# 适合大群组的格基动态群签名方案

李雪莲[1*]，吕晓琳[1]，郭利娟[1]，高军涛[2]

(1. 西安电子科技大学数学与统计学院 西安 710126；2. 西安电子科技大学计算机网络与信息安全教育部重点实验室 西安 710071)

【摘要】动态群签名方案的设计难点在于给出有效的群成员撤销机制。该文构造了一种新的撤销机制，撤销时不需要更新群管理员和群成员的任何信息，仅需群管理员或群成员本人与撤销图灵机通信，图灵机确定其身份后将撤销token添加到撤销列表即完成了撤销操作，因此更适用于群成员数量基数较大的群体。利用此撤销机制，提出了一种基于错误学习(LWE)假设和小整数解(SIS)假设的动态群签名方案，支持在任意时刻加入和撤销用户。对比已有方案，该方案的群公钥尺寸固定且更小，用户加入时下载量小，方案效率更高。

**关 键 词** 群签名； 格密码； 撤销列表； Stern协议； VLR撤销

**中图分类号** TN918 **文献标志码** A doi:10.3969/j.issn.1001-0548.2019.01.014

# A Dynamic Group Signature Scheme Based on Lattice for Large Groups

LI Xue-lian[1*], LÜ Xiao-lin[1], GUO Li-juan[1], and GAO Jun-tao[2]

(1. School of Mathematics and Statistics, Xidian University Xi'an 710126;
2. State Key Laboratory of Integrated Services Networks, Xidian University Xi'an 710071)

**Abstract** The challenge of designing a dynamic group signature scheme is to construct an efficient group member revocation mechanism. We design a new revocation mechanism. For the group manager and group member, all need to do is to communicate with the revocation Turing. When the Turing determines their identities, the revocation token is added into the revocation list to complete the revocation operation. So it is more suitable for groups with more members. Using this mechanism, we propose a new dynamic group signature scheme based on learning with errors (LWE) problem and the small integer solution (SIS) assumption, in which any user can join and leave the group at any time. Compared with existing schemes, group public key is fixed in length and shorter. When a user joins into the group, he needs less downloads. So, we can provide a higher efficiency in practical applications.

**Key words** group signature; lattice cipher; revocation list; Stern protocol; VLR revocation

群签名一直是公钥密码体系中的一个研究热点[1]。一个群签名方案通常有两个基本要求：匿名性和可追踪性。这两个特性使得它在很多场景中都有应用，如：可信计算、数字证书管理、匿名在线通信及电子商务系统等。考虑到群签名方案的实用性，用户加入应该是在任意时间都可以发生的。另外，考虑到有些群成员行为不端，或者群成员本人不想再接收群信息等情况的存在，系统需要有相应的成员撤销机制[2]。而且撤销群用户身份时，应该不影响其他群成员签名的安全性，且撤销代价较小。

近几年，基于格的密码学因其潜在优势引起了密码工作者的广泛关注：渐近高效性、抗量子计算安全，以及最坏情况困难性假设。设计安全高效的格基密码方案充满挑战性。2010年，文献[3]构造了一个安全的格基群签名方案，其群公钥和签名尺寸都是$\mathcal{O}(N)$，其中$N$是当前群成员个数。文献[4]给出了一个高效的格基群签名方案，将群公钥和签名尺寸降为对数级$\mathcal{O}(\log N)$。但这些群签名方案都不支持群成员撤销。文献[5]提出一种简单的撤销机制——验证本地可撤销(verifier-local revocation, VLR)模型。由于这种撤销方式只需验证者下载撤销链表，

收稿日期：2017－07－18；修回日期：2017－12－04
基金项目：国家重点研发计划(2016YFB0800601)；国家自然科学基金(61303217，61502372)；陕西省自然科学基金(2013JQ8002，2014JQ8313)
作者简介：李雪莲(1979－)，女，副教授，主要从事通信与信息系统方面的研究. E-mail: xuelian202@163.com

且撤销时运算量小，较符合实际应用的要求。2014年，文献[6]成功将VLR撤销机制应用到格基群签名方案中。2016年，文献[7]基于SIS和LWE假设，在文献[8]的工作上构造了一个动态群签名方案，该方案允许用户在任意时间加入群，但缺少群成员撤销算法。最近，文献[9]基于Merkle哈希树构造了一个完全动态群签名方案，但撤销成员时，需更新哈希树。

VLR模型不具有追踪到具体用户的功能，为了获得一种可追踪动态群签名，本文对VLR撤销机制和文献[7]的方法进行了仔细研究，并参考文献[10]中的动态群签名模型，构造了一个新的群签名方案，该方案允许用户动态加入和撤销，且与已有方案相比，群公钥尺寸较小。

# 1　基础知识

对 $\boldsymbol{x}=(x_1,x_2,\cdots,x_n)\in\mathbb{R}^n$，及 $1\leqslant i_1\leqslant i_2\leqslant n$，定义 $\mathrm{Parse}(\boldsymbol{x},i_1,i_2)=(x_{i_1},x_{i_1+1},\cdots,x_{i_2})$。

## 1.1　参数说明

本文方案用到的参数如表1所示：$\lambda$ 为安全参数，群容量 $N_{gs}=2^l\in\mathrm{poly}(\lambda)$。

**表1　方案中主要用到的几个参数**

| 参数 | 取值范围 |
|---|---|
| 格参数$n$ | $\mathcal{O}(\lambda)$ |
| 素数模$q$ | $\tilde{\mathcal{O}}(ln^3)$ |
| 维数$m$ | $2n\lceil\log q\rceil$ |
| 高斯参数 $\sigma$ | $\Omega\sqrt{n\log q}\log n$ |
| 无穷范数界限 $\beta$ | $\sigma\omega(\log m)$ |
| 无穷范数 $B$ | $\sqrt{n}\omega(\log n)$ 且 $(4B+1)^2\leqslant q$ |

## 1.2　困难问题

本文方案的安全性依赖于SIS和LWE假设。

**引理 1**[8]　对于实数$m$，$\beta=\mathrm{poly}(n)$，$q\geqslant\beta\cdot\omega(\sqrt{n\log n})$，$\mathrm{SIS}_{n,m,q,\beta}$ 问题描述如下：给定矩阵 $\boldsymbol{A}\leftarrow_R\mathbb{Z}_q^{n\times m}$，寻找向量 $\boldsymbol{x}\in\Lambda_q^{\perp}(\boldsymbol{A})$，使得 $0<\|\boldsymbol{x}\|\leqslant\beta$。求解 $\mathrm{SIS}_{n,m,q,\beta}$ 问题的困难性至少等价于求解 $\mathrm{SIVP}_{\gamma}$ 问题，其中 $\gamma=\beta\cdot\widetilde{\mathcal{O}}(\sqrt{n})$。

**引理 2**[11,12]　给定 $n,m\geqslant1$，$q\geqslant2$，以及 $\mathbb{Z}$ 上的一个概率分布 $\chi$。对于 $\boldsymbol{s}\in\mathbb{Z}_q^n$，通过采样向量 $\boldsymbol{a}\leftarrow_R\mathbb{Z}_q^n$ 和误差 $e\leftarrow\chi$，获得分布 $A_{\boldsymbol{s},\chi}$，同时输出 $(\boldsymbol{a},\boldsymbol{a}^{\mathrm{T}}\boldsymbol{s}+e)\in\mathbb{Z}_q^n\times\mathbb{Z}_q$。区分取自分布 $A_{\boldsymbol{s},\chi}$ 的 $m$ 个样本和取自 $\mathbb{Z}_q^n\times\mathbb{Z}_q$ 上的均匀分布的 $m$ 个样本就是 $\mathrm{LWE}_{n,q,\chi}$ 问题。对于某一 $\gamma=\widetilde{\mathcal{O}}(nq/\beta)$ 和 $\beta$ 界的分布 $\chi$，以及素数$q$，其中 $\beta\geqslant\sqrt{n}\omega(\log q)$，求解 $\mathrm{LWE}_{n,q,\chi}$ 问题等价于求解 $\mathrm{SIVP}_{\gamma}$ 问题。

## 1.3　相关算法

**引理 3**[13]　对实数 $\delta>0$，安全参数 $n$，奇素数 $q=\mathrm{poly}(n)$，及整数 $m=\mathcal{O}(n\log q)$，存在一个PPT算法TrapGen，输出 $(\boldsymbol{A},\boldsymbol{T}_{\boldsymbol{A}})$，其中 $\boldsymbol{T}_{\boldsymbol{A}}\subset\Lambda_q^{\perp}(\boldsymbol{A})$，使得 $\boldsymbol{A}\sim U(\mathbb{Z}_q^{n\times m})$(要求满足统计距离不大于 $2^{-\Omega(n)}$)，$\|\boldsymbol{T}_{\boldsymbol{A}}\|\leqslant\mathcal{O}(n\log q)$，且 $\|\tilde{\boldsymbol{T}}_{\boldsymbol{A}}\|\leqslant\mathcal{O}(\sqrt{n\log q})=\mathcal{O}(\sqrt{m})$。进一步，对任意 $\boldsymbol{u}\in\mathbb{Z}_q^n$ 和 $\sigma=\mathcal{O}(\sqrt{n\log q})$，存在一个PPT算法SampleD($\boldsymbol{T}_{\boldsymbol{A}},\boldsymbol{A},\boldsymbol{u},\sigma$)，依分布 $D_{\mathbb{Z}^m,\sigma}$ 采样一个向量 $\boldsymbol{x}\in\mathbb{Z}^m$，满足 $\boldsymbol{A}\cdot\boldsymbol{x}=\boldsymbol{u}\bmod q$。

