标准模型下可撤销的基于身份的代理重签名方案

杨小东[1,2]，李雨潼[1]，王晋利[1]，麻婷春[1]，王彩芬[1]
（1. 西北师范大学计算机科学与工程学院，甘肃 兰州 730070；2. 密码科学技术国家重点实验室，北京 100878）

摘 要：用户撤销是基于身份的代理重签名方案在应用中必须解决的重要问题。针对目前基于身份的代理重签名方案不支持用户撤销的问题，引入了可撤销的基于身份代理重签名密码体制，并给出了相应的形式化定义和安全模型。基于代理重签名方案和二叉树结构，构造了一个可撤销的基于身份的代理重签名方案。在所构造的方案中，用户的签名密钥由秘密密钥和更新密钥两部分组成。通过安全信道传输的秘密密钥是固定的，但利用公开信道广播的更新密钥是周期性变化的。只有未被撤销的用户才能获得更新密钥，并使秘密密钥随机化，更新密钥生成当前时间段的签名密钥。在标准模型下证明了所构造的方案在适应性选择身份和消息攻击下是存在不可伪造的，并满足双向性、多用性和抗签名密钥泄露攻击性。分析结果表明，所构造的方案高效地实现了用户的撤销与密钥的更新，具有良好的延展性。

关键词：基于身份的代理重签名；用户撤销；标准模型；签名密钥泄露；二叉树
中图分类号：TP309.7
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2019072

Revocable identity-based proxy re-signature scheme in the standard model

YANG Xiaodong[1,2], LI Yutong[1], WANG Jinli[1], MA Tingchun[1], WANG Caifen[1]
1. College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China
2. State Key Laboratory of Cryptology, Beijing 100878, China

Abstract: User revocation is necessary to the practical application of identity-based proxy re-signature scheme. To solve the problem that the existing identity-based proxy re-signature schemes cannot provide revocation functionality, the notion of revocable identity-based proxy re-signature was introduced. Furthermore, the formal definition and security model of revocable identity-based proxy re-signature were presented. Based on proxy re-signature scheme and binary tree structure, a revocable identity-based proxy re-signature scheme was proposed. In the proposed, scheme, the user's signing key consists of two parts, a secret key and an update key. The secret key transmitted over the secure channel is fixed, but the update key broadcasted by the public channel is periodically changed. Only the user who has not been revoked can obtain the update key, and then randomize the secret key and update the key to generate the corresponding signature key of the current time period. In the standard model, the proposed scheme is proved to be existentially unforgeable against adaptive chosen-identity and chosen-message attacks. In addition, the proposed scheme has properties of bidirectionality and multi-use, and can resist signing key exposure attacks. The analysis results show that the proposed scheme can efficiently revoke the user and update the user’s key, and thus it has good scalability.

Key words: identity-based proxy re-signature, user revocation, standard model, signing key exposure, binary tree

收稿日期：2018−05−14；**修回日期**：2018−08−16
基金项目：国家自然科学基金资助项目（No.61662069, No.61562077）；中国博士后科学基金资助项目（No.2017M610817）；兰州市科技计划基金资助项目（No.2013-4-22）；西北师范大学青年教师科研能力提升计划基金资助项目（No.NWNU-LKQN-14-7）
Foundation Items: The National Natural Science Foundation of China (No.61662069, No.61562077), China Postdoctoral Science Foundation Project (No.2017M610817), The Science and Technology Project of Lanzhou (No.2013-4-22), The Foudation for Excellent Yong Teachers by Northwest Normal University (No.NWNU-LKQN-14-7)

1 引言

基于身份的代理重签名是一种具有转换签名功能的密码体制，一个半可信的代理者能将 Alice 的签名转换为 Bob 对同一个消息的签名，但代理者无法获得 Alice 或 Bob 的签名密钥的任何信息。在一种基于身份的代理重签名方案中，用户选择如 E-mail 地址、手机号码等身份标识作为公钥，而私钥由可信的密钥生成中心（PKG, private key generator）生成并通过安全的信道发送给用户。基于身份的代理重签名体制避免了复杂的公钥证书，简化了密钥管理，在云计算、大数据隐私保护等方面有广泛的应用[1-2]。

