Dynamic Group Signature Scheme on Lattice with Verifier-local Revocation

Xiuju Huang, Jiashuo Song, and Zichen Li

School of Information Engineering,
Beijing Institute of Graphic Communication,
N0.1 Xinghua Street, Daxing District, Beijing, 102600, china
xiujuhuang@163.com

Abstract. The verifier-local revocation mechanism (VLR) is an ideal function of group signature. As long as the verifier knows the revocation list, he/she can verify the legitimacy of the signer, prevent the revoked user from impersonating a legitimate user for signature, ensure the timeliness of signature information and save resources. Group signature is often required to realize users' dynamic addition and revocation. Therefore, an efficient lattice signature scheme with a local revocation mechanism and alter the number of users has become an important topic. In this paper, a zero-knowledge proof scheme on the lattice has been proposed. Based on it, a group signature scheme with VLR has been constructed. This scheme can effectively join and revocation without generating the key pair again. The tracking mechanism uses an encryption scheme. As long as given a correct tracking key, the signer index can be opened quickly. And this algorithm has short public key, logarithmic signature length, and efficient implementation of the VLR function.

Keywords: dynamic group signature · lattice · Zero—knowledge proof · VLR.

1 Introduction

With the development of network, digital signature has become an important research topic and has been widely used in various places such as anonymous voting, electronic bidding, etc. In order to realize the anonymity of signature, it is necessary to ensure that the verifier can only verify the validity of the signature without any information from the signer. However, When the signature has problems, the entire signature scheme is required to be traceable. In order to satisfy the above requirements, Chaum [1] proposed the concept of group signature in 1991 at first. In the group signature scheme, a user signs the message, and any verifier can verifie it when the signer's identity cannot be confirmed. The group administrator GM has a tracking key which can query the actual signer in case of problems. Later, group signature schemes based on classical mathematical difficult problems were proposed and improved [2,3].
With the development of quantum computers, various post-quantum cryptography schemes have been proposed [4,5], and lattice cryptography has become

one of the best. In 2010, Gordon et al. constructed the first group signature scheme on lattice according to the LWE difficulty problem on lattice [6], which constructed a zero-knowledge proof, and realized the opened operation using the distance size. This scheme is a static group signature scheme, which can't realize the function of joining and revocation. The length of the public key and signature has a linear relationship with the number of group members, so it is not suitable for the case of too many group members. In 2013, a new lattice signature scheme was proposed[7], which reduced the signature size and made the signature size logarithmic with the group members. In order to revoke the signature of misbehaving users, the whole system often needs to be reinitialized. All public keys and private keys need to be updated such that illegal users can't know the private keys of legal users to sign, which is cumbersome and inefficient. In order to improve work efficiency, a solution that group signature with verifier local revocation (VLR) was proposed[8] in 2014, . It is an effective solution to realize the traceability and anonymity of signatures by constructing an underlying protocol. In September 2015, a zero-knowledge proof scheme based on LWE and SIS was constructed, which reduced the length of the public key and improved the signature efficiency, such that the signature length is fixed and independent of the number of groups members. The zero-knowledge of the scheme construction has become the basis of many schemes construction [10,11]. In the same year, San Ling et al. proposed an improved group signature scheme by constructing an interactive zero-knowledge proof and combining OTS and GVP-IBE [12]. In recent years, lattice group signature schemes have been proposed and optimized to meet the different needs of users [13,14].

In this paper, a zero-knowledge proof scheme is proposed, a group signature scheme with a VLR attribute is constructed on this basis. The scheme uses zero-knowledge proof to prove the effectiveness of the signature. If the verifier has the revocation list RL, he/she can check the legitimacy of the signer. The traceability of the scheme is realized by using the encryption scheme, and the scheme can realize the mechanism of joining and revocation of legal members through simple operation without frequently updating the key. The public key is short, and the signature length is the logarithm of group members.

2 Preliminaries

NOTATIONS.For a positive integer n, we let $[n]$ denote the set $\{1, \ldots, n\}$, S_k denote the set of all permutations of k elements.B_{3m} denotes the set of all vectors in $\{-1, 0, 1\}^{3m}$, which have the number of - 1, 0, and 1 is m, respectively. B_{2l} is the set of all vectors in $\{0, 1\}^{2l}$,which have the number of 0 and 1 is l, respectively. $(x||y) \in R^{m+k}$ denote the concatenation of vectors $x \in R^m$ and $y \in R^k$. $[A|B] \in R^{n\times(m+k)}$ denote the column concatenation of matrices $A \in R^{n\times m}$ and matrices $A \in R^{n\times m}$.I is the identity matrix.

2.1 Lattice

Definition1: Let $b_1 b_2, ..., b_m \in R^n$ be m linearly independent vectors, and the lattice composed of them as the basis is as follows:

$$\Lambda\left(b_1 b_2, ..., b_m\right) = \left\{ \sum_{i=1}^{m} a_i b_i | a_i \in Z \right\} \tag{1}$$

Definition2: For a given matrix $A \in Z_q^{n \times m}$, vectors $\in Z^n$, define the m-dimensional lattice as follows:

$$L\left(A^T\right) = \left\{ y \in Z^m | y \equiv A^T s \left(mod q\right) \right\} \tag{2}$$

define the orthogonal lattice as:

$$\Lambda^{\perp}\left(A\right) = \left\{ w \in Z^m | A \cdot w = 0 \bmod q \right\} \tag{3}$$

For any u in the image of , define the coset:

$$\Lambda_u^{\perp}\left(A\right) = \left\{ w \in Z^m | A \cdot w = u \bmod q \right\} \tag{4}$$

2.2 Difficult problems on the lattice

Given positive integers$n$, $m \geq n$, $q \geq 2$, the probability distribution χ on interval $[0, q)^m$ and vector $s \in Z_q^n$, we define as follows:
$LWE_{m,q,\chi}\left(s\right)$: For a vector $s \in Z_q^n$ and a distributionχ. Given $e \leftarrow \chi$ and $A \in Z_q^{n \times m}$ which uniformly at random, obtain the pair $\left(A, A^T s + e \left(\bmod q\right)\right)$. The problem is finding a vector $s \in Z_q^n$ by $\left(A, A^T s + e \left(\bmod q\right)\right)$. The output of $\left(A, A^T s + e \left(\bmod q\right)\right)$ is indistinguishable from the (A, y), where $A \in Z_q^{n \times m}, y \in [0, q)^m$ are uniformly random.
$SIS_{\ n,m,q,\beta}\left(s\right)$: Given a matrix $A \in Z_q^{n \times m}$ which uniformly at random, the problem is finding a non-zero vector $s \in \Lambda^{\perp}\left(A\right)$ such that $\left\|s\right\|_{\infty} \leq \beta$ and $A \cdot s = 0 \bmod q$.
$ISIS_{n,m,q,\beta}\left(s\right)$: Given randomly uniformly selected matrix $A \in Z_q^{n \times m}$ and vector $y \in Z_q^n$, the problem is finding a non-zero vector $s \in \Lambda_u^{\perp}\left(A\right)$such that $\left\|s\right\|_{\infty} \leq \beta$ and $A \cdot s = y \bmod q$.