**引理 4**[14]　存在一个PPT算法ExtRndBasis，输入 $\boldsymbol{A}'=[\boldsymbol{A}|\boldsymbol{B}]\in\mathbb{Z}_q^{n\times(m+k)}$，格 $\Lambda_q^{\perp}(\boldsymbol{A})$ 的基 $\boldsymbol{T}_{\boldsymbol{A}}\in\mathbb{Z}_q^{n\times m}$，以及实数 $\sigma\geqslant\|\tilde{\boldsymbol{T}}_{\boldsymbol{A}}\|\cdot\omega(\sqrt{\log m})$，输出格 $\Lambda_q^{\perp}(\boldsymbol{A}')$ 的基 $\boldsymbol{T}_{\boldsymbol{A}'}$，满足 $\|\boldsymbol{T}_{\boldsymbol{A}'}\|\leqslant s(m+m')$，且 $\|\tilde{\boldsymbol{T}}_{\boldsymbol{A}'}\|\leqslant s\sqrt{m+m'}$。

**引理 5**[15] 令 $m>n$，$\sigma>\|\tilde{\boldsymbol{T}}_{\boldsymbol{A}}\|\cdot\sqrt{m}\cdot\omega(\log m)$，$q>2$，存在PPT算法SampleRight ($\boldsymbol{A},\boldsymbol{B},\boldsymbol{R},\boldsymbol{T}_{\boldsymbol{B}},\boldsymbol{u},\sigma$)，该算法输入矩阵 $\boldsymbol{A},\boldsymbol{B}\in\mathbb{Z}_q^{n\times m}$，其中$\boldsymbol{B}$为列满秩矩阵；随机矩阵 $\boldsymbol{R}\in\{-1,1\}^{m\times m}$，格 $\Lambda_q^{\perp}(\boldsymbol{B})$ 的一个短基 $\boldsymbol{T}_{\boldsymbol{B}}$ 以及向量 $\boldsymbol{u}\in\mathbb{Z}_q^n$。记 $\boldsymbol{F}=(\boldsymbol{A}\,|\,\boldsymbol{AR}+\boldsymbol{B})$，输出向量 $\boldsymbol{e}\in\mathbb{Z}^{2m}$，且 $\boldsymbol{e}\sim D_{\Lambda_q^u(\boldsymbol{F}),\sigma}$。

## 1.4　分解扩展技术(decomposition-extension, DE)

首先，定义一些集合：

$\boldsymbol{B}_{2i}$：$\boldsymbol{B}_{2i}\in\{0,1\}^{2i}$，恰有 $i$ 个分量为 1。

$\boldsymbol{B}_{3i}$：$\boldsymbol{B}_{3i}\in\{-1,0,1\}^{3i}$，对 $j\in\{-1,0,1\}$，恰好有 $i$ 个分量等于 $j$。

$\beta_j$：$\beta_j=\left\lfloor\beta+2^{j+1}/2^j\right\rfloor$，$\forall j\in[1,\delta_\beta]$。则 $\sum_{j=1}^{\delta_\beta}\beta_j=\beta$。$\forall w\in[-\beta,\beta]$，$\exists w^{(1)},w^{(2)},\cdots,w^{(\delta_\beta)}\in\{-1,0,1\}$，使得 $\sum_{j=1}^{\delta_\beta}\delta_j\cdot w^{(j)}=w$。

$\mathcal{S}$：所有置换 $\pi$ 的集合。同时，记 $T_\pi$ 为置换函数。

对于 $\boldsymbol{w}\in\{0,1\}^m$，有：

**引理 6**[16]　存在一个PPT算法EleDE，当输入 $\boldsymbol{w}\in\{0,1\}^m$ 时，输出为 $\boldsymbol{w}^*\in\boldsymbol{B}_{2m}$。对于任意置换 $\pi\in\mathcal{S}$，有 $\boldsymbol{w}^*\in\boldsymbol{B}_{2m}\Leftrightarrow\pi(\boldsymbol{w}^*)\in\boldsymbol{B}_{2m}$。

对于 $\boldsymbol{w}\in[-\beta,\beta]^m$，有：

**引理 7**[16]　存在一个PPT算法VecDE，当输入

$\boldsymbol{w}\in[-\beta,\beta]^m$ 时，输出 $\boldsymbol{w}^*\in\boldsymbol{B}_{3m\delta_\beta}$。对任意置换 $\pi\in\mathcal{S}$，有 $\boldsymbol{w}^*\in\boldsymbol{B}_{3m\delta_\beta}\Leftrightarrow\pi(\boldsymbol{w}^*)\in\boldsymbol{B}_{3m\delta_\beta}$。

## 2 零知识证明协议

本文要构造一个协议，使验证者相信：

1) 签名者正确执行了签名算法：签名是用签名者私钥和公开信息正确运算所得。

2) 签名者的撤销标签隐藏于LWE函数：验证者可以验证签名者身份未被撤销。

Stern型协议：正整数 $D,L,q\geqslant 2$，VALID为 $\{-1,0,1\}^L$ 的子集。任取置换 $\pi\in\mathcal{S}$，对应置换函数 $T_\pi$，满足以下关系：

$$\boldsymbol{x}\in\mathrm{VALID}\Leftrightarrow T_\pi(\boldsymbol{x})\in\mathrm{VALID}\tag{1}$$

目标是要构造一个满足下面关系的统计零知识证明：

$$\boldsymbol{R}_{\mathcal{GSS}}=\{(\boldsymbol{P},\boldsymbol{v},\boldsymbol{x})\in\mathbb{Z}_q^{D\times L}\times\mathbb{Z}_q^{D}\times \mathrm{VALID}:\boldsymbol{P}\cdot\boldsymbol{x}=\boldsymbol{v}\bmod q\}\tag{2}$$

具体协议如下：

1) Commitment：选取随机数 $\rho_1,\rho_2,\rho_3$，置换 $\pi\leftarrow_R\mathcal{S}$，以及向量 $\boldsymbol{r}_1\leftarrow_R\mathbb{Z}_q^{3(n+14m)\delta_\beta}$ 和 $\boldsymbol{r}_2\leftarrow_R[-B,B]^{2m}$。令 $\boldsymbol{r}_{1,0}=$ Parse$(r_1,9m+1,11m)$，发送承诺 $\mathrm{CMT}=(C_1,C_2,C_3)$ 给验证者：

$C_1=\mathrm{COM}(\pi,\boldsymbol{P}\cdot\boldsymbol{r}_1,\boldsymbol{G}\cdot\boldsymbol{A}_1\cdot\boldsymbol{H}_{4n\times 2m}\cdot\boldsymbol{r}_{1,0}+\boldsymbol{r}_2;\rho_1)$，

$C_2=\mathrm{COM}(T_\pi(\boldsymbol{r}_1),T_\pi(\boldsymbol{r}_2);\rho_2)$，

$C_3=\mathrm{COM}(T_\pi(\boldsymbol{x}+\boldsymbol{r}_1),T_\pi(\boldsymbol{e}+\boldsymbol{r}_2);\rho_3)$。

2) Challenge：验证者发送挑战 $\mathrm{Ch}\leftarrow_R\{1,2,3\}$ 给证明者。

3) Response：根据挑战Ch，证明者做出以下回应：

$\mathrm{Ch}=1$：$\boldsymbol{t}_x=T_\pi(\boldsymbol{x})$，$\boldsymbol{t}_e=T_\pi(\boldsymbol{e})$，$\boldsymbol{t}_{r_1}=T_\pi(\boldsymbol{r}_1)$，$\boldsymbol{t}_{r_2}=T_\pi(\boldsymbol{r}_2)$，回应为 $\mathrm{RSP}=(\boldsymbol{t}_x,\boldsymbol{t}_e,\boldsymbol{t}_{r_1},\boldsymbol{t}_{r_2};\rho_2,\rho_3)$；

$\mathrm{Ch}=2$：令 $\pi_2=\pi$，$\boldsymbol{y}_1=\boldsymbol{x}+\boldsymbol{r}_1$，$\boldsymbol{y}_2=\boldsymbol{e}+\boldsymbol{r}_2$，则回应为 $\mathrm{RSP}=(\pi_2,\boldsymbol{y}_1,\boldsymbol{y}_1;\rho_1,\rho_1)$；