基于文献[3]的加密方案，Shao 等[4]构造了第一种标准模型下安全的基于身份的代理重签名方案；Feng 等[5]利用抗碰撞的散列函数设计了一种基于身份的代理重签名方案，但该方案不满足多用性；Hu 等[6]提出了一种紧规约的基于身份的代理重签名方案，但该方案的安全性依赖于较强的困难问题假设；Shao 等[7]又设计了一种随机预言模型下安全的单向基于身份的代理重签名方案，但该方案的验证重签名的计算开销与重签名级数呈线性增长关系；Huang 等[8]提出了一种无双线性对的基于身份的代理重签名方案，并在随机预言模型中证明了该方案的安全性可归约到离散对数问题。文献[9-10]分别构造了格上基于身份的代理重签名方案，但这 2 种方案的参数尺寸、密钥长度和签名长度都比较大，并且安全性依赖于理想的随机预言机。然而，在随机模型中被证明安全的密码方案，当利用具体的散列函数实例化随机预言机时，随机预言模型并不能确保方案的实际安全性[11]。因此，研究标准模型下安全的基于身份的代理重签名方案具有一定的现实意义。

在实际应用中，任何实用的密码系统都面临用户撤销的问题，比如用户权限到期或用户密钥泄露，这就需要从密码系统中撤销用户。为了实现基于身份的用户撤销机制，文献[12]提出了 PKG 定期更新未撤销用户密钥的撤销方法，但 PKG 与未撤销用户之间需要建立一个安全信道来传输更新密钥。Boldyreva 等[13]提出了 PKG 通过公开信道实现用户撤销的方法，随后一些基于 Boldyreva 所提方法的可撤销加密方案相继被提出[14-15]。为了解决签名方案中的用户撤销问题，Tsai 等[16]提出了第一种可撤销的基于身份签名方案，但 Liu 等[17]发现该方案存在签名密钥泄露的安全风险。近年来，一些具有特殊性质的可撤销签名方案[18-20]也相继被提出，如可撤销的属性基签名[21]、可撤销的无证书签名[22-23]、可撤销的代理签名[24]、可撤销的前向安全签名[25]等。

现有的基于身份的代理重签名方案均未考虑用户撤销问题，而用户撤销功能对一种实用的基于身份的代理重签名方案是非常重要的。例如 Alice 是某公司的总经理，负责该公司所有文件的签名。如果 Alice 因公出差，由该公司的副总经理 Bob 生成公司文件的签名，利用 Alice 和 Bob 之间的重签名密钥，将 Bob 的签名转换为 Alice 对同一文件的签名。由于工作调动或身体健康的原因，Bob 离开了公司。为了公司的利益，需要撤销 Bob 的签名权限。因此，基于身份代理重签名体制在实现签名转换的同时，还必须支持高效的用户撤销机制。

鉴于此，本文提出了可撤销的基于身份代理重签名方案，并给出了相应的形式化定义与安全模型。基于 Shao 等[4]提出的代理重签名方案和二叉树结构，本文构造了一种可撤销的双向基于身份的代理重签名方案，在标准模型下证明了所构造方案的安全性可规约到计算性 Diffie-Hellman Diffie-Hellman（CDH, computational Diffie-Hellman）困难问题。本文所提方案实现了用户的撤销与密钥的更新，能抵抗签名密钥泄露攻击，并且 PKG 的工作量与用户的数量呈对数增长关系，具有良好的延展性。

2 准备知识

2.1 符号说明

本文所使用的符号及其说明如表 1 所示。

2.2 双线性映射

令 p 是一个大素数，G 和 G_T 是 2 个阶为 p 的乘法循环群，g 是 G 的一个生成元。如果一个可计算的映射 $e: G\times G\to G_T$ 满足以下条件，则称 e 是一个双线性映射[4]。