2.3 Sampling function on lattice

For a vector $c \in R^m$, m dimensional continuous Gaussian distribution is defined as follow:

$$D_{s,c}\left(x\right) = {}^{1}\!/_{s^m} \cdot \exp\left(-\pi\left({}^{\left\|x-c\right\|}\!/_{s}\right)^2\right) \tag{5}$$

where c is the center of the distribution. The discrete Gaussian distribution on the lattice is defined as follow:

$$\forall x \in \wedge : D_{\wedge,s,c}\left(x\right) = \frac{D_{s,c}\left(x\right)}{\sum_{y \in \wedge} D_{s,c}\left(y\right)} \tag{6}$$

Lemma1[16]: Let n, $q \geq 2$, and $m \geq 8nlogq$. Matrix $A \in Z_q^{n\times m}$ is statistically close to uniform over $Z_q^{n\times m}$ and its trapdoor $T \in Z^{m\times m}$ is a short basis for the lattice $\Lambda^{\perp}(A)$ such that $A \cdot T = 0\,(modq)$, $\|T\| = O\,(nlogq)$ and $\|T\| \leq C \cdot \sqrt{nlogq}$,$C < 40$. The pair (A, T_A) is generated by a PPT algorithm $TrapGen\,(1^n, 1^m, q)$.
Lemma2: For matrix $A \in Z_q^{n\times m}$, its trapdoor T_A, real $s \geq C\cdot\sqrt{n\log q}\cdot w\left(\sqrt{\log n}\right)$ and any vector $u \in Z_q^n$, there exists a PPT algorithm $SamplePre\,(A, T_A, u, s)$ outputting z $\leftarrow D_{Z^m,s}\,(x)$ that is statistically $Az = u\,(\mathrm{mod}q)$.

2.4 Decomposition-Extension technique

Lemma3[12]: Let $p = \lfloor \log\beta \rfloor + 1$; $\beta_1 = \lceil \beta/2 \rceil$; $\beta_2 = \lceil (\beta - \beta_1)/2 \rceil$; $\beta_3 = \lceil (\beta - \beta_1 - \beta_2)/2 \rceil$;. . . ; $\beta_p = 1$. For any vector $x \in [-\beta; \beta]^m$, there exists vectors $\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_p \in \{-1, 0, 1\}^m$ such that $x = \sum_{j=1}^{p} \beta_j \tilde{x}_j$. Let$\lambda^{(-1)}$,$\lambda^{(0)}$, and$\lambda^{(1)}$ be number of coordinates -1,0, and 1 in vector $\tilde{x}_j \in \{-1, 0, 1\}^m$. Let $x_j = (\tilde{x}_j || \hat{x}_j) \in B_{3m}$, where $\tilde{x}_j \in \{-1, 0, 1\}^3$ m is randomly selected that has exactly $\left(m - \lambda^{(-1)}\right)$ coordinates -1, $\left(m - \lambda^{(0)}\right)$ coordinates 0 and $\left(m - \lambda^{(1)}\right)$ coordinates 1, it has the following properties:
1, For each random permutation of x_j belongs to B_{3m}, it implies that $x_j \in B_{3m}$ and $\tilde{x}_j \in \{-1, 0, 1\}^m$, consequently,$x \in [-\beta; \beta]^m$.
2, Through appending zero-columns, matrix$A \in Z_q^{n\times m}$ extends to matrix $A^* \in Z_q^{n\times 3m}$ such that $Ax = A^* \sum_{j=1}^{p} \beta_j x_j = u \bmod q$. By adding the uniformly random selected "hidden" vector $r_1, r_2, ..., r_p \in Z_q^{3m}$, the equation of
$A^* \sum_{j=1}^{p} \beta_j (x_j + r_j) - u = A^* \sum_{j=1}^{p} \beta_j r_j \bmod q$ is established.

2.5 Zero-knowledge protocols

S. Goldwasser et al. put forward the concept of zero-knowledge proof in the 1980s. There are the prover and the verifier, and the prover should prove their information without revealing any useful information to the verifier. The whole process is as follows: first, the prover sends a commitment COM. Next, the verifier initiates any challenge value Ch. And then, the prover responds RSP according to the commitment value COM and challenge value Ch. Finally, the verifier verifies whether the prover has the correct information according to COM, Ch and RSP. Fiat, Amos, and Adi Shamir [17] proposed to convert interactive into non-interactive through a hash function. According to the one-way and random characteristics of the hash function, one interaction can be reduced through the hash operation. Since zero-knowledge proof is a probabilistic proof scheme, in order to verify the correctness of the protocol, multiple operations are required to make the error probability close to 0.

3 Underlying interactive protocol

The underlying interaction protocol in this paper uses decomposition extension technology to convince the verifier that the prover has the information without