$\mathrm{Ch}=3$：令 $\pi_3=\pi$，$\boldsymbol{r}_{3,1}=\boldsymbol{r}_1$，$\boldsymbol{r}_{3,2}=\boldsymbol{r}_2$，则回应为 $\mathrm{RSP}=(\pi_3,\boldsymbol{r}_{3,1},\boldsymbol{r}_{3,2};\rho_1,\rho_2)$。

Verification：收到RSP，令 $\boldsymbol{y}_{1,0}=\mathrm{Parse}(\boldsymbol{y}_1,9m+1,11m)$，$\boldsymbol{r}_{3,0}=\mathrm{Parse}(\boldsymbol{r}_{3,1},m+1,2m)$，验证如下：

$\mathrm{Ch}=1$：验证 $\boldsymbol{t}_x\in\mathrm{VALID}$，$C_2=\mathrm{COM}(\boldsymbol{t}_{r_1},\boldsymbol{t}_{r_2};\rho_1,\rho_2)$，$C_3=\mathrm{COM}(\boldsymbol{t}_x+\boldsymbol{t}_{r_1},\boldsymbol{t}_e+\boldsymbol{t}_{r_2};\rho_3)$；

$\mathrm{Ch}=2$：验证 $C_1=\mathrm{COM}(\pi_2,\boldsymbol{P}\cdot\boldsymbol{y}_1-\boldsymbol{v},\boldsymbol{G}\cdot\boldsymbol{A}_1\cdot\boldsymbol{y}_{1,0}+\boldsymbol{y}_2-\boldsymbol{b};\rho_1)$，$C_3=\mathrm{COM}(T_{\pi_2}(\boldsymbol{y}_1),T_\pi(\boldsymbol{y}_2);\rho_3)$；

$\mathrm{Ch}=3$：验证 $C_1=\mathrm{COM}(\pi_3,\boldsymbol{P}\cdot\boldsymbol{r}_{3,1},\boldsymbol{G}\cdot\boldsymbol{A}_1\cdot\boldsymbol{r}_{3,0}+\boldsymbol{r}_{3,2};\rho_1)$，$C_2=\mathrm{COM}(T_{\pi_3}(\boldsymbol{r}_{3,1}),T_{\pi_3}(\boldsymbol{r}_{3,2});\rho_2)$。

如果每种情况下的验证都通过，输出Valid。

该协议调用COM承诺方案[17]，满足：

**引理 8**[17] 上述协议是关于 $\boldsymbol{P}\cdot\boldsymbol{x}=\boldsymbol{v}\bmod q$ 的零知识证明协议，具有完美的完整性，2/3的合理性误差，$\widetilde{\mathcal{O}}(L\log q)$ 的通信开销。且：

1) 存在一个有效的模拟器，在输入 $(\boldsymbol{P},\boldsymbol{v})$ 时，输出一个可接受的副本，该副本统计上接近于实际证明者所产生的副本。

2) 存在一个有效的知识提取器，在输入承诺CMT和分别对应值 $Ch=\{1,2,3\}$ 的有效 $(\mathrm{RSP}_1,\mathrm{RSP}_2,\mathrm{RSP}_3)$ 时，输出向量 $\boldsymbol{x}'\in\mathrm{VALID}$，满足 $\boldsymbol{P}\cdot\boldsymbol{x}'=\boldsymbol{v}\bmod q$。

## 3 基于格的动态群签名方案

### 3.1 本文的群签名方案模型

一个动态群签名方案应该有3个参与方[18]：群管理员 $\mathcal{GM}$、打开权威 $\mathcal{OA}$ 以及用户集合 $\mathcal{U}$。具体描述如下：

**定义 1**(动态群签名) 一个动态群签名方案由下面几个算法组成：

Setup$(1^\lambda,1^{N_{gs}})$：该算法由可信第三方完成。输入安全参数 $\lambda$ 和群最大容纳量 $N_{gs}\in\mathbb{N}$，输出公钥 $\mathcal{Y}$，$\mathcal{GM}$ 的私钥 $\mathcal{S}_{\mathcal{GM}}$ 以及 $\mathcal{OA}$ 的打开钥 $\mathcal{S}_{\mathcal{OA}}$，同时，初始化数据库状态St。St由三部分组成——当前群成员数据集合 $\mathrm{St}_{\mathrm{users}}$、群用户信息数据库 $\mathrm{St}_{\mathrm{trans}}=\bigcup<i,\mathrm{transcript}_i>$，及撤销用户集合RL。其初始化状态分别为 $\mathrm{St}_{\mathrm{users}}=\varnothing$，$\mathrm{St}_{\mathrm{trans}}=\epsilon$，$\mathrm{RL}=\varnothing$。

Join$(\mathcal{Y},\mathrm{St},\mathcal{S}_{\mathcal{GM}})$：该算法由 $\mathcal{GM}$ 和 $\mathcal{U}_i$ 交互完成。它包含两个交互图灵机 $\boldsymbol{J}_{\mathrm{user}}$ 和 $\boldsymbol{J}_{\mathcal{GM}}$，其输入均为 $\mathcal{Y}$。定义该协议为$[\boldsymbol{J}_{\mathrm{user}}(\lambda,\mathcal{Y})$，$\boldsymbol{J}_{\mathcal{GM}}(\lambda,\mathrm{St},\mathcal{Y},\mathcal{S}_{\mathcal{GM}})]$。正确执行协议后，$\mathcal{U}_i$ 将获得私钥 $\mathrm{sec}_i$ 和群身份证书 $\mathrm{cert}_i$，同时，$\mathcal{GM}$ 更新数据库 St：$\mathrm{St}_{\mathrm{users}}:=\mathrm{St}_{\mathrm{users}}\bigcup\{i\}$，$\mathrm{St}_{\mathrm{trans}}:=\mathrm{St}_{\mathrm{trans}}\|<i,\mathrm{transcript}_i>$。

Revoke$(\mathcal{GM},\mathcal{U}_i)$：该算法由群管理员或用户访问执行。如果要更新撤销列表RL，首先验证其身份，验证通过，更新撤销链表为 $\mathrm{RL}=\mathrm{RL}\bigcup\mathrm{grt}[i]$，同时，更新群成员数据库 $\mathbf{St}_{\mathrm{users}}=\mathbf{St}_{\mathrm{users}}/i$；验证不通过，输出⊥(其中⊥表示失败，终止协议)。

Sign$(\mathcal{Y},\mathrm{cert}_i,\mathrm{sec}_i,M)$：输入消息$M$，公钥 $\mathcal{Y}$，群成员证书 $\mathrm{cert}_i$，以及群私钥 $\mathrm{sec}_i$，输出签名 $\Sigma$。

Verify$(\mathcal{Y},\Sigma)$：输入群签名 $\Sigma$，消息$M$，和群公钥 $\mathcal{Y}$，该算法返回1或0。

Open($\mathcal{Y},\mathcal{S}_{\mathcal{OA}},M,\Sigma$)：打开权威运行该算法。输入消息$M$，公钥$\mathcal{Y}$，签名$\Sigma$，及打开钥$\mathcal{S}_{\mathcal{OA}}$，输出身份$i\in \mathrm{St}_{\mathrm{users}}\cup\perp$(其中$\perp$表示打开失败)。

对群签名方案的最基本要求是：一个由合法群成员通过诚实计算产生的签名一定能通过验证，且打开权威一定能成功进行追踪(正确性)。即，对于任意$(\mathcal{Y},\mathcal{S}_{\mathcal{OA}},\mathrm{sec})$，消息$M$，$i\in[N_{gs}]$，如果$\Sigma\leftarrow\mathrm{Sign}(\mathcal{Y},\mathrm{cert}_i,\mathrm{sec}_i,M)$，那么

$\mathrm{Verify}(\Sigma,M,\mathcal{Y})=1$且$\mathrm{Open}(\mathcal{Y},M,\Sigma)=i$。

另外，群签名方案还需要满足两个安全性要求：匿名性和可追踪性。下面用游戏的方式，给出这两个性质的具体说明。

1) 匿名性：在匿名性游戏中，敌手的目标是判定两个用户中的哪一个是真正产生签名的签名者。具体的游戏过程如下：

① Setup：挑战者运行算法$\mathrm{Setup}(1^{\lambda},1^{N_{gs}})$以产生$(\mathrm{St},\mathcal{Y},\mathcal{S}_{\mathcal{GM}},\mathcal{S}_{\mathcal{OA}})$，然后把$\mathcal{Y},\mathcal{S}_{\mathcal{GM}}$发送给敌手$\mathcal{A}$。