1) 双线性：对任意的 $a,b\in Z_p$，有 $e(g^a,g^b)=e(g,g)^{ab}=e(g^b,g^a)$。

2) 非退化性：$e(g,g)\neq 1_{G_T}$。

2.3 KUNode 算法

选择基于二叉树的 KUNode 算法[26]来提高未撤销用户的密钥更新效率。令 BT 表示一棵具有 N 个叶子节点的二叉树，root 表示树的根节点。对于

表 1　　**符号列表及其说明**

符号	说明	符号	说明	符号	说明
BT	二叉树	η	叶子节点	G 或 G_T	循环群
N	叶子节点个数、用户数	path(η)	η 到 root 的节点集合	p	素数
root	根节点	t_i	时间周期	g	G 的生成元
θ	非叶子节点	Y	更新的最小节点集合	e	双线性映射
θ_l、θ_r	左/右孩子节点	RL 或 RL′	用户撤销列表	T	最大时间周期
st	状态	msk	主密钥	M	消息
ID	用户身份	m 和 n	字符串的比特长度	sk_θ	秘密密钥
uk_t	更新密钥	$\mathrm{dk}_{\mathrm{ID},t}$	签名密钥	σ_A	原始签名
σ_B	重签名	$\mathrm{rk}_{A\to B}$	重签名密钥	$\perp$	错误符号

每个非叶子节点 θ，用 θ_l 和 θ_r 表示 θ 的左孩子节点和右孩子节点。每个用户将分配至 BT 上的一个叶子节点 η，用 path(η)表示从叶子节点 η 到根节点 root 的路径上的所有节点集合。RL 表示一个由叶子节点 η_i 和时间周期 t_i 组成的撤销列表，Y 表示 BT 中需要更新的最小节点集合。如果 $\eta\in$RL，则 path(η) $\cap Y=\varnothing$。KUNode 算法的输入是 BT、RL 和时间周期 t，输出是 Y。KUNode 算法的具体描述如算法 1 所示。

算法 1　KUNode 算法

输入　BT、RL 和时间周期 t

输出　需要更新的最小节点集合 Y

1) 设置集合 $X=\varnothing$, $Y=\varnothing$
2) ∀ $(\eta_i, t_i)\in$ RL
3)　　如果 $t_i\leqslant t$，在 X 中添加 path(η_i)
4) ∀ $\theta\in X$
5)　　如果 $\theta_l\notin X$，在 Y 中添加 θ_l
6)　　如果 $\theta_r\notin X$，在 Y 中添加 θ_r
7) 如果 $Y=\varnothing$，在 Y 中添加 root
8) 返回 Y

2.4　困难问题假设

计算性问题：给定 $(g,g^a,g^b)\in G^3$，其中 $a,b\in Z_p$，计算 $g^{ab}\in G$。

定义 1　CDH 假设。如果所有多项式时间算法无法解决 G 上 CDH 问题，则称 G 上的 CDH 问题是困难的[4]。

3　可撤销的基于身份代理重签名的安全性定义

3.1　形式化定义

一种撤销的双向基于身份的代理重签名方案由下面 9 种算法构成。

1) setup $(1^\lambda,N,T)\to$ (pp,msk,RL,st) 是系统参数建立算法：输入安全参数 λ、最大用户数 N 和用户签名密钥有效期的最大时间周期 T，输出公开参数 pp、PKG 的主密钥 msk、初始化为空集的用户撤销列表 RL 和状态 st。

2) extract $(\mathrm{pp},\mathrm{msk},\mathrm{ID})\to \mathrm{sk}_{\mathrm{ID}}$ 是秘密密钥生成算法：输入 pp、msk 和用户身份 ID，输出秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$。

3) KeyUp $(\mathrm{pp},\mathrm{msk},t,\mathrm{RL},\mathrm{st})\to \mathrm{uk}_t$ 是密钥更新算法：输入 pp、msk、更新的时间周期 t、当前的用户撤销列表 RL 和状态 st，如果 $t>T$，输出一个错误符号“⊥”；否则，输出一个更新密钥 uk_t。

4) SKGen $(\mathrm{pp},\mathrm{sk}_{\mathrm{ID}},\mathrm{uk}_t)\to \mathrm{dk}_{\mathrm{ID},t}$ 是签名密钥生成算法：输入 pp、$\mathrm{sk}_{\mathrm{ID}}$ 和 uk_t，如果用户身份 ID 在时间周期 t 已被撤销，输出⊥；否则，输出对应于 ID 和 t 的签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$。