revealing any useful information. We use the decomposition-expansion method, add "masking" vectors, and Change sort to realize information hiding. In this algorithm, the prover's goal is to convince the verifier in zero-knowledge that:
a) $A \cdot x = u \bmod q$, $x \in Secret_{\beta}(d)$, where $A \in Z_q^{n\times(2l+1)m}$, $x \in Z_q^{(2l+1)m}$ and $u \in Z_q^n$.
b) $\hat{V} \cdot r + I \cdot f = v \bmod q$, $\|f\|_\infty \le \beta$, where $\hat{V} \in Z_q^{m\times m}$,$f \in Z_q^m$,$\|f\|_\infty \le \beta$,$r \in Z_q^m$,$I \in Z_q^{m\times m}$and $v \in Z_q^m$.
c) $Pe + \left(0^{k_1-l} || \lfloor q/2 \rfloor d\right) = c\,(modq)$, $\|e\|_\infty \le b$, where $P \in Z_q^{k_1\times k_2}$,$c \in Z^{k_1}$,$e \in Z^{k_2}$, and $d \in \{0,1\}^l$.
ISIS problem ($A \cdot x = u \bmod q$) is often used to generate key pairs in signature algorithms. Matrices A and vector u are used as public keys and vector x that satisfies $\|x\|_\infty \le \beta$ as private keys. We will introduce in detail how the prover proves that he/she has a private key x under the zero-knowledge.
Let $p = \lfloor \log\beta \rfloor + 1$; $\beta_1 = \lceil \beta/2 \rceil$; $\beta_2 = \lceil (\beta - \beta_1)/2 \rceil$; $\beta_3 = \lceil (\beta - \beta_1 - \beta_2)/2 \rceil$;...; $\beta_p = 1$. For vector $x \in Z_q^{(2l+1)m}$, there exists vectors $\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_p \in \{-1,0,1\}^{(2l+1)m}$ such that $x = \sum_{j=1}^{p} \beta_j \tilde{x}_j$. We can extend $\tilde{x}_j \in \{-1,0,1\}^{(2l+1)m}, j \in [p]$ to $x_j \in B_{(2l+1)3m}$ by technique of Lemma3. Let $A^* = \left[A|0^{n\times(2l+1)2m}\right] \in Z_q^{n\times(2l+1)3m}$. We can get $Ax = A^* \sum_{j=1}^{p} \beta_j x_j = u \bmod q$ and $A^* \sum_{j=1}^{p} \beta_j \left(x_j + r_x^{(j)}\right) - u = A^* \sum_{j=1}^{p} \beta_j r_x^{(j)} \bmod q$, where $r_x^{(j)} \in Z_q^{(2l+1)3m}$.
As the same reason, for $\hat{V}\cdot r + I\cdot f = v \bmod q$, we can get $\hat{V}^* r^* + I^* \left(\sum_{j=1}^{p} \beta_j f_j\right) = \hat{V} \cdot r + I \cdot f = v \bmod q$ and $\hat{V}^* (r^* + r_r) + I^* \left(\sum_{j=1}^{p} \beta_j \left(f_j + r_f^{(j)}\right)\right) - v = \hat{V}^* r_r + I^* \left(\sum_{j=1}^{p} \beta_j r_f^{(j)}\right) \bmod q$, where $r^* \in B_{2m}$, $r_r \in Z_q^{2m}$, $f_j \in B_{3m}$, $r_f^{(j)} \in Z_q^{3m}$, $\hat{V}^* = \left[\hat{V}|0^{m\times 2m}\right] \in Z_q^{m\times 3m}$and $I^* = \left[I|0^{m\times 2m}\right] \in Z_q^{m\times 3m}$.
For $Pe + \left(0^{k_1-l} || \lfloor q/2 \rfloor d\right) = c \bmod q$, we can get$P^* \left(\sum_{j=1}^{\bar{p}} b_j e_j\right) + Qd^* = Pe + \left(0^{k_1-l} || \lfloor q/2 \rfloor d\right) = c \bmod q$ and $P^* \left(\sum_{j=1}^{\bar{p}} b_j \left(e_j + r_e^{(j)}\right)\right) + Q(d^* + r_d) - c = P^* \left(\sum_{j=1}^{\bar{p}} b_j \cdot r_e^{(j)}\right) + Q \cdot r_d \bmod q$, where $\bar{p} = \lfloor \log b \rfloor + 1$; $b_1 = \lceil b/2 \rceil$; $b_2 = \lceil (b - b_1)/2 \rceil$; $b_3 = \lceil (b - b_1 - b_2)/2 \rceil$;...;$b_p = 1$, $P^* = [P|0^{k_1\times 2k_2}] \in Z_q^{k_1\times 3k_2}$, $Q = \begin{pmatrix} 0^{(k_1-l)\cdot l} & 0^{(k_1-l)\cdot l} \\ \lfloor q/2 \rfloor I_l & 0^{l\cdot l} \end{pmatrix} \in \{0, \lfloor q/2 \rfloor\}^{k_1\times 2l}$, $e_j \in B_{3k_2}$, $r_e^{(j)} \in Z_q^{3k_2}$, $d^* \in B_{2l}$ and $r_d \in Z_q^{2l}$.
P represents the prover, and V represents the verifier. The details of the underlying protocol are as follows:
1 Commitment
P samples: $\left(r_x^{(j)}\right)_{j=1}^{p} \in Z_q^{(2l+1)3m}$, $\left(r_e^{(j)}\right)_{j=1}^{\bar{p}} \in Z_q^{3k_2}$, $r_d \in Z_q^{2l}$, $r_r \in Z_q^{2m}$, $\left(r_f^{(j)}\right)_{j=1}^{p} \in Z_q^{3m}$, $\tau \in S_{2l}$, $\pi \in S_{2m}$, $(\psi_j)_{j=1}^{p} \in S_{(2l+1)3m}$, $(\rho_j)_{j=1}^{p} \in S_{3m}$, $(\zeta_j)_{j=1}^{\bar{p}} \in S_{3k_2}$

Then P sends the commitment $CMT = (c_1, c_2, c_3)$to V, where

$$c_1 = COM\left(\begin{array}{l}\tau;\pi;(\psi_j)_{j=1}^{p};(\rho_j)_{j=1}^{p};(\zeta_j)_{j=1}^{\bar{p}};A^*\left(\sum_{j=1}^{p}\beta_j r_x^{(j)}\right);\\ P^*\left(\sum_{j=1}^{\bar{p}} b_j\cdot r_e^{(j)}\right)+Q\cdot r_d;I^*\left(\sum_{j=1}^{p}\beta_j\cdot r_f^{(j)}\right)+\hat{V}^*\cdot r_r\end{array}\right) \tag{7}$$

$$c_2 = COM\left(\begin{array}{l}\tau\left(r_d\right);\pi\left(r_r\right);\left(\psi_j\left(r_x^{(j)}\right)\right)_{j=1}^{p};\\ \left(\rho_j\left(r_f^{(j)}\right)\right)_{j=1}^{p};\left(\zeta_j\left(r_e^{(j)}\right)\right)_{j=1}^{\bar{p}}\end{array}\right) \tag{8}$$

$$c_3 = COM\left(\begin{array}{l}\tau\left(d^*+r_d\right);\pi\left(r^*+r_r\right);\left(\psi_j\left(x_j+r_x^{(j)}\right)\right)_{j=1}^{p};\\ \left(\rho_j\left(f_j+r_f^{(j)}\right)\right)_{j=1}^{p};\left(\zeta_j\left(e_j+r_e^{(j)}\right)\right)_{j=1}^{\bar{p}}\end{array}\right) \tag{9}$$

2 Challenge
V sends the challenge $Ch \leftarrow \{1,2,3\}$to P.
3 Response
P computes the response RSP depending on Ch, and returns it to V.
Case $Ch = 1$: For each $j \in [p]$, let $v_{r_d} = \tau\left(r_d\right), t_d = \tau\left(d^*\right)$, $v_{r_r} = \pi\left(r_r\right), t_r = \pi\left(r^*\right)$, $v_{r_x}^{(j)} = \psi_j\left(r_x^{(j)}\right), t_x^{(j)} = \psi_j\left(x_j\right)$ and $v_{r_f}^{(j)} = \rho_j\left(r_f^{(j)}\right), t_f^{(j)} = \rho_j\left(f_j\right)$; For each$j \in [\bar{p}]$, let $v_{r_e}^{(j)} = \zeta_j\left(r_e^{(j)}\right)$ and $t_e^{(j)} = \zeta_j\left(e_j\right)$. Then P sends:

$$RSP = \left(\begin{array}{l}v_{r_d};t_d;\left(v_{r_x}^{(j)}\right)_{j=1}^{p};\left(t_x^{(j)}\right)_{j=1}^{p};\left(v_{r_f}^{(j)}\right)_{j=1}^{p};\\ \left(t_f^{(j)}\right)_{j=1}^{p};\left(v_{r_e}^{(j)}\right)_{j=1}^{\bar{p}};\left(t_e^{j}\right)_{j=1}^{\bar{p}}\end{array}\right) \tag{10}$$