② 查询：敌手$\mathcal{A}$可以做以下的查询：

签名查询：敌手$\mathcal{A}$可以查询签名预言机，得到任意消息$M\in\{0,1\}^*$任意用户的签名。

腐败：$\mathcal{A}$可以查询任意用户的私钥，挑战者将被询问过的用户加入腐败集合$U_a$。

撤销：$\mathcal{A}$查询任意用户的撤销 token，挑战者将被询问过的用户加入集合$U_b$。

③ 挑战：敌手$\mathcal{A}$选择未查询过的消息$M^*$，并选择两个用户$d_0$和$d_1$，$d_i\notin U_a\cup U_b$，其私钥证书对分别为$(\mathrm{sec}_0^*,\mathrm{cert}_0^*)$和$(\mathrm{sec}_1^*,\mathrm{cert}_1^*)$，发送这些给挑战者。挑战者选择$b\leftarrow_R\{0,1\}$，并用$(\mathrm{sec}_b^*,\mathrm{cert}_b^*)$计算挑战签名$\Sigma^*$，将$\Sigma^*$发送给敌手$\mathcal{A}$。

④ 受限查询：这一阶段，敌手$\mathcal{A}$仍能做一些查询，但不允许对用户$d_0$和$d_1$做腐败和撤销查询。

⑤ 输出：$\mathcal{A}$输出$b^*$。如果$b^*=b$，敌手获胜。

敌手的优势可定义为：

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{anony}}=|\Pr[b^*=b]-1/2| \tag{3}$$

当$\mathrm{Adv}_{\mathcal{A}}$可忽略时，该群签名方案具有匿名性。

2) 可追踪性：在可追踪性游戏中，敌手的目标是伪造一个签名，利用追踪算法不能追踪到腐败集合中的用户。敌手能腐化群管理员和打开权威，也能在查询阶段腐化用户，并可利用加入算法产生一些虚拟成员。

① Setup：挑战者运行$\mathrm{Setup}(1^{\lambda},1^{N_{gs}})$算法产生$(\mathrm{St},\mathcal{Y},\mathcal{S}_{\mathcal{GM}},\mathcal{S}_{\mathcal{OA}})$，然后发送$(\mathrm{St},\mathcal{Y},\mathcal{S}_{\mathcal{GM}},\mathcal{S}_{\mathcal{OA}})$给敌手$\mathcal{A}$，并设置$U_a=\varnothing$。

② 查询：该阶段，敌手$\mathcal{A}$可查询以下预言机。

签名：$\mathcal{A}$输入任意消息和用户，签名预言机返回对应签名。将所有询问过的签名添加到集合 Sig 中(集合 Sig 是由$(i,M,\Sigma)$组成的集合，其中$i$指用户身份，$M$是消息，$\Sigma$是对应的签名)。

腐败：敌手$\mathcal{A}$询问任意用户的私钥时，挑战者将该用户加入集合$U_a$中，并返回其私钥。

③ 伪造：敌手$\mathcal{A}$利用其所拥有的信息产生一个消息签名对$(M^*,\Sigma^*)$和撤销链表$\mathrm{RL}^*$，使其满足：

Ⅰ $\mathrm{Verify}(\mathrm{RL}^*,\Sigma^*,M^*,\mathcal{Y})=1$；

Ⅱ 打开失败，或$i=\mathrm{Open}(M^*,\Sigma^*,\mathcal{S}_{\mathcal{OA}},\mathcal{Y},\mathrm{St}')$，且$i\in U_a/\mathrm{RL}^*$；

Ⅲ Sigs 表示用户$i$曾产生的签名，$\Sigma^*\notin\mathrm{Sigs}$；

则敌手赢得该游戏，返回1；否则，返回0。

当敌手的优势可忽略时，称该群签名方案具有可追踪性。

### 3.2 具体方案构造

Setup($1^{\lambda},1^{N_{gs}}$)：给定群安全参数$\lambda$以及群最大容量$N_{gs}$，参数$n,q,m,\sigma,\beta,B$。$\chi$为$B$界的分布。对于$t=\omega(\log n)$，随机预言机$H:\{0,1\}^*\to\{1,2,3\}^t$，$H_0:\{0,1\}^*\to\mathbb{Z}_q^{n\times 2m}$。依次运行以下步骤：

1) 运行算法获得$(\boldsymbol{A}_1,\boldsymbol{T}_{\boldsymbol{A}_1})\leftarrow\mathrm{TrapGen}(1^n,1^m,q)$，其中$\boldsymbol{A}_1\in\mathbb{Z}_q^{n\times m}$，$\boldsymbol{T}_{\boldsymbol{A}_1}$为格$\Lambda_q^{\perp}(\boldsymbol{A}_1)$的一个短基；

2) $\boldsymbol{A}_{2,1},\boldsymbol{A}_{2,2},\boldsymbol{D}\leftarrow_R\mathbb{Z}_q^{n\times m}$，$\boldsymbol{D}_0,\boldsymbol{D}_1\leftarrow_R\mathbb{Z}_q^{2n\times 2m}$，$\boldsymbol{u}\leftarrow_R\mathbb{Z}_q^n$；

3) $\boldsymbol{F}\leftarrow_R\mathbb{Z}_q^{4n\times 4m}$，该矩阵将被用来确保能抵抗指定攻击；

4) 生成满足 GPV-IBE 方案的公私钥对$(\boldsymbol{B},\boldsymbol{T}_{\boldsymbol{B}})$，其中$\boldsymbol{B}\in\mathbb{Z}_q^{n\times m}$，$\boldsymbol{T}_{\boldsymbol{B}}$为格$\Lambda_q^{\perp}(\boldsymbol{B})$的一个短基；

5) 选取具备强不可伪造性的一次签名方案$\Pi^{\mathrm{OTS}}=(\mathcal{G},\mathcal{S}_{\mathrm{sig}},\mathcal{V}_{\mathrm{ver}})$。

则群公钥为：

$$\mathcal{Y}=\{\boldsymbol{A}_1,\boldsymbol{A}_{2,1},\boldsymbol{A}_{2,2},\boldsymbol{B},\boldsymbol{D},\boldsymbol{D}_0,\boldsymbol{D}_1,\boldsymbol{F},\boldsymbol{u},\Pi^{\mathrm{OTS}},H,H_0\} \tag{4}$$

群管理员$\mathcal{GM}$的私钥为$\mathcal{S}_{\mathcal{GM}}=\boldsymbol{T}_{\boldsymbol{A}_1}$；打开权威$\mathcal{OA}$的密钥为$\mathcal{S}_{\mathcal{OA}}=\boldsymbol{T}_{\boldsymbol{B}}$。

设置初始的撤销列表为$\mathrm{RL}=\varnothing$，群成员集合为$\mathrm{St}_{\mathrm{users}}=\varnothing$，用户信息库为$\mathrm{St}_{\mathrm{trans}}=\epsilon$。其中，$\mathrm{St}_{\mathrm{trans}}$仅允许$\mathcal{GM}$和$\mathcal{OA}$访问；$\mathrm{St}_{\mathrm{users}}$允许任何人访问，但仅允许$\mathcal{GM}$做写入操作；任何人都可访问 RL，仅允许撤销图灵机进行写入操作。

Join($\mathcal{Y},\mathrm{St},\mathcal{S}_{\mathcal{GM}}$)：

1) $\mathcal{U}_i$

选取 $\boldsymbol{z}_i \leftarrow_R D_{\mathbb{Z}^{4m},\sigma}$，及 $\boldsymbol{s}_{i,2} \leftarrow_R D_{\mathbb{Z}^m,\sigma}$，并计算 $\boldsymbol{v}_i = \boldsymbol{F}\cdot\boldsymbol{z}_i \bmod q \in \mathbb{Z}_q^{4n}$；利用 PKI 系统中常用密钥签名 $\mathrm{sig}_i = \mathrm{Sign}_{\boldsymbol{usk}[i]}(\boldsymbol{v}_i)$。发送 $\boldsymbol{v}_i, \boldsymbol{s}_{i,2}$，及签名 $\mathrm{sig}_i$ 到 $\mathcal{GM}$；

2) $\mathcal{GM}$

验证 $\boldsymbol{v}_i$ 被注册与否，并验证签名 $\mathrm{sig}_i$ 的正确性。如果被注册过或签名不正确，终止协议；否则，$\mathcal{GM}$ 要做以下工作：

① 选取新身份 $i(0<i\leqslant N_{gs})$，定义

$$\overline{\boldsymbol{A}}_i = [\boldsymbol{A}_1 \,|\, \boldsymbol{A}_{2,1} + i\boldsymbol{A}_{2,2}] \tag{5}$$

计算 $\boldsymbol{u}_i = (\boldsymbol{A}_{2,1} + i\boldsymbol{A}_{2,2})\cdot\boldsymbol{s}_{i,2}$；

② 运行 $\boldsymbol{T}_i \leftarrow \mathrm{ExtRndBasis}(\overline{\boldsymbol{A}}_i, \boldsymbol{T}_{\boldsymbol{A}_1})$，$\boldsymbol{T}_i \in \mathbb{Z}^{2m\times 2m}$ 和 $\boldsymbol{s}_{i,1} \leftarrow \mathrm{SampleD}(\boldsymbol{A}_1, \boldsymbol{T}_{\boldsymbol{A}_1}, \boldsymbol{u}-\boldsymbol{u}_i, \sigma)$；