5) ReKey $(\mathrm{pp},\mathrm{dk}_{A,t},\mathrm{dk}_{B,t})\to \mathrm{rk}_{A\to B,t}$ 是重签名密钥生成算法：输入 pp、用户身份 ID_A 和 ID_B 对应的签名密钥 $\mathrm{dk}_{A,t}$ 与 $\mathrm{dk}_{B,t}$，输出代理者的一个重签名密钥 $\mathrm{rk}_{A\to B,t}$。

6) sign $(\mathrm{pp},\mathrm{dk}_{\mathrm{ID},t},t,M)\to\sigma$ 是签名生成算法：输入 pp、$\mathrm{dk}_{\mathrm{ID},t}$、当前时间周期 t ($t\leqslant T$)和消息 M，输出关于 M 的签名 σ，其中 $\mathrm{dk}_{\mathrm{ID},t}$ 是用户身份 ID 在时间周期 t 的签名密钥。

7) resign $(\mathrm{pp},\mathrm{rk}_{A\to B,t},\mathrm{ID}_A,t,M,\sigma_A)\to\sigma_B$ 是重签名生成算法：输入 pp、$\mathrm{rk}_{A\to B,t}$ 和对应于身份 ID_A 和时间周期 t 的关于消息 M 的签名 σ_A，如果 verify $(\mathrm{pp},\mathrm{ID}_A,t,M,\sigma_A)$=0，输为⊥；否则，输出对应于

用户身份 ID_B 和 t 的关于 M 的签名 σ_B。

8) verify $(\mathrm{pp},\mathrm{ID},t,M,\sigma)\rightarrow\{0,1\}$ 是签名验证算法：输入 pp、身份 ID、时间周期 t ($t\leqslant T$)、消息 M 和签名 σ，如果 σ 是合法的，输出 1；否则，输出 0。

9) revoke $(\mathrm{ID},t,\mathrm{RL},\mathrm{st})\rightarrow \mathrm{RL}'$ 是用户撤销算法：输入时间周期 t ($t\leqslant T$)内，撤销的用户身份 ID、用户撤销列表 RL 和状态 st，输出更新后的用户撤销列表 RL′。

3.2 安全模型

借鉴基于身份的代理重签名方案的安全模型[4,7]和可撤销的基于身份签名方案的安全性定义[17-19]，本文通过攻击者 A 和挑战者 C 之间的安全游戏，给出可撤销的双向基于身份的代理重签名方案的安全模型。

初始化 C 运行 $\mathrm{setup}(1^{\lambda},N,T)$算法，将生成的系统参数 pp 发送给 A，自己秘密保存主密钥 msk。

询问 A 自适应地向 C 发起有限次的如下询问。

1) extract-query：对于 A 请求的关于身份 ID 的秘密密钥询问，C 运行算法 $\mathrm{extract}(\mathrm{pp},\mathrm{msk},\mathrm{ID})$，将输出的秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$ 发送给 A。

2) KeyUp-query：对于 A 请求的关于时间周期 t($t\leqslant T$)的更新密钥询问，其中 t 不能小于以前所有询问过的时间周期，C 运行算法 $\mathrm{KeyUp}(\mathrm{pp},\mathrm{msk},t,\mathrm{RL},\mathrm{st})$，将生成的更新密钥 uk_t 发送给 A。

3) SKGen-query：收到 A 发送的关于 (ID,t) 的签名密钥询问后，C 首先询问关于 ID 的 extract-query 和关于 $t\leqslant T$ 的 KeyUp-query，分别获得相应的秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$ 与更新密钥 uk_t；然后运行算法 $\mathrm{SKGen}(\mathrm{pp},\mathrm{sk}_{\mathrm{ID}},\mathrm{uk}_t)$生成签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$，并将 $\mathrm{dk}_{\mathrm{ID},t}$ 发送给 A。为了不失一般性，这里要求 A 未进行关于 t 的 KeyUp-query 询问前，不能发起关于 t 的 SKGen-query 询问。