Case $Ch = 2$: For each $j \in [p]$, let $\tau' = \tau, \pi' = \pi, \psi_j' = \psi_j, \rho_j' = \rho_j$, $w_x^{(j)} = x_j + r_x^{(j)}$, $w_r = r^* + r_r$ and $w_f^{(j)} = f_j + r_f^{(j)}$. For each $j \in [\bar{p}]$, $\zeta_j' = \zeta_j$, $w_e^{(j)} = e_j + r_e^{(j)}$ and $w_d = d^* + r_d$. Then P sends:

$$RSP = \left(\begin{array}{l}\tau';\pi';\left(\psi'_j\right)_{j=1}^{p};\left(\rho'_j\right)_{j=1}^{p};\left(\zeta'_j\right)_{j=1}^{\bar{p}};\\ \left(w_x^{(j)}\right)_{j=1}^{p};w_r;\left(w_f^{(j)}\right)_{j=1}^{p};\left(w_e^{(j)}\right)_{j=1}^{\bar{p}};w_d\end{array}\right) \tag{11}$$

Case$Ch = 3$: For each $j \in [p]$, let $\tau'' = \tau, \pi'' = \pi, \psi_j'' = \psi_j, \rho_j'' = \rho_j$, $y_x^{(j)} = r_x^{(j)}$, $y_d = r_d$ and $y_f^{(j)} = r_f^{(j)}$. For each $j \in [\bar{p}]$, $\zeta_j'' = \zeta_j$,$y_e^{(j)} = r_e^{(j)}$ and $y_r = r_r$. Then P sends:

$$RSP = \left(\begin{array}{l}\tau'';\pi'';\left(\psi''_j\right)_{j=1}^{p};\left(\rho''_j\right)_{j=1}^{p};\left(\zeta''_j\right)_{j=1}^{\bar{p}};\\ \left(y_x^{(j)}\right)_{j=1}^{p};\left(y_f^{(j)}\right)_{j=1}^{p};\left(y_e^{(j)}\right)_{j=1}^{\bar{p}}\end{array}\right) \tag{12}$$

Verification: V verifies depending on COM, RSP, and Ch, the verification process as follow:

Case $Ch = 1$: Pares RSP as in(10),check that $t_d \in B_{2l}$, $t_x^{(j)} \in B_{(2l+1)3m}$, $t_r \in B_{2m}$, $t_f^{(j)} \in B_{3m}$, $t_e^{(j)} \in B_{3k_2}$ and that

$$\begin{cases} c_2 = COM\left(v_{r_d}; v_{r_r}; \left(v_{r_x}^{(j)}\right)_{j=1}^{p}; \left(v_{r_f}^{(j)}\right)_{j=1}^{p}; \left(v_{r_e}^{(j)}\right)_{j=1}^{\bar{p}}\right) \\ c_3 = COM\left(\begin{array}{l} v_{r_d} + t_d; v_{r_r} + t_r; \left(v_{r_x}^{(j)}\right)_{j=1}^{p} + \left(t_x^{(j)}\right)_{j=1}^{p}; \\ \left(v_{r_f}^{(j)}\right)_{j=1}^{p} + \left(t_f^{(j)}\right)_{j=1}^{p}; \left(v_{r_e}^{(j)}\right)_{j=1}^{\bar{p}} + \left(t_e^{j}\right)_{j=1}^{\bar{p}} \end{array}\right) \end{cases}$$

.

Case $Ch = 2$, Parse RSP as in (11), Check that:

$$\begin{cases} c_1 = COM\left(\begin{array}{l} \tau'; \pi'; \left(\psi'_j\right)_{j=1}^{p}; \left(\rho'_j\right)_{j=1}^{p}; \left(\zeta'_j\right)_{j=1}^{\bar{p}}; A^*\left(\sum_{j=1}^{p} \beta_j w_x^{(j)}\right) - u; \\ P^*\left(\sum_{j=1}^{\bar{p}} b_j \cdot w_e^{(j)}\right) + Q \cdot w_d - c; I^*\left(\sum_{j=1}^{p} \beta_j \cdot w_f^{(j)}\right) + \hat{V}^* \cdot w_r - v \end{array}\right) \\ c_3 = COM\left(\tau'\left(w_d\right); \pi'\left(w_r\right); \left(\psi'_j\left(w_x^{(j)}\right)\right)_{j=1}^{p}; \left(\rho'_j\left(w_f^{(j)}\right)\right)_{j=1}^{p}; \left(\zeta_j\left(w_e^{(j)}\right)\right)_{j=1}^{\bar{p}}\right) \end{cases}$$

Case $Ch = 3$, Parse RSP as in (12), Check that:

$$\begin{cases} c_1 = COM\left(\begin{array}{l} \tau''; \pi''; \left(\psi''_j\right)_{j=1}^{p}; \left(\rho''_j\right)_{j=1}^{p}; \left(\zeta''_j\right)_{j=1}^{\bar{p}}; A^*\left(\sum_{j=1}^{p} \beta_j y_x^{(j)}\right); \\ P^*\left(\sum_{j=1}^{\bar{p}} b_j \cdot y_e^{(j)}\right) + Q \cdot y_d; I^*\left(\sum_{j=1}^{p} \beta_j \cdot y_f^{(j)}\right) + \hat{V}^* \cdot y_r \end{array}\right) \\ c_2 = COM\left(\tau''\left(y_d\right); \pi''\left(y_r\right); \left(\psi''_j\left(y_x^{(j)}\right)\right)_{j=1}^{p}; \left(\rho''_j\left(y_f^{(j)}\right)\right)_{j=1}^{p}; \left(\zeta''_j\left(y_e^{(j)}\right)\right)_{j=1}^{\bar{p}}\right) \end{cases}$$

In each case, V outputs 1 if and only if all the conditions hold. Otherwise, it output 0.

4 The VLR Dynamic Group Signature Scheme

Description of the scheme: We specify the parameters of the scheme. Let n be the security parameter, N be the maximum expected number of group users, $q = w\left(n^2 \log n\right)$, $m \geq 2n \log q$, $\sigma = w\left(\sqrt{n \log q \log n}\right)$, $\beta = \lceil \sigma \cdot \log m \rceil$ $s.t(4\beta+1)^2 \leq q$, $p = \lfloor \log \beta \rfloor + 1$, $t = w\left(\log n\right)$, $l = \log N$, $k_1 := m + l$ and $k_2 := n + m + l$. Let integer b be the norm bound for LWE noises such that $q/b = l\tilde{O}\left(n\right)$, and let $id_i, i \in Z$ as the identity of all users. (For example, ID number.)

Choose hash functions $H : \{0,1\}^* \rightarrow \{1,2,3\}^t$,$H_2 : \{0,1\}^* \rightarrow Z_q^m$, and $H_3 : \{0,1\}^* \rightarrow Z_q^{m \times n}$.