③ 定义 $\boldsymbol{s}_i = (\boldsymbol{s}_{i,1}, \boldsymbol{s}_{i,2}) \in \mathbb{Z}_q^{2m}$，计算其变色龙哈希函数 $\boldsymbol{c}_s = \boldsymbol{D}_0\cdot\mathrm{bin}(\boldsymbol{v}_i) + \boldsymbol{D}_1\cdot\boldsymbol{s}_i$；

④ 定义 $\boldsymbol{u}_s = \boldsymbol{u} + \boldsymbol{D}\cdot\mathrm{bin}(\boldsymbol{c}_s)$，采样获得 $\boldsymbol{d}_i \leftarrow \mathrm{SampleD}(\boldsymbol{T}_i, \overline{\boldsymbol{A}}_i, \boldsymbol{u}_s, \sigma)$；

⑤ 发送 $(i, \boldsymbol{d}_i, \boldsymbol{s}_i)$ 给用户 $\mathcal{U}_i$；

⑥ 定义用户 $\mathcal{U}_i$ 的撤销标签

$$\mathbf{grt}[i] = \boldsymbol{A}_1\cdot\boldsymbol{v}_i \bmod q \in \mathbb{Z}_q^n \tag{6}$$

令 $\mathrm{St}_{\mathrm{users}} := \mathrm{St}_{\mathrm{users}} \cup \{i\}$，将副本信息 $\mathrm{transcript}_i = (\boldsymbol{v}_i, \mathrm{cert}_i, \mathrm{upk}[i], \mathrm{sig}_i, \mathbf{grt}[i])$ 存储到 $\mathrm{St}_{\mathrm{trans}}$ 中。

3) $\mathcal{U}_i$

验证接收信息是否满足：$\overline{\boldsymbol{A}}_i\cdot\boldsymbol{d}_i = \boldsymbol{u} + \boldsymbol{D}\cdot\mathrm{bin}(\boldsymbol{D}_0\cdot\mathrm{bin}(\boldsymbol{v}_i) + \boldsymbol{D}_1\cdot\boldsymbol{s}_i) \bmod q$，$\|\boldsymbol{d}_i\|_\infty \leqslant \beta$，以及 $\|\boldsymbol{s}_i\|_\infty \leqslant \beta$。如果不满足，终止协议；满足，则定义其成员身份证书为 $\mathrm{cert}_i = (i, \boldsymbol{d}_i, \boldsymbol{s}_i)$，私钥为 $\mathrm{sec}_i = \boldsymbol{z}_i$。

Revoke($\mathcal{GM}, \mathcal{U}_i$)：

撤销算法相当于一个可信第三方 $\mathcal{W}$，可根据访问者身份不同分两种情况——群管理员访问算法和群成员访问算法。要撤销群成员 $\mathcal{U}_i$ 的合法群身份，具体过程如下。

1) $\mathcal{GM}$ 访问时

直接发送 $\mathbf{grt}[i]$ 到撤销第三方 $\mathcal{W}$。假设 $\mathcal{W}$ 能分辨管理员身份(如果不能，$\mathcal{GM}$ 与 $\mathcal{W}$ 交互通信来验证 $\mathcal{GM}$ 身份)，更新撤销列表 $\mathrm{RL} := \mathrm{RL}\cup\{\mathbf{grt}[i]\}$。

2) $\mathcal{U}_i$ 访问时

发送证书 $\mathrm{cert}_i$ 和 $(\boldsymbol{v}_i, \mathrm{sig}_i)$ 到撤销第三方 $\mathcal{W}$。$\mathcal{W}$ 接到 $\mathcal{U}_i$ 的请求后，将收到信息转发给 $\mathcal{GM}$，由 $\mathcal{GM}$ 验证其信息正确性。验证通过，将 $\mathbf{grt}[i]$ 返回给 $\mathcal{W}$，$\mathcal{W}$ 更新撤销列表 $\mathrm{RL} := \mathrm{RL}\cup\{\mathbf{grt}[i]\}$。

撤销完成后，$\mathcal{GM}$ 更新 $\mathrm{St}_{\mathrm{users}} := \mathrm{St}_{\mathrm{users}}/\{i\}$。

Sign($\mathcal{Y}, \mathrm{cert}_i, \mathrm{sec}_i, M$)：

对于消息 $M\in\{0,1\}^*$，群成员 $\mathcal{U}_i$ 要对其做群签名，具体操作如下。

1) 选取一次签名密钥对 $(\mathrm{svk}, \mathrm{ssk}) \leftarrow \mathcal{G}(n)$，并记 $\boldsymbol{d}_i = [\boldsymbol{d}_{i,1} / \boldsymbol{d}_{i,2}] \in \mathbb{Z}^{2m}$；

2) 令 $\boldsymbol{G} = H_0(\mathrm{svk}) \in \mathbb{Z}_q^{n\times 2m}$，选取 $\boldsymbol{e} \leftarrow_R \chi^{2m}$，$\boldsymbol{e}_0 \leftarrow_R \chi^n$，$\boldsymbol{x}_1 \leftarrow_R \chi^m$，$\boldsymbol{x}_2 \leftarrow_R \chi^{2m}$，得：

$$\boldsymbol{b} = \boldsymbol{G}^{\mathrm{T}}\cdot\mathbf{grt}[i] + \boldsymbol{e} \bmod q \tag{7}$$

$$\boldsymbol{c}_1 = \boldsymbol{B}^{\mathrm{T}}\cdot\boldsymbol{e}_0 + \boldsymbol{x}_1 \bmod q \tag{8}$$

$$c_2 = \boldsymbol{G}^{\mathrm{T}}\cdot e_0 + x_2 + \mathrm{bin}(v_i)\cdot\lfloor q/2\rfloor \bmod q \tag{9}$$

3) 产生一个零知识证明(简记为ZKAoK)协议(具体见下节)。该协议可以用来证明：① 签名者是拥有合法群身份的成员；② $\boldsymbol{b}$ 是诚实计算所得。并行执行 $t=\omega(\log n)$ 次该协议，使得合理性误差足够小。然后利用Fiat-Shamir启发式将其转化为非交互ZKAoK协议：

$$\tau = \{\{\mathrm{CMT}^{(k)}\}_{k=1}^t, \mathrm{CH}, \{\mathrm{RSP}^{(k)}\}_{k=1}^t\} \tag{10}$$

$$\mathrm{CH} = H(M, \mathrm{svk}, \boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{b}, \{\mathrm{CMT}^{(k)}\}_{k=1}^t) \in \{1,2,3\}^t \tag{11}$$

4) 定义 $\Sigma_1 = (\boldsymbol{b}, \boldsymbol{c}_1, \boldsymbol{c}_2, \tau)$，计算 $\Sigma_2 = \mathcal{S}_{\mathrm{sig}}(\mathrm{ssk}, \Sigma_1)$；

5) 输出群签名

$$\Sigma = (M, \mathrm{svk}, \Sigma_1, \Sigma_2) \tag{12}$$

Verify($Y, M, \Sigma$)：

按式(12)分解签名 $\Sigma$，依次执行下列步骤：

1) 计算 $\boldsymbol{G} = H_0(\mathrm{svk})$；

2) 验证 $\mathcal{V}_{\mathrm{ver}}(M, \mathrm{svk}, \mathrm{RL}, \boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{b}, \tau, \Sigma_2) = 1$，如果通过，执行下一步；

3) 验证 ZKAoK 协议 $\tau$ 是否成立，若成立，则进入下一步；

4) 遍历撤销列表 RL，$\forall\mathbf{grt}[i]\in\mathrm{RL}$，计算 $\boldsymbol{e}_i' = \boldsymbol{b} - \boldsymbol{G}^{\mathrm{T}}\cdot\mathbf{grt}[i] \bmod q$。验证是否存在 $i$，使得 $\|e_i'\|_\infty \leqslant B$。

如果中间不终止，且最后不存在 $i$，则验证通过，返回 1；否则，返回 0。

Open($\mathcal{Y}, \mathcal{S}_{\mathcal{OA}}, M, \Sigma$)：

打开权威 $\mathcal{OA}$ 可利用其私钥 $\mathcal{S}_{\mathcal{OA}} = \boldsymbol{T}_{\boldsymbol{B}}$ 打开签名获得真实签名者身份，步骤如下：

1) 计算 $\boldsymbol{G} = H_0(\mathrm{svk})$，运行 $2m$ 次 SampleD 算法获得小范数矩阵 $\boldsymbol{E}_{\mathrm{svk}} \in \mathbb{Z}^{m\times 2m}$，使得其满足 $\boldsymbol{B}\cdot\boldsymbol{E}_{\mathrm{svk}} = \boldsymbol{G} \bmod q$；

2) 利用 $\boldsymbol{E}_{\mathrm{svk}}$ 解密部分密文 $\boldsymbol{c}_1$ 和 $\boldsymbol{c}_2$，解密结果为 $\mathrm{bin}(\boldsymbol{v}) = \lfloor(\boldsymbol{c}_2 - \boldsymbol{E}_{\mathrm{svk}}^{\mathrm{T}}\cdot\boldsymbol{c}_2)/(q/2)\rfloor$；