4) ReKey-query：收到 A 发送的关于 2 个身份 $(\mathrm{ID}_A,\mathrm{ID}_B)$ 和时间周期 t 的重签名密钥询问后，C 首先询问关于 (ID_A,t) 和 (ID_B,t) 的 SKGen-query，获得对应的签名密钥 $\mathrm{dk}_{A,t}$ 与 $\mathrm{dk}_{B,t}$；然后将算法 $\mathrm{ReKey}(\mathrm{pp},\mathrm{dk}_{A,t},\mathrm{dk}_{B,t})$生成的重签名密钥 $\mathrm{rk}_{A\rightarrow B,t}$ 发送给 A。

5) sign-query：收到 A 发送的关于身份 ID、时间周期 t($t\leqslant T$)与消息 M 的签名询问后，C 询问关于 (ID,t) 的 SKGen-query 获得签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$，并运行算法 $\mathrm{sign}(\mathrm{pp},\mathrm{dk}_{\mathrm{ID},t},t,M)$，将生成的 M 的签名 σ 返回给 A。

6) revoke-query：如果 A 在时间周期 t($t\leqslant T$)发起过关于身份 ID 的 KeyUp-query，则 A 在时间周期 t 不能发起关于 ID 的 revoke-query。收到 A 发送的 (ID,t) 后，其中 t 不能小于以前所有询问过的时间周期，C 运行算法 $\mathrm{revoke}(\mathrm{ID},t,\mathrm{RL},\mathrm{st})$，并将输出的用户撤销列表 RL′ 返回给 A。

伪造 攻击者 A 最后输出身份 ID^*、时间周期 $t^*(t\leqslant T)$、消息 M^* 和签名 σ^*。如果以下 5 个条件均成立，则称 A 在以上游戏中获胜。

1) $\mathrm{verify}(\mathrm{pp},\mathrm{ID}^*,t^*,M^*,\sigma^*)=1$。

2) (ID^*,t^*) 未进行过 SKGen-query 询问。

3) ID^* 未进行过 extract-query 询问，并且 $t^*\leqslant T$ 未进行过 KeyUp-query 询问。

4) ID^* 未进行过 ReKey-query 询问。

5) (ID^*,t^*,M^*) 未进行过 sign-query 询问。

定义 2 若任何一个多项式时间攻击者 A 在上述游戏中获胜的概率是可忽略的，则称可撤销的双向基于身份的代理重签名方案在适应性选择身份和消息攻击下是存在不可伪造的。

定义 3 如果攻击者无法从当前时间周期 t 泄露的签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$ 中获取秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$ 或威胁其他时间周期 $\tilde{t}(\tilde{t}\neq t)$ 签名密钥 $\mathrm{dk}_{\mathrm{ID},\tilde{t}}$ 的安全性，则称可撤销的基于身份的代理重签名方案满足抗签名密钥泄露攻击性。

4 可撤销的双向基于身份的代理重签名方案

在 Shao 方案[4]基础上，本文 2.3 节的 KUNode 算法构造了一种可撤销的双向基于身份的代理重签名方案。假设用户身份的长度和签名消息的长度分别为 m bit 和 n bit，可通过 2 个散列函数 $H_1:\{0,1\}^*\rightarrow\{0,1\}^m$ 和 $H_2:\{0,1\}^*\rightarrow\{0,1\}^n$，将用户身份和签名消息的固定长度延伸为可变长度，即对于任意长度的身份 ID′ 和签名消息 M'，计算 $H_1(\mathrm{ID}')=(\mathrm{ID}'_1,\cdots,\mathrm{ID}'_m)\in\{0,1\}^m$ 和 $H_2(M')=(M'_1,\cdots,M'_n)\in\{0,1\}^n$，从而将身份 ID′ 和消息 M' 分别映射为长度为 m 和 n 的比特串。为了简化表述，假设方案中的用户身份和消息都进行了散列函数 H_1 和 H_2 的预处理，即用符号 ID 和 M 分别表示长度为 m bit 的用户身份和长度为 n bit 的签名消息。