Our group signature scheme is described as follows:

Key Generation:$KeyGen\left(1^n, 1^N\right)$

The group manager (GM) performs the following steps:

Step1: Run $\left(A_0, T_{A_0}\right), \left(E, T_E\right) \leftarrow TrapGen\left(m, n, q\right)$, where T_{A_0}, T_E is the trapdoor of A_0 and E, respectively. GM randomly selects matrix$A_1, A_2 \leftarrow Z_q^{n \times m}$ and generates matrix $A = \left(A_0 | A_1^0 | A_1^1 | ... | A_l^0 | A_l^1\right) \in Z_q^{n \times (2l+1)m}$, where $A_i^0 = iA_1, A_i^1 = iA_2$.

Step2:Sample $u \leftarrow Z_q^n$. Then compute $r = H_2\left(id_i\right) \in Z_q^m$and $B = {H_3}^T\left(A_0, A_1, A_2, u\right) \in$

$Z_q^{n\times m}$, where id_i is the user ID number. The GM issues an index $d \in \{0,1,2,...,N-1\}$ to each group user, let $d_{[1]}d_{[2]}...d_{[l]} \in \{0,1\}^l$ denote the binary representation of d, and do the following:
Sample vectors $x_1^{d_{[1]}},...,x_l^{d_{[l]}} \leftarrow D_{z^m,\sigma}$. Compute $z = \sum_{i=1}^{l} A_i^{d_{[i]}} \cdot x_i^{d_{[i]}} \bmod q$. Run $x_0 \leftarrow SamplePre\left(T_{A_0}, A_0, u-z, \sigma\right)$. Let $x_1^{1-d_{[1]}},...,x_l^{1-d_{[l]}}$ be m-dimensional zero-vectors. Define $x^{(d)} = \left(x_0||x_1^0||x_1^1||...||x_l^0||x_l^1\right) \in Z_q^{(2l+1)m}$. If $\left\|x^{(d)}\right\|_\infty \leq \beta$ then continue; else, resample until the conditions are met.
Let the group user private key be$gsk\left[d\right] = x^{(d)}$, and the revocation token be $grt\left[d\right] = B \cdot r \in Z_q^n$.
Step3:Output
Group public key: $gpk = \left(\left(A_0, A_1, A_2\right), E, u, B\right)$. Tracking key: $gsk = T_E$. Group private key: $gmsk = \left(gsk\left[0\right], gsk\left[1\right], ..., gsk\left[N-1\right]\right)$. Revocation token: $grt = \left(grt\left[0\right], grt\left[1\right], ..., grt\left[N-1\right]\right)$.
Join and revoke group users:
Join:Let RL is the initial revocation list and $St_{revoca}$is the effective label set of group members .To prevent illegal user tags from being added to the revocation list, they forge their identity to re-apply for the group. Users are required to use their unique identification ID (for example, ID number) as their entry certificate. The details of joining the group are as follows:
User samples $f \leftarrow \chi^\beta$ and computes $r_j = H_2\left(id_j\right) \in Z_q^m$, V=$H_3\left(A_0, A_1, A_2, B, \mathrm{u}\right) \in Z_q^{m\times n}$ and v_j=$V \cdot \left(B \cdot \mathrm{r_j}\right) + f$. Let $grt\left[j\right] = B \cdot r_j$. Use the common key in PKI to sign $grt\left[j\right]$, and the signature is $sig = Sign\left(grt\left[j\right]\right)$. Send $grt\left[j\right]$ and the signature sig to GM.
GM verifies whether $grt\left[j\right]$ has been registered. Calculate ${f_i}' = v_j - V\left(B \cdot r_i\right)\left(\mathrm{mod} q\right)$ for all users. Refuse to join if $\left\|{f_i}'\right\|_\infty \leq \beta$ or the signature verification is incorrect. Otherwise, to join.
GM runs the key generation algorithm to calculate the new user's private key, and distributes it to the new members. The user's revocation token $grt\left[j\right]$ is added to the effective label set St_{revoca}, updated and announced. The join is completed.
Revoke:When revoking, the user's revocation token is added to the revoking list RL and announced. There is no need to update all keys again.
Signing Algorithm:$Sign\left(gsk\left[d\right], M\right)$
Group user $user_j, j \in (0,1,...,N-1)$ uses the secret key $gsk\left[d\right] = x \in Secret_\beta\left(d\right)$ to sign a massage $M \in \{0,1\}^*$. $user_j$ samples a matrix $G \in Z_q^{n\times l}$ to encrypt own index $d \in \{0,1,2,...,N-1\}$, and signs. The $user_j$ performs the following step:
Step1: Let $G \in Z_q^{n\times l}$. Samples $\leftarrow \chi^n; e_1 \leftarrow \chi^m, e_2 \leftarrow \chi^l$, then compute the ciphertext of index d:$c_1 = E^T s + e_1, c_2 = G^T s + e_2 + \lfloor q/2 \rfloor\, d \in Z_q^m \times Z_q^l$.
Step2:Let $P = \begin{pmatrix} E^T \\ G^T\ I_{m+l} \end{pmatrix} \in Z_q^{k_1\times k_2}$, $c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \in Z^{k_1}$, $e = \begin{pmatrix} s \\ e_1 \\ e_2 \end{pmatrix} \in Z^{k_2}$.
According to the ciphertext of d, we have $\|e\|_\infty \leq b$ and $Pe + \left(0^{k_1-l}||\,\lfloor q/2 \rfloor\, d\right) = c \bmod q$, where b is the norm bound for LWE noises such that $q/b = l\tilde{O}\left(n\right)$.

Step3:Sample $f \leftarrow \chi^{\beta}$. Compute $V{=}H_3\left(A_0, A_1, A_2, B, \mathrm{u}\right) \in Z_q^{m\times n}$ and $V \cdot (B \cdot r) + f = v \bmod q$. Let $\hat{V} = V \cdot B$, we have $\hat{V} \cdot r + I \cdot f = v \bmod q$.
Step4:The input $\left((A, u, B, V, v, P, c), x, d, r, e\right)$ conforms to the underlying interaction protocol described in the second chapter, and the protocol runs $t = w(\log n)$ times so that the soundness error is negligible. The protocol is transformed into non-interactive triples $\Pi{=}\left(\left\{CMT^{(k)}\right\}_{k=1}^{t}, CH, \left\{RSP^{(k)}\right\}_{k=1}^{t}\right)$ by Fiat - Shamir heuristic, where

$$CH = \left(\left\{Ch^{(k)}\right\}_{k=1}^{t}\right) = H\left(M, \left\{CMT^{(k)}\right\}_{k=1}^{t}, (c_1, c_2), G, u, v\right) \in \{1,2,3\}^{t}$$

.
Step5:Output the group signature $\Sigma{=}\left(M, \Pi, G, (c_1, c_2), u, v\right)$
Verification Algorithm:$Verify\left(pgk, M, \Sigma, RL\right)$
Step1: Parse the signature Σ as $(M, \Pi, G, (c_1, c_2), u, v)$. If

$$\left\{Ch^{(k)}\right\}_{k=1}^{t} \neq H\left(M, \left\{CMT^{(k)}\right\}_{k=1}^{t}, (c_1, c_2), G, u, v\right)$$