3) 计算 $\boldsymbol{v}=\boldsymbol{H}_{4n\times 2m}\cdot \mathrm{bin}(\boldsymbol{v}) \bmod q$，对照数据库 $\mathrm{St}_{\mathrm{trans}}$，找出与 $v$ 一致的用户身份 $i$，输出 $i$。若查找失败，输出 $\perp$。

## 4 安全性及效率分析

### 4.1 安全性分析

**引理 9**[6] 令 $B=\sqrt{n}\omega(\log n)$，$q\geqslant(4B+1)^2$，$m=2n\lceil\log q\rceil$，那么，在 $\boldsymbol{G}\in\mathbb{Z}_q^{n\times 2m}$ 的随机性基础上，有：

$$\Pr\left[\exists\text{非零的}\boldsymbol{g}\in\mathbb{Z}_q^n:\left\|\boldsymbol{G}^{\mathrm{T}}\cdot\boldsymbol{g}\right\|_\infty\leqslant 2B\right]\leqslant \mathrm{negl}(\lambda)。$$

**定理 1**(正确性)极大概率本文的方案是正确的。

**证明**：正确性定义见文献[7]。很容易验证，该方案满足定义中所提出条件。在此，本文只需另外证明关于向量 $\boldsymbol{b}$ 的部分。

对每个 $\mathbf{grt}[j]\in$ RL，计算 $\boldsymbol{e}'_j=\boldsymbol{b}-\boldsymbol{G}^{\mathrm{T}}\cdot\mathbf{grt}[j]=\boldsymbol{G}^{\mathrm{T}}\cdot(\mathbf{grt}[i]-\mathbf{grt}[j])+\boldsymbol{e} \bmod q$。如果存在指标 $i$ 使得 $\mathbf{grt}[i]=\mathbf{grt}[j]$，这时有 $\left\|\boldsymbol{e}'_i\right\|_\infty=\left\|\boldsymbol{e}\right\|_\infty\leqslant B$，验证不通过，矛盾。如果 $\mathbf{grt}[i]\notin$ RL，那么对任意 $i$，向量 $\boldsymbol{g}=\mathbf{grt}[i]-\mathbf{grt}[j]$ 非零。由引理9知，以很大概率有 $\left\|\boldsymbol{G}^{\mathrm{T}}\cdot\boldsymbol{g}\right\|_\infty\leqslant 2B$。另一方面，$\left\|\boldsymbol{G}^{\mathrm{T}}\cdot\boldsymbol{g}\right\|_\infty\leqslant\left\|\boldsymbol{e}'_j\right\|_\infty+\left\|\boldsymbol{e}\right\|_\infty\leqslant\left\|\boldsymbol{e}'_j\right\|_\infty+B$，故 $\left\|\boldsymbol{e}'_j\right\|_\infty\geqslant 2B-B=B$ 有很大概率是成立的，验证通过。

**定理 2**(匿名性)随机预言机模型下，基于 $\mathrm{LWE}_{n,q,\chi}$ 假设，如果 $\Pi^{\mathrm{OTS}}$ 为强不可伪造一次签名，那么本文的方案具有匿名性。

**证明**：利用一系列游戏来证明匿名性。

Game $G_0^{(b)}$：

1) 运行算法 $\mathrm{Setup}(1^\lambda,1^{N_{gs}})$ 产生 $(\mathrm{St},\mathcal{Y},\mathcal{S}_{\mathcal{GM}},\mathcal{S}_{\mathcal{OA}})$，然后将 $\mathcal{Y},\mathcal{S}_{\mathcal{GM}}$ 发送给 $\mathcal{A}$。初始化撤销链表 $RL=\varnothing$，腐败集合 $U_a=\varnothing$，撤销查询集合 $U_b=\varnothing$；

2) 在查询阶段，$\mathcal{A}$ 不仅可以查询任意用户对任意消息$M$的签名，还能更新 $U_a$ 和 $U_b$；

3) $\mathcal{A}$ 输出选定消息 $M^*$ 和两个指标 $d_0,d_1\notin U_a\bigcup U_b$，满足 $\mathbf{grt}[d_0]$，$\mathbf{grt}[d_1]\notin$ RL；

4) 选择 $b\leftarrow_R\{0,1\}$，产生一个合法签名 $\Sigma_0=(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}^*,\pi^*,\Sigma_2^*)\leftarrow\mathrm{Sign}(\mathcal{Y},\mathrm{cert}_b^*,\mathrm{sec}_b^*,M^*)$，返回给敌手 $\mathcal{A}$；

5) $\mathcal{A}$ 仍能做一些查询，但不允许查询 $d_0$ 和 $d_1$ 的私钥和撤销令牌；

6) $\mathcal{A}$ 返回 $b'$。

Game $G_1^{(b)}$：

此游戏通过修改Game $G_0$ 获得：在步骤4)中，利用模拟器来模拟产生 $\pi$。输出签名 $\Sigma_1=(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}^*,\pi^*,\Sigma_2^*)$。

Game $G_{2,0}^{(b)}$：

此游戏在Game $G_1^{(b)}$ 上修改获得：选取 $\boldsymbol{g}_c\leftarrow_R\mathbb{Z}_q^n$，计算 $\boldsymbol{b}=\boldsymbol{G}^{\mathrm{T}}\cdot\boldsymbol{g}_c+\boldsymbol{e} \bmod q$。输出签名 $\Sigma_{2,0}=(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b},\pi^*,\Sigma_2^*)$。

Game $G_{2,1}^{(b)}$：

对Game $G_{2,0}$ 作修改：随机选择 $\boldsymbol{b}'\leftarrow_R\mathbb{Z}_q^{2m}$，输出签名 $\Sigma_{2,1}=(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}',\pi^*,\Sigma_2^*)$。

Game $G_{3,0}^{(b)}$：

对 Game $G_{2,1}^{(\mathrm{b})}$ 作以下修改：选择 $\boldsymbol{y}_1\leftarrow_R\mathbb{Z}_q^m$，$\boldsymbol{y}_2\leftarrow_R\mathbb{Z}_q^{2m}$，计算 $\boldsymbol{c}_1=\boldsymbol{y}_1$；$\boldsymbol{c}_2=\boldsymbol{y}_2+\mathrm{bin}(\boldsymbol{v}_{d_b}^*)\cdot\lfloor q/2\rfloor \bmod q$。则 $\Sigma_{3,0}=(M^*,\mathrm{svk}^*,\boldsymbol{c}_1,\boldsymbol{c}_2,\boldsymbol{b}',\pi,\Sigma_2^*)$。

Game $G_{3,1}$：

这一游戏在Game $G_{3,0}^{(b)}$ 基础上修改获得：随机选择 $\boldsymbol{c}'_1\leftarrow_R\mathbb{Z}_q^m$，$\boldsymbol{c}'_2\leftarrow_R\mathbb{Z}_q^{2m}$。输出签名 $\Sigma_{3,1}=(M^*,\mathrm{svk}^*,\boldsymbol{c}'_1,\boldsymbol{c}'_2,\boldsymbol{b}',\pi,\Sigma_2^*)$。

根据引理8和LWE假设，故有：$G_0^{(0)}\overset{s}{\approx}G_1^{(0)}\overset{s}{\approx}G_{2,0}^{(0)}\overset{c}{\approx}G_{2,1}^{(0)}\overset{c}{\approx}G_{3,0}^{(0)}\approx G_{3,1}\approx G_{3,0}^{(1)}\overset{c}{\approx}G_{2,1}^{(0)}\overset{c}{\approx}G_{2,0}^{(0)}\overset{s}{\approx}G_1^{(0)}\overset{s}{\approx}G_0^{(0)}$。($\overset{s}{\approx}$ 表示统计上不可区分；$\overset{c}{\approx}$ 表示计算上不可区分)

而Game $G_{3,1}$ 不依赖于挑战者的选择，所以匿名性得证。

**定理 3** (可追踪性)随机预言机模型下，在SIS假设下，本文的方案具有可追踪性。

**证明**：假设存在一个PPT敌手 $\mathcal{A}$，能以不可忽略的概率 $\varepsilon$ 来伪造一个签名 $\Sigma^*$，为攻击方案的可追踪性，构造一个PPT算法 $\mathcal{F}$，本文将证明 $\mathcal{F}$ 利用 $\mathcal{A}$ 能攻破SIS假设。具体来说，给算法 $\mathcal{F}$ 一个输入 $\hat{\boldsymbol{A}}\in\mathbb{Z}_q^{n\times m}$，找到一个 $\boldsymbol{w}\in\Lambda_q^\perp(\hat{\boldsymbol{A}})$，$0<\|\boldsymbol{w}\|\leqslant\beta'$。