4.1 方案描述

1) setup 算法

输入安全参数 λ、最大用户数 N 和最大时间周

期 T，PKG 执行如下操作。

① 选择一个大素数 p，2 个阶为 p 的乘法循环群 G 和 G_T，一个 G 的生成元 g 和一个双线性映射 $e: G\times G\to G_T$。

② 在 G 中随机选取 $g_2, u_0, u_1, \cdots, u_m, v, v', w_0, w_1, \cdots, w_n$。

③ 随机选择 $\alpha\in Z_p^*$，计算 $g_1=g^\alpha$。

④ 选择一棵具有 N 个叶子节点的二叉树 BT，设置用户撤销列表 RL=∅和状态 st=BT。

⑤ 秘密保存主密钥 $\text{msk}=\alpha$，公开参数 $\text{pp}=(G, G_T, e, p, g, g_1, g_2, u_0, u_1, \cdots, u_m, v, v', w_0, w_1, \cdots, w_n)$。

为了表述方便，对于长度为 m bit 的用户身份 $\text{ID}=(\text{ID}_1, \cdots, \text{ID}_m)\in\{0,1\}^m$ 和长度为 n bit 的签名消息 $M=(M_1, \cdots, M_n)\in\{0,1\}^n$，令 $F_{W,1}(\text{ID})=u_0\prod_{i=1}^{m}(u_i)_1^{\text{ID}_i}$ 和 $F_{W,2}(M)=w_0\prod_{j=1}^{n}(w_j)^{M_j}$。

2) extract 算法

为了生成用户身份 ID 的秘密密钥，PKG 执行如下操作。

① 在 BT 上随机选择一个空的叶子节点 η，并在 η 中保存 ID。

② 对于每个节点 $\theta\in\text{path}(\eta)$，随机选择 $g_\theta\in G$，并在 θ 中保存 $(g_\theta, \tilde{g}_\theta=\frac{g_2}{g_\theta})$；选择一个随机数 $r_\theta\in Z_p$，计算 $\text{sk}_\theta=(\text{sk}_{\theta,1}, \text{sk}_{\theta,2})=(g_\theta^\alpha F_{W,1}(\text{ID})^{r_\theta}, g^{r_\theta})$。

③ 将秘密密钥 $\text{sk}_{\text{ID}}=\{(\theta, sk_\theta)\}_{\theta\in\text{path}(\eta)}$ 通过一个安全信道发送给用户。

3) KeyUp 算法

对于时间周期 t、撤销列表 RL 和状态 st，如果 $t>T$，输出⊥；否则，PKG 通过如下步骤生成更新密钥 uk_t。

① 对于每个节点 $\theta\in\text{KUNode (BT, RL, }t)$，首先在 θ 中提取 $\tilde{g}_\theta$，然后随机选取 $s_\theta\in Z_p$，计算 $\text{uk}_\theta=(\text{uk}_{\theta,1}, \text{uk}_{\theta,2})=((\tilde{g}_\theta)^\alpha(v'v^t)^{s_\theta}, g^{s_\theta})$。

② 通过一个公开信道广播更新密钥 $\text{uk}_t=\{(\theta, \text{uk}_\theta)\}_{\theta\in\text{KUNode(BT,RL,}t)}$。

如果在时间周期 t 的撤销列表 RL 中包含了被撤销的用户身份 ID，则 PKG 调用 KUNode 算法[26]生成需要更新的最小节点集合 Y=KUNode (BT, RL, t)，但被撤销的用户无法获得 PKG 生成的更新密钥 uk_t。下面通过一个实例来说明撤销用户的方法。

如图 1 所示，假设 4 个用户 u_1, u_2, u_3 和 u_4 被分配至 4 个叶子节点 η_3, η_4, η_5 和 η_6。假设用户 u_3 被撤销，则 X=path(η_5)={η_5, η_2, root}，Y={η_1, η_6}，path(η_5)∩Y=∅。除了 u_3 外，从每个用户对应的叶子节点到根节点的路径上至少包含 Y 中的一个节点。用户 u_1 和 u_2 的路径上包含 Y 中的节点 η_1，用户 u_4 的路径上包含 Y 中的节点 η_6。在时间周期 t，PKG 只需更新 Y={η_1, η_6}中每个节点的密钥，便可完成未撤销用户 u_1, u_2 和 u_4 的密钥更新，从而实现对用户 u_3 的撤销。