, then return 0.
Step2:Generating $P$and C according to the signature process, the verifier runs the underlying protocol, checks the validity of $\left\{RSP^{(k)}\right\}_{k=1}^{t}$ according to $\left\{Ch^{(k)}\right\}_{k=1}^{t}$ and $\left\{CMT^{(k)}\right\}_{k=1}^{t}$ which provided by the signer. If any condition is not established, then returns 0.
Step3: For each $u_i \in RL$, calculate$f_i{}' = v - V\left(B \cdot r_i\right) (\mathrm{mod} q)$. Verify whether there exists u_i so that $\left\|f'\right\|_{\infty} \leq \beta$. If it exists, then return 0.
Step4:Return 1.
Open Algorithm: $Open\left(gmsk, M, \Sigma\right)$
GM uses the tracking private key T_A to track the index of the signer as follows:
Step1:Let $G = [g_1|g_2|...|g_l]$. Sample $y_i \leftarrow SamplePre\left(T_E, E, \mathrm{g}_i, s\right), i \in [l]$. GM gets matrix $Y = [y_1|y_2|...|y_l] \in Z_q^{m\times l}$ such that $E \cdot Y = G (\mathrm{mod} q)$.
Step2:Calculate $d' = \left(d_1{}', ..., d_l{}'\right) = c_2 - Y^T c_1 \in Z_q^l$ Because $e_1$and e_2 are very small, when the result is close to 0, then let $d_i = 0$, when the result is close to $\lfloor q/2 \rfloor$, then let $d_i = 1$.
Step3:Return $d = (d_1, ..., d_l) \in \{0,1\}^l$,which is the index of the signer.

5 Analysis of the Scheme

5.1 Correctness analysis

Through the analysis of the signature and verification of the algorithm, we can see that the signature uses non-interactive zero-knowledge proof to ensure the normal operation of the algorithm without revealing any information, with correctness and security. $\left\{Ch^{(k)}\right\}_{k=1}^{t}{=}H\left(M, \left\{CMT^{(k)}\right\}_{k=1}^{t}, (c_1, c_2)\right)$ ensures the integrity and authenticity of the information M. we use RL to detect the legitimacy of the signer and realize the VLR function of the algorithm.

Lemma4[8]: Let β=$ploy(n), q \geq (4\beta+1)^2, m \geq 3n$. For each random matrix $V \in Z_q^{m\times n}$ and non-zero vector $s \in Z_q^n$, satisfying $Pr\left[\exists s \in Z_q^n : \|V \cdot s\|_\infty \leq 2\beta\right] \leq negl(n)$.

According to the algorithm, Zero-knowledge proof ensures that the signer has the private key, revocation token, and correct ciphertext of index $d = (d_1, ..., d_l) \in \{0,1\}^l$. According to step3 of verification algorithm, we can judge whether the revocation token is valid.

Proof:For each j, the vector $f_i{}'$ can be delivered as

$f_i{}' = v - V(B \cdot r_i)(\mathrm{mod}q)$ =$V \cdot grt[d] + f - V(B \cdot r_i) = V \cdot grt[d] + f - V \cdot u_i$ $= V \cdot (grt[d] - u_i) + f$.

According to Lemma4, for an honest signer, $grt[d] - u_i \neq 0 \in Z_q^n$ and $\left\|f_i{}'\right\|_\infty > \beta$. For an illegal user, the revocation token is in the revocation list, therefore $grt[d] - u_i$=0,$f_i{}'$=f and $\left\|f_i{}'\right\|_\infty$=$\|f\|_\infty \leq \beta$.

For open algorithm, take $E \cdot Y = G(\mathrm{mod}q)$ into $c_2 - Y^T c_1$ and get

$$\begin{aligned} c_2 - Y^T c_1 &= G^T s + e_2 + \lfloor q/2 \rfloor d - \left(G^T \left(E^T\right)^{(-1)}\right)\left(E^T s + e_1\right) \\ &= e_2 + \lfloor q/2 \rfloor d - \left(G^T \left(E^T\right)^{(-1)}\right) e_1 \in Z^n \end{aligned}$$

Since e_1 and $e_2$are small, the signer's index d can be successfully decrypted by judging whether $c_2 - Y^T c_1$ is close to 0 or $q/2$.

5.2 Fully Anonymity

Our VLR group signature scheme is fully-anonymous in the random oracle model. Let A be the adversary of the PPT computing power and prove its anonymity through a sequence of indistinguishable experiments G_0, G_1, G_2, G_3, G_4, where $Adv_A(G_0) = \varepsilon$,$Adv_A(G_4) = 0$.

Experiment G_0: Suppose attacker A has the advantage of ε, the challenge is successful. And it is allowed to query the private key, revocation token, and signature. The challenger runs $KeyGen\left(1^n, 1^N\right)$ to obtain gpk, gsk, grt= $\{grt[i]\}_{i=0}^{N-1}$, $gmsk$= $\{gsk[i]\}_{i=0}^{N-1}$, and gives gpk to the A. Set lists L_1,L_2 and L_3 store the results of revocation token query, private key query, and signature query, respectively. Their initial state is 0.

a)Revocation token query: Adversary A queries the revocation token of d. The challenger returns $grt[d]$ and stores the result in L_1.

b)Private key query: Adversary A queries the private key of user of index d. The challenger returns $gsk[d]$ and stores the result in L_2.

c)Signature query: Adversary A queries the Signature on any message M by user of index$d$. The challenger returns Σ= $(M, \Pi, G, (c_1, c_2), u, v)$ and stores the result in L_3.

The adversary A begin to challenge. A sends a massage M^* and indexes $d_0, d_1 \in \{0,1\}^l$ which are not queried to the challenger. The challenger returns the valid signature

$$\Sigma^*= (M^*, \Pi^*, G^*, ({c_1}^*, {c_2}^*), u^*, v^*) \leftarrow Sign(gpk, gsk[d_b], M)$$

, where $b \in \{0,1\}$. A can still make queries as before except index $d_0, d_1 \in \{0,1\}^l$. A outputs $b' \in \{0,1\}$, if d_b=$d_{b'}$ challenge succeeds, otherwise, it fails.
Experiment G_1: In this experiment, the challenger didn't generate a valid signature Σ, but used a valid simulator to simulate. According to the underlying interaction protocol used in this scheme, the demonstration system is statistically zero-knowledge. Therefore, G_0 and G_1 are statistically indistinguishable.
ExperimentG_2: In the challenge phase, the returned signature is