1) Setup：$\mathcal{F}$ 获得 $(\boldsymbol{A}_{2,2},\boldsymbol{T}_{\boldsymbol{A}_{2,2}})\leftarrow\mathrm{TrapGen}(1^n,1^m,q)$，选取 $i^*\in[N_{gs}]$，$\boldsymbol{s}_{i^*,1}\leftarrow_R D_{\mathbb{Z}^m,\sigma}$，$\boldsymbol{s}_{i^*,2}\leftarrow_R D_{\mathbb{Z}^m,\sigma}$。令 $\boldsymbol{A}_1=\hat{\boldsymbol{A}}$，$\boldsymbol{A}_{2,1}=\boldsymbol{A}_1\boldsymbol{R}-i^*\boldsymbol{A}_{2,2}$，$\boldsymbol{u}^*=\boldsymbol{A}_1\boldsymbol{s}_{i^*,1}+\boldsymbol{A}_1\boldsymbol{R}\boldsymbol{s}_{i^*,2}$，令 $\boldsymbol{u}^*=\boldsymbol{u}$。记 $\boldsymbol{s}_{i^*}=(\boldsymbol{s}_{i^*,1},\boldsymbol{s}_{i^*,2})$，其他参数不变。设置 $\mathrm{RL}=\varnothing$，腐败集合 $U_a=\varnothing$。

2) Join：对 $i\in[N_{gs}]$，当 $i\neq i^*$ 时，选择 $\boldsymbol{z}_i\leftarrow_R D_{\mathbb{Z}^{4m},\sigma}$，$\boldsymbol{s}_{i,2}\leftarrow_R D_{\mathbb{Z}^m,\sigma}$，则 $\boldsymbol{v}_i=\boldsymbol{F}\cdot\boldsymbol{z}_i\in\mathbb{Z}_q^{4n}$。发

送 $\boldsymbol{v}_i$，$\boldsymbol{s}_{i,2}$，$\mathrm{sig}_i=\mathrm{Sign}_{\mathrm{usk}[i]}(\boldsymbol{v}_i)$ 到 $\mathcal{F}$。$\mathcal{F}$ 验证获得 $\boldsymbol{u}_i=[\boldsymbol{A}_1\boldsymbol{R}+(i-i^*)\boldsymbol{A}_{2,2}]\cdot\boldsymbol{s}_{i,2}$，$(\boldsymbol{s}_{i,1},\boldsymbol{s}'_{i,2})\leftarrow\mathrm{SampleRight}(\boldsymbol{A}_1,\boldsymbol{A}_{2,2},\boldsymbol{R},\boldsymbol{T}_{\boldsymbol{A}_{2,2}},\boldsymbol{u}-\boldsymbol{u}_i,\sigma)$。定义 $\boldsymbol{s}_i=(\boldsymbol{s}_{i,1},\boldsymbol{s}_{i,2}+\boldsymbol{s}'_{i,2})$，则 $\overline{\boldsymbol{A}}_i=[\boldsymbol{A}_1\,|\,\boldsymbol{A}\boldsymbol{R}+(i-i^*)\boldsymbol{A}_{2,2}]$，$\overline{\boldsymbol{A}}_i\cdot\boldsymbol{s}_i=\boldsymbol{u}$。对 $i\in[N_{gs}]$，令 $\boldsymbol{c}_s=\ \boldsymbol{D}_0\cdot\mathrm{bin}(\boldsymbol{v}_i)+\boldsymbol{D}_1\cdot\boldsymbol{s}_i$，$\boldsymbol{u}_s=\boldsymbol{u}+\boldsymbol{D}\cdot\mathrm{bin}(\boldsymbol{c}_s)$，$\mathcal{F}$ 获得 $\boldsymbol{d}_i=\left[\boldsymbol{d}_{i,1},\boldsymbol{d}_{i,2}\right]^{\mathrm{T}}\leftarrow\mathrm{SampleRight}(\boldsymbol{A}_1,\boldsymbol{A}_{2,2},\boldsymbol{R},\boldsymbol{T}_{\boldsymbol{A}_{2,2}},\boldsymbol{u}_s,\sigma)$，发送 $\mathrm{cert}_i=(i,\boldsymbol{d}_i,\boldsymbol{s}_i)$ 给用户 $i$。$\mathbf{grt}[i]$ 产生方式不变。

3) 查询：敌手 $\mathcal{A}$ 做下面的查询：

① 腐败：询问任意用户 $i\in\{1,2,\cdots,N_{gs}\}$ 的私钥，$\mathcal{F}$ 将 $i$ 加入到集合 $U_a$ 中，返回 $\boldsymbol{z}_i$。

② 签名：查询用户 $i$ 关于消息 $M$ 的签名，$\mathcal{F}$ 返回 $\Sigma$，并将 $(i,M,\Sigma)$ 加入到集合Sigs中。

4) 伪造：假如敌手 $\mathcal{A}$ 能以优势 $\varepsilon$ 成功伪造签名 $\Sigma^*=(M^*,\mathrm{svk}^*,\Sigma_1^*,\Sigma_2^*)$，则利用打开算法，可追踪到诚实用户或追踪失败。

对 $\Sigma^*$，$\pi^*=\{\{\mathrm{CMT}^{*(k)}\}_{k=1}^t,\mathrm{CH}^*,\{\mathrm{RSP}^{*(k)}\}_{k=1}^t\}$，$\mathcal{A}$ 必须调用预言机 $H$（且输入为 $(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}^*,\{\mathrm{CMT}^{*(k)}\}_{k=1}^t)$），否则，等式 $H(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}^*,\{\mathrm{CMT}^{*(k)}\}_{k=1}^t)=\mathrm{CH}^*$ 成立的概率至多为 $3^{-t}$。记 $Q_H$ 为查询 $H$ 的上限，则 $\mathcal{A}$ 的查询结果与第 $\kappa^*$ 次查询重合概率至少为 $\varepsilon'=\varepsilon-3^{-t}$，$\kappa^*\leqslant Q_H$。$\mathcal{F}$ 调用 $32\cdot Q_H/(\varepsilon-3^{-t})$ 次 $\mathcal{A}$ 算法。这些查询中，前 $\kappa^*-1$ 查询保持输入和随机预言机不变，则挑战值 $\mathrm{CH}_1,\mathrm{CH}_2,\cdots,\mathrm{CH}_{\kappa^*-1}$ 相同。但从第 $\kappa^*$ 次查询开始，挑战值 $\mathrm{CH}_{\kappa^*},\mathrm{CH}_{\kappa^*+1},\cdots,\mathrm{CH}_{Q_H}$ 开始不同。改进的 Forking 引理[16]保证了，对于 $(M^*,\mathrm{svk}^*,\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}^*,\{\mathrm{CMT}^{*(k)}\}_{k=1}^t)$，$\mathcal{F}$ 能以大于 1/2 的概率获得 $\mathrm{CH}_{\kappa^*}^{(1)},\mathrm{CH}_{\kappa^*}^{(2)},\mathrm{CH}_{\kappa^*}^{(3)}\in\{1,2,3\}^t$。以 $1-(7/9)^t$ 的概率，存在 $j\in\ \{1,2,\cdots,t\}$，使得 $\mathrm{CH}_{\kappa^*}^{(1)},\mathrm{CH}_{\kappa^*}^{(2)},\mathrm{CH}_{\kappa^*}^{(3)}$ 的第 $j$ 位是 $(\mathrm{CH}_{\kappa^*}^{(1)},\mathrm{CH}_{\kappa^*}^{(2)},\mathrm{CH}_{\kappa^*}^{(3)})=(1,2,3)$。从相对应的回应 $(\mathrm{RSP}_{(j)}^{*\,(1)},\mathrm{RSP}_{(j)}^{*\,(2)},\ \mathrm{RSP}_{(j)}^{*\,(3)})$，$\mathcal{F}$ 能提取 $\boldsymbol{s}^*=(\boldsymbol{s}_{i,1}^*,\boldsymbol{s}_{i,2}^*)\in\mathbb{Z}^{2m}$，$\boldsymbol{v}'\in\mathbb{Z}^{4m}$，$\boldsymbol{e}^*\in\mathbb{Z}^n$（引理8）。

考虑以下情况：

1) $i\neq i^*$，发生概率至多为 $\dfrac{N_{gs}-1}{N_{gs}}$。此时，$\mathcal{F}$ 输出“Fail”并终止；

2) $i=i^*$ 时，$\boldsymbol{A}_1\boldsymbol{s}_{i^*,1}+\boldsymbol{A}_1\boldsymbol{R}\boldsymbol{s}_{i^*,2}=\boldsymbol{A}_1\boldsymbol{s}_{i,1}^*+\boldsymbol{A}_1\boldsymbol{R}\boldsymbol{s}_{i,2}^*\bmod q$，且 $\boldsymbol{A}_1[(\boldsymbol{s}_{i^*,1}-\boldsymbol{s}_{i,1}^*)+\boldsymbol{R}(\boldsymbol{s}_{i^*,1}-\boldsymbol{s}_{i,1}^*)]=0\bmod q$。

下面证明以很大概率有 $(\boldsymbol{s}_{i^*,1}-\boldsymbol{s}_{i,1}^*)+\boldsymbol{R}(\boldsymbol{s}_{i^*,1}-\boldsymbol{s}_{i,1}^*)\neq 0\bmod q$。

