatures with verifier-local revocation and backward unlinkability in the standard model. In: Garay JA, Miyaji A, Otsuka A (eds). Proceedings of Conference CANS: 12-14 December 2009; Kanazawa. Springer, Beilin Heidelberg. pp 498–517

Ling S, Nguyen K, Wang HX (2015) Group signatures from lattices: simpler, tighter, shorter, ring-based. In: Katz J (ed). Proceedings of Conference PKC: 30 March-1 April 2015; Gaithersburg. Springer, Beilin Heidelberg. pp 427–449

Ling S, Nguyen K, Wang H, Xu Y (2017) Lattice-based group signatures: achieving full dynamicity with ease. In: Gollmann D, Miyaji A, Kikuchi H (eds). Proceedings of Conference ACNS: 10-12 July 2017; Kanazawa. Springer, Beilin Heidelberg. pp 293–312

Lyubashevsky V (2008) Lattice-based identification schemes secure under active attacks. In: Cramer R (ed). Proceedings of Conference PKC: 9-12 March 2008; Barcelona. Springer, Beilin Heidelberg. pp 162–179

Lyubashevsky V (2012) Lattice signatures without trapdoors. In: Pointcheval D, Johansson T (eds). Proceedings of Conference EUROCRYPT: 15-19 April 2012; Cambridge. Springer, Beilin Heidelberg. pp 738–755
Lyubashevsky V, Micciancio D (2006) Generalized compact knapsacks are collision resistant. In: Bugliesi M, Preneel B, Sassone V, Wegener I (eds). Proceedings of Conference ICALP: 10-14 July 2006; Venice. Springer, Beilin Heidelberg. pp 144–155
Lyubashevsky V, Peikert C, Regev O (2010) On ideal lattices and learning with errors over rings. In: Gilbert H (ed). Proceedings of Conference EUROCRYPT: 30 May-3 June 2010; Riviera. Springer, Beilin Heidelberg. pp 1–23
Lyubashevsky V, Peikert C, Regev O (2013) A toolkit for ring-lwe cryptography. In: Johansson T, Nguyen PQ (eds). Proceedings of Conference EUROCRYPT: 26-30 May 2013; Athens. Springer, Beilin Heidelberg. pp 35–54
Nakanishi T, Funabiki N (2005) Verifier-local revocation group signature schemes with backward unlinkability from bilinear maps. In: Roy B (ed). Proceedings of Conference ASIACRYPT: 4-8 December 2005; Chennai. Springer, Beilin Heidelberg. pp 533–548
Nakanishi T, Fujii H, Hira Y, Funabiki N (2009) Revocable group signature schemes with constant costs for signing and verifying. In: Jarecki S, Tsudik G (eds). Proceedings of Conference PKC: 18-20 March 2009; Irvine. Springer, Beilin Heidelberg. pp 463–480
Naor D, Naor M, Lotspiech J (2001) Revocation and tracing schemes for stateless receivers. In: Kilian J (ed). Proceedings of Conference CRYPTO: 19-23 August 2001; Santa Barbara. Springer, Beilin Heidelberg. pp 41–62
Naor M, Yung M (1990) Public-key cryptosystems provably secure against chosen ciphertext attacks. In: Proceedings of Conference STOC: 1990; Baltimore. ACM DL. pp 427–437
Nguyen L (2005) Accumulators from bilinear pairings and applications. In: Menezes A (ed). Proceedings of Conference CT-RSA: 14-18 February 2005; San Francisco. Springer, Beilin Heidelberg. pp 275–292
Nguyen L, Naini RS (2004) Efficient and provably secure trapdoor-free group signature schemes from bilinear pairings. In: Lee PJ (ed). Proceedings of Conference ASIACRYPT: 5-9 December 2004; Jeju Island. Springer, Beilin Heidelberg. pp 372–386
Nguyen PQ, Zhang J, Zhang Z (2015) Simpler efficient group signatures from lattices. In: Katz J (ed). Proceedings of Conference PKC: 30 March-1 April 2015; Gaithersburg. Springer, Beilin Heidelberg. pp 401–426
Peikert C (2016) A decade of lattice cryptography. Found Trends Theor Comput Sci 10:283–424
Peikert C, Rosen A (2006) Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In: Halevi S, Rabin T (eds). Proceedings of Conference TCC: 4-7 March 2006; New York. Springer, Beilin Heidelberg. pp 145–166
Peikert C, Rosen A (2007) Lattices that admit logarithmic worst-case to average-case connection factors. In: Proceedings of Conference STOC: 11-13 June 2007; San Diego. ACM DL, Beilin Heidelberg. pp 478–487
Pointcheval D, Stern J (2000) Security arguments for digital signatures and blind signatures. Cryptology 13:361–396
Practical Group Signatures Without Random Oracles. http://citeseerx.ist.psu.edu/viewdoc. Accessed 2005
Regev O (2009) On lattices, learning with errors, random linear codes, and cryptography. J ACM 56:1–40
Sakai Y, Schuldt JCN, Emura K, Hanaoka G, Ohta K (2012) On the security of dynamic group signatures: preventing signature hijacking. In: Fischlin M, Buchmann J, Manulis M (eds). Proceedings of Conference PKC: 21-23 May 2012; Darmstadt. Springer, Beilin Heidelberg. pp 715–732
Signing on Elements in Bilinear Groups for Modular Protocol Design. https://eprint.iacr.org/2010/133.pdf. Accessed 2010
Song DX (2001) Practical forward secure group signature schemes. In: Proceedings of Conference CCS: 5-8 November 2001; Philadelphia. ACM DL. pp 225–234
Stern J (1996) A new paradigm for public key identification. IEEE Trans Inf Theory 42:1757–1768

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen journal and benefit from:

- ► Convenient online submission
- ► Rigorous peer review
- ► Open access: articles freely available online
- ► High visibility within the field
- ► Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com