$$\Sigma^* = (M, \Pi^*, G^*, (c_1{}^*, c_2{}^*), u^*, v^*)$$

, where $v^* = V \cdot grt[d_b] + f \pmod q, b \in \{0,1\}$,$f$ sample in error distribution χ. In this experiment, the challenger uniformly random sampled $t \leftarrow Z_q^n$ and calculated $v^* = V \cdot t + f \pmod q$. Replace $grt[d_b]$ with t, and the rest remains unchanged, so G_1 and G_2 are statistically indistinguishable.
Experiment G_3: In this experiment, the challenger modify the generation of the ciphertext $(c_1{}^*, c_2{}^*)$. Review the signature algorithm in Chapter 4, one has $c_1{}^* = (E^*)^T s + e_1, c_2{}^* = (G^*)^T s + e_2 + \lfloor q/2 \rfloor d_b \in Z_q^m \times Z_q^l$, where s, e_1, e_2, G^* are uniformly random. Let $c_1{}^* = z_1, c_2{}^* = z_2 + \lfloor q/2 \rfloor d_b$, where $x_1 \in Z^m, x_2 \in Z^l$. According to the LWE difficult problem, A has no way to distinguish $\left(E^*, (E^*)^T s + e_1\right)$ and $\left(G^*, (G^*)^T s + e_2\right)$from (E^*, z_1) and (G^*, z_2), respectively, which implies that G_2 and G_3 are computationally indistinguishable.
Experiment G_4: The Challenger make a conceptual modification so that the signature has nothing to do with the identity index. Let $c_1{}^* = z_1{}', c_2{}^* = z_2{}'$,where $x_1{}' \in Z^m, x_2{}' \in Z^l$ are uniformly random. It is clear that G_3 and G_4 are statistically indistinguishable. Moreover, because this experiment does not rely on the challenger's index b, the advantage of adversary A in this experiment is 0.
The above five experiments show that our VLR group signature scheme is fully anonymous in the random oracle model, and the adversary A does not have any advantage in this scheme.

5.3 Fully Traceability

In the random oracle model, the VLR group signature scheme is fully traceability if the $SIS^{\infty}_{n,(l+1)m,q,2\beta}$ problem is hard.
Lemma5[a]: If there is a traceability adversary A with success probabilityε and running time T, then there exists an algorithm F that solves the $SIS^{\infty}_{n,(l+1)m,q,2\beta}$ problem with success probability $\varepsilon' > \left(1 - (7/9)^t\right) \cdot 1/2N$ and running time $T' = 32 \cdot Q_H/(\varepsilon - 3^{-t}) + poly(n, N)$, where Q_H is the number of queries to the random oracle $H : \{0,1\}^* \to \{1,2,3\}^t$.
Suppose that there is an adversary A who breaks the computational binding property of the commitment scheme COM with a non-negligible probability ε, so the adversary A can find an effective method to solve the $SIS^{\infty}_{n,(l+1)m,q,2\beta}$ problem. Generally, the string commitment schemes COM used by the underlying protocol is computationally bound. We can construct a forger F with PPT

computing power to solve the $SIS^{\infty}_{n,(l+1)m,q,2\beta}$ problem with non-negligible probability.

Given the verification key (A_0, A_1, A_2, u), F runs $TrapGen\,(m, n, q)$ to obtain (E, T_E). F interacts with the adversary A by sending $gpk = (A_0, A_1, A_2, u, E)$ and responding to A's query. The specific response query process is as follows:

Random oracle H queries: Return the uniformly random values in $\{1, 2, 3\}^t$. Assuming that A queries H at most Q_H times, for each $k \leq Q_H$, we use r_k denotes the answer to the $k-th$ query.

Corruption queries:Let corruption set U_a store the results of private key query, which the initial state is 0. If the adversary A asks the private key of user i, then F adds i to the U_a and returns the $gsk\,[i]$.

Signatures queries:If A requires user $i$to sign the information M, then F returns $\Sigma = (M, \Pi', G, (c_1, c_2)\,, u, v)$, where Π' is simulated by the simulator without a legal user, and the others are real. According to the zero knowledge of the underlying interaction protocol, the signature Σ is indistinguishable from the legal group signature.

Finally, A sends a message M^*, revocation set RL^* and a non-trivial forged signature $\Sigma^* = (M, \Pi', G, (c_1, c_2)\,, u, v)$, such that $Verify\,(gpk, M^*, RL^*, \Sigma^*) = Valid$ and open fails or outputs the index of non-group members.

Adversary A uses forger F to forge group signature. The method is as follows: By analyzing, A does not know that H input is $\left(M, \left\{CMT^{(k)}\right\}^t_{k=1}, (c_1, c_2)\right)$, and at least 3^{-t} probability can completely guess

$$(Ch_1, ..., Ch_t) = H\left(M, \left\{CMT^{(k)}\right\}^t_{k=1}, (c_1, c_2)\right)$$

. Therefore, with probability at least $\varepsilon - 3^{-t}$, there exists $k^* \leq Q_H$ such that the $k-th$ random oracle H queries involves the tuple $\left(M, \left\{CMT^{(k)}\right\}^t_{k=1}, (c_1, c_2)\right)$. For a fixed k^*, execute A many times. For the query before k^*, the query result remains unchanged. That is, the challenge value $(Ch_1, ..., Ch_{k^*-1})$ remains unchanged. Starting from the $k^* - th$ query, the new random value is output. According to lemma 5, with a probability greater than 1/2, the output is obtained after executing A less than $32 \cdot Q_H/(\varepsilon - 3^{-t})$ times: $r^{(1)}_{k^*} = \left(Ch^{(1)}_1, ..., Ch^{(1)}_t\right)$,$r^{(2)}_{k^*} = \left(Ch^{(2)}_1, ..., Ch^{(2)}_t\right)$, and $r^{(3)}_{k^*} = \left(Ch^{(3)}_1, ..., Ch^{(3)}_t\right)$.

We can get the probability $Pr\,[\exists j \in \{1, ..., t\}] : \left\{Ch^{(1)}_i, Ch^{(2)}_i, Ch^{(3)}_i\right\} = \{1, 2, 3\} = 1 - (7/9)^t$and the $\left(RSP^{(1)}_i, RSP^{(2)}_i, RSP^{(3)}_i\right)$ under the such index j. According to the underlying interaction protocol used, using the knowledge extractor, we can obtain that the vector (y, f', r', e'), which satisfy the following.

1, $y = \left(y_0||y^0_1||y^1_1||...||y^0_l||y^1_l\right), \|y\|_{\infty} \leq \beta$ and $A \cdot y = u \bmod q$.

2, $\|f'\|_{\infty} \leq \beta$, $V \cdot (B \cdot r) + f' = v \bmod q$.

3, $\|e'\| \leq b$, $Pe' + \left(0^{k_1 - l}||\,\lfloor q/2 \rfloor\, d'\right) = c \bmod q$.

We can observe that c is ciphertext of d', and the open algorithm returns to d', which satisfies $Verify\,(gpk, M^*, RL^*, \Sigma^*) = Valid$ and $Verify\,(gpk, M^*, grt\,[j^*]\,, \Sigma^*) = invalid$. It then follows that $grt\,[j^*] \notin RL$ and $j^* \notin U_a$, therefore (y, d^*) is an

effective forgery. By analyzing the forgery signature, we know that if A has a non-negligible probability of returning valid forged signatures in polynomial time, F also has this ability.
There is no effective algorithm F to solve the $SIS^{\infty}_{n,(l+1)m,q,2\beta}$ problem with the advantage of $\varepsilon' > \left(1-(7/9)^{t}\right) \cdot 1/2N$ in $T' = 32 \cdot Q_H/(\varepsilon - 3^{-t}) + poly\,(n, N)$ times, therefore our algorithm is traceable.