1) 如果追踪失败，即 $\mathcal{V}(M^*,\mathrm{svk},\mathrm{grt}[i^*],\boldsymbol{c}_1^*,\boldsymbol{c}_2^*,\boldsymbol{b}^*,\pi,\Sigma_2)=1$。由签名的正确性可知 $\boldsymbol{A}_1\cdot\boldsymbol{v}'=\boldsymbol{A}_1\cdot\boldsymbol{v}_{i^*}\neq\mathrm{grt}[i^*]$，则 $\boldsymbol{v}'\neq\boldsymbol{v}_{i^*}$，进而有 $\boldsymbol{z}_i^*\neq\boldsymbol{z}_{i^*}$。由于 $\mathcal{Y}$ 给定时，cert 与 sec 一一对应，可知 $\boldsymbol{s}_i^*\neq\boldsymbol{s}_{i^*}$。

2) 如果追踪到用户 $k\notin U_a$ / RL。考虑以下两种情况：

① 如果 $\mathcal{A}$ 从未查询过 $\boldsymbol{z}_{i^*}$，那么 $\boldsymbol{v}_{i^*}$ 对 $\mathcal{A}$ 来说是未知的。这时，很大概率有 $\boldsymbol{v}'\neq\boldsymbol{v}_{i^*}$。

② 如果 $\mathcal{A}$ 查询过 $\boldsymbol{z}_{i^*}$，那么 $i^*\in U_a$。此时有 $k\neq i^*$，$\mathbf{grt}[k]\neq\mathbf{grt}[i^*]$，从而有 $\boldsymbol{v}'\neq\boldsymbol{v}_{i^*}$。

记 $\boldsymbol{x}=(\boldsymbol{s}_{i^*,1}-\boldsymbol{s}_{i,1}^*)+\boldsymbol{R}(\boldsymbol{s}_{i^*,1}-\boldsymbol{s}_{i,1}^*)\in\mathbb{Z}^m$，因此 $\boldsymbol{x}$ 为 $\boldsymbol{A}_1\bullet\boldsymbol{x}=0\bmod q$ 的一个非零解。从而解决了SIS问题。由引理1知，敌手成功伪造签名的优势是可忽略的。

### 4.2 效率分析

文献[3]是格基群签名的经典方案，文献[7]是第一个实现了群成员动态加入的格基群签名方案，文献[9]是第一个动态格基群签名方案。本文选取文献[3, 7, 9]进行储存空间对比和计算开销对比，结果如表2所示。其中，$N$ 表示群成员个数，$S$ 表示哈希运算的时间，$W$ 表示零知识证明的运算时间，$V$ 表示一次签名运算时间，$T_1$ 表示一次随机采样算法的时间，$T_2$ 表示一次数乘运算的时间，$T_3$ 表示运行一次特殊采样算法的时间(不同采样算法的运行时间是不同的，在这里不考虑这点)。

**表2 方案效率对比**

| 方案 | 公钥大小 | 私钥大小 | 签名尺寸 | 签名开销 | 有无撤销 | 撤销时更新 |
|---|---|---|---|---|---|---|
| 文献[3] | $\mathcal{O}(nmN\log q)$ | $\mathcal{O}(nm\log q)$ | $\mathcal{O}(nmN\log q)$ | $NS+(mq+2)T_1+W+nmNT_2+T_3$ | × | / |
| 文献[7] | $\tilde{\mathcal{O}}(nm\log N\log q)$ | $\tilde{\mathcal{O}}(m)$ | $\tilde{\mathcal{O}}(m\log q)$ | $S+3T_1+3nmT_2+V+W$ | × | / |
| 文献[9] | $\mathcal{O}(nm\log q)$ | $\mathcal{O}(m+n\log q+\log N)$ | $\mathcal{O}(n\log N)$ | $2(nm+n\log N_{gs})T_2+W$ | √ | 群和哈希树 |
| 本文方案 | $\tilde{\mathcal{O}}(nm\log q)$ | $\tilde{\mathcal{O}}(m)$ | $\tilde{\mathcal{O}}(m\log q)$ | $S+4T_1+5nmT_2+V+W$ | √ | 撤销列表 |

数的加法运算时间忽略不计。从表2可以看出，与文献[7]相比，本文的方案牺牲了一点计算开销：一个矩阵乘运算和一次随机采样，但实现了群成员的撤销；与方案[9]相比，本文方案的私钥尺寸较小，且撤销方式简单，当群成员加入和撤销时，需要更新的数量较小，更适用于成员变化较频繁的群。

## 5 结 束 语

本文给出了一个动态格基群签名方案，并在随机预言机模型下，基于SIS和LWE假设证明了方案的安全性。与已有的方案相比，本文的方案允许群成员在任何时间加入和退出群，而且公钥尺寸较小。另一方面，构造依赖于LWE加密方案，而且在随机预言机模型下是CPA-匿名的。构造一个更加简单高效，且在标准模型下具有CCA-匿名性的格基群签名方案是我们下一步的工作。

## 参 考 文 献


[1] CHAUM D, HEYST E V. Group signatures[C]// Theory and Application of Cryptographic Techniques. Brighton: Springer, 1991, 547: 257-265.

[2] BRESSON E, STERN J. Efficient revocation in group signatures[C]//International Workshop on Practice and Theory in Public Key Cryptography: Public Key Cryptography. Cheju Island: Springer, 2001: 190-206.

[3] GORDON S D, KATZ J, VAIKUNTANATHAN V. A group signature scheme from lattice assumptions[C]// International Conference on the Theory and Application of Cryptology and Information Security. Singapore: Springer, 2010, 2011: 395-412.

[4] LAGUILLAUMIE F, LANGLOIS A, LIBERT B, et al. Lattice-based group signatures with logarithmic signature size[C]//International Conference on the Theory and Application of Cryptology and Information Security. Gordon: Springer, 2013, 8270: 41-61.

[5] BONEH D, SHACHAM H. Group signatures with verifier-local revocation[C]// ACM Conference on Computer and Communications Security. Washington: ACM, 2004, 8383: 168-177.

[6] LANGLOIS A, LING S, NGUYEN K, et al. Lattice-based group signature scheme with verifier-local revocation[C]// Advances in Public-Key Cryptography – PKC 2014. Berlin Heidelberg: Springer, 2014, 8383: 345-361.

[7] LIBERT B, LING S, MOUHARTEM F, et al. Signature schemes with efficient protocols and dynamic group signatures from lattice assumptions[C]//Advances in Cryptology – ASIACRYPT 2016. Berlin Heidelberg: Springer, 2016, 10032: 373-403.

[8] GENTRY C, PEIKERT C, VAIKUNTANATHAN V. Trapdoors for hard lattices and new cryptographic constructions[C]//Proceedings of the fortieth annual ACM Symposium on Theory of Computing. Victoria: ACM, 2008: 197-206.

[9] LING S, NGUYEN K, WANG H, et al. Lattice-based group signatures: achieving full dynamicity with ease[C]//Applied Cryptography and Network Security. Kanazawa: Springer, 2017, 10355: 293-312.

[10] BOOTLE J, CERULLI A, CHAIDOS P, et al. Foundations of fully dynamic group signatures[C]//Applied Cryptography and Network Security. Guildford: Springer, 2016: 117-136.

[11] BRICKELL E, POINTCHEVAL D, VAUDENAY S, et al. Classical hardness of learning with errors[C]//ACM Symposium on Theory of Computing. Palo Alto: ACM, 2013: 575-584.

[12] REGEV O. On lattices, learning with errors, random linear codes, and cryptography[C]//ACM Symposium on Theory of Computing. Baltimore: ACM, 2005: 84-93.

[13] MICCIANCIO D, PEIKERT C. Trapdoors for lattices: Simpler, tighter, faster, smaller[C]// Theory and Application of Cryptographic Techniques. Cambridge: Springer, 2012, 7237: 700-718.

[14] CASH D, HOFHEINZ D, KILTZ E, et al. Bonsai trees, or how to delegate a lattice basis[J]. Journal of Cryptology, 2012, 25(4): 601-639.

[15] AGRAWAL S, DAN B, BOYEN X. Efficient lattice (H)IBE in the standard model[C]//Advances in Cryptology – EUROCRYPT 2010. Riviera: Springer, 2010, 6110: 553-572.

[16] LING S, NGUYEN K, STEHLE D, et al. Improved zero-knowledge proofs of knowledge for the ISIS problem, and applications[C]//Public-Key Cryptography – PKC 2013. Berlin Heidelberg: Springer, 2013, 7778: 107-124.

[17] KAWACHI A, TANAKA K, XAGAWA K. Concurrently secure identification schemes based on the worst-case hardness of lattice problems[C]//International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology. Melbourne: Springer, 2008, 5350: 372-389.

[18] KIAYIAS A, YUNG M. Secure scalable group signature with dynamic joins and separable authorities[J]. International Journal of Security and Networks, 2006, 1(1/2): 24-45.


编　辑　蒋　晓