5.4 Efficiency analysis

We select the following four schemes for efficiency analysis compared with this scheme. Ref [8] is the first group signature scheme with a local revocation function. This scheme is a static group signature scheme, which is implicit tracking. Ref [10] uses zero-knowledge proof of the Ref[9] to prove that the size of the public key and signature is small, and there is no revocation function dynamically. The group signature scheme proposed in Ref [12] uses an explicit tracking method, making the opening function simple, however, this scheme is a static signature scheme without a revocation function. Ref [18] uses lattice signature. The signature is small and does not need a trapdoor, which improves the efficiency, but has no revocation function.

Table 1. comparison of efficiency of different schemes.

scheme	Public key size	Signature size	Revoke	Dynamic
Ref [8]	$\tilde{O}\left(n^2 \cdot \log N\right)$	$\tilde{O}\left(n \cdot \log N\right)$	Yes	No
Ref [10]	$\tilde{O}\left(mn \log q\right)$	$\tilde{O}\left(mn \log q\right)$	No	Yes
Ref [12]	$\tilde{O}\left(mn \log \mathrm{N} \cdot \log q\right)$	$\tilde{O}\left(tm \log N \cdot \log q\right)$	No	No
Ref [18]	$\tilde{O}\left(n \cdot (\log N)^2\right)$	$\tilde{O}\left(n\right)$	No	No
This scheme	$\tilde{O}\left(mn \log q\right)$	$\tilde{O}\left(n \cdot \log N\right)$	Yes	Yes

6 Conclusion

The revocation mechanism is a crucial function in group signature. In this scheme, we design a dynamic group signature with a revocation mechanism. The size of the signature is the logarithm of the number of group members, which realizes a balance between the size of the signature and the function of VLR. Through the analysis of correctness and security, our scheme realizes almost all anonymity and traceability and meets the requirements of group signature.

Acknowledgments

Project supported by Joint Funding Project of Natural Science Foundation of China (61370188), Beijing Municipal Education Commission Scientific Research

Project (KM202010015009), Beijing Municipal Education Commission Scientific Research Project Funding (No.KM202110015004), Beijing Institute of Graphic Communication Doctoral Funding Project (27170120003/020), Beijing Institute of Graphic Communication Research Innovation Team Project (Eb202101), Intramural Discipline Construction Project of Beijing Institute of Graphic Communication (21090121021), Key Educational Reform Project of Beijing Institute of Graphic Communication (22150121033/009) and General Research Project of Basic Research of Beijing Institute of Graphic Communication (Ec202201)

References

1. D. Chaum, E. Van Heyst, "Group Signatures". *EUROCRYPT*, Volume 547, pp. 257–265, 1991.
2. S. Devidas, S. Rao Y.V., and N. R. Rekha, "A decentralized group signature scheme for privacy protection in a blockchain," *Int. J. Appl. Math. Comput. Sci.*,Vol.31, No.2,353-364 2021.
3. K. Gu, L. Yang, Y. Wang, and S. Wen, "Traceable Identity-Based Group Signature," *RAIRO-Theor. Inf. Appl.*, vol. 50, no. 3, pp. 193–226, Jul. 2016.
4. V. Chamola, A. Jolfaei, V. Chanana, P. Parashari, and V. Hassija, "Information security in the post quantum era for 5G and beyond networks: Threats to existing cryptography, and post-quantum cryptography," *Comput. Commun.*, vol. 176, pp. 99–118, Aug. 2021.
5. S. E. Yunakovsky et al., "Towards security recommendations for public-key infrastructures for production environments in the post-quantum era," *EPJ Quantum Technol.*, vol. 8, no. 1, p. 14, Dec. 2021.
6. S. D. Gordon, J. Katz, and V. Vaikuntanathan, "A Group Signature Scheme from Lattice Assumptions," *Advances in Cryptology - ASIACRYPT 2010*, vol. 6477, pp. 395–412.
7. F. Laguillaumie, A. Langlois, B. Libert, and D. Stehlé, "Lattice-Based Group Signatures with Logarithmic Signature Size," *Advances in Cryptology - ASIACRYPT 2013*, vol. 8270, pp. 41–61, 2013
8. A. Langlois, S. Ling, K. Nguyen, and H. Wang, "Lattice-Based Group Signature Scheme with Verifier-Local Revocation," *Public-Key Cryptography – PKC 2014*, vol. 8383, pp. 345–361, 2014
9. P. Q. Nguyen, J. Zhang, and Z. Zhang, "Simpler Efficient Group Signatures from Lattices," *Public-Key Cryptography – PKC 2015*, pp. 401–426., 2015
10. Z. C. Li, Y. L. Zhang, Y. X. Zhang and Y. T. Yang, "An Improved Dynamic Group Signature Scheme Based on Lattice," *Wuhan Univ. (Nat. Sci. Ed.)*, vol. 62, no. 02, pp. 135–140, 2016.
11. Q. Ye, N. N. Zhao and Z. Q. Zhao, et.al, "Efficient Fully Dynamic Group Signature Scheme from Lattice," *Computer Engineering*, vol. 47, no. 02, pp. 160-167+175, 2021.
12. S. Ling, K. Nguyen, and H. Wang, "Group Signatures from Lattices: Simpler, Tighter, Shorter, Ring-Based," *Public-Key Cryptography – PKC 2015*, vol. 9020, pp. 427–449, 2015.
13. M. N. S. Perera, T. Nakamura, M. Hashimoto, H. Yokoyama, and K. Sakurai, "Almost fully anonymous attribute-based group signatures with verifier-local revocation and member registration from lattice assumptions," *Theoretical Computer Science*, vol. 891, pp. 131–148, Nov. 2021.

14. S. Ling, K. Nguyen, A. Roux-Langlois, and H. Wang, "A lattice-based group signature scheme with verifier-local revocation," *Theoretical Computer Science*, vol. 730, pp. 1–20, Jun. 2018.
15. M. N. S. Perera and T. Koshiba, "Combined interactive protocol for lattice-based group signature schemes with verifier-local revocation," *lnt.J. Grid and Utility Computing*, Vol.11, No.5, pp. 662-673, 2020
16. M. Ajtai, "Generating Hard Instances of the Short Basis Problem," *Automata, Languages and Programming*, vol. 1644, pp. 1–9, 1999.
17. A. Fiat and A. Shamir, "How To Prove Yourself: Practical Solutions to Identification and Signature Problems," *Advances in Cryptology — CRYPTO' 86* , vol. 263, pp. 186–194, 2006.
18. Q. Y, X. M. Yang, and P. K. Qin, "Novel Against Quantum Attacks Group Signature Scheme Based on NTRU Lattice," *Computer Engineering and Applications*, vol. 56, no. 02, pp. 89–96, 2020.