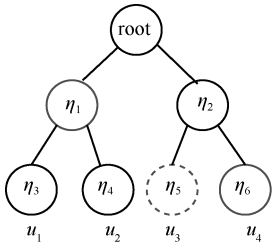

图 1　KUNode 算法的一个实例

4) SKGen 算法

如果用户身份 ID 在时间周期 $t\leqslant T$ 已被撤销，输出⊥；否则，一定存在一个节点 $\theta\in$KUNode (BT, RL, t)∩path(η)；用户随机选取 r_{ID}、$s_{\text{ID}}\in Z_p$，利用自己的秘密密钥 $\text{sk}_\theta=(\text{sk}_{\theta,1}, \text{sk}_{\theta,2})$ 和公开的更新密钥 $\text{uk}_\theta=(\text{uk}_{\theta,1}, \text{uk}_{\theta,2})$，计算签名密钥为

$$\text{dk}_{\text{ID},t}=(\text{dk}_{\text{ID},t,1}, \text{dk}_{\text{ID},t,2}, \text{dk}_{\text{ID},t,3})=(\text{sk}_{\theta,1}$$
$$F_{W,1}(\text{ID})^{r_{\text{ID}}}uk_{\theta,1}(v'v^t)^{s_{\text{ID}}}, \text{sk}_{\theta,2}g^{r_{\text{ID}}}, \text{uk}_{\theta,2}g^{s_{\text{ID}}})=$$
$$(g_2^\alpha F_{W,1}(\text{ID})^{r_\theta+r_{\text{ID}}}(v'v^t)^{s_\theta+s_{\text{ID}}}, g^{r_\theta+r_{\text{ID}}}, g^{s_\theta+s_{\text{ID}}})$$

5) ReKey 算法

给定 2 个用户身份 ID_A 和 ID_B 的签名密钥 $\text{dk}_{A,t}=(\text{dk}_{A,t,1}, \text{dk}_{A,t,2}, \text{dk}_{A,t,3})$ 和 $\text{dk}_{B,t}=(\text{dk}_{B,t,1}, \text{dk}_{B,t,2}, \text{dk}_{B,t,3})$，采用类似文献[4]的安全协议，生成代理者的一个重签名密钥为

$$\text{rk}_{A\to B,t}=(\text{rk}_{A\to B,t,1}, \text{rk}_{A\to B,t,2}, \text{rk}_{A\to B,t,3})=$$
$$\left(\frac{\text{dk}_{B,t,1}}{\text{dk}_{A,t,1}}, \frac{\text{dk}_{B,t,2}}{\text{dk}_{A,t,2}}, \frac{\text{dk}_{B,t,3}}{\text{dk}_{A,t,3}}\right)$$

6) sign 算法

对于当前时间周期 $t\leqslant T$ 和一个消息 M，身份为 ID_A 的签名者随机选取 $r_m\in Z_p$，利用签名密钥 $\text{dk}_{A,t}=(\text{dk}_{A,t,1}, \text{dk}_{A,t,2}, \text{dk}_{A,t,3})$ 计算 $\sigma_{A,1}=\text{dk}_{A,t,1}F_{W,2}(M)^{r_m}$，$\sigma_{A,2}=\text{dk}_{A,t,2}$，$\sigma_{A,3}=\text{dk}_{A,t,3}$ 和 $\sigma_{A,4}=g^{r_m}$，输

出一个关于 M 的签名 $\sigma_A=(\sigma_{A,1},\sigma_{A,2},\sigma_{A,3},\sigma_{A,4})$。

7) resign 算法

给定重签名密钥 $\mathrm{rk}_{A\to B,t}=(\mathrm{rk}_{A\to B,t,1},\mathrm{rk}_{A\to B,t,2},\mathrm{rk}_{A\to B,t,3})$、对应于身份 ID_A 和时间周期 $t\leqslant T$ 的关于消息 M 的签名 $\sigma_A=(\sigma_{A,1},\sigma_{A,2},\sigma_{A,3},\sigma_{A,4})$，如果 verify (pp,$\mathrm{ID}_A$,$t$,$M$,$\sigma_A$)=0，输出 $\perp$；否则，代理者随机选取 $r_m^{'}\in Z_p$，计算 $\sigma_{B,1}=\sigma_{A,1}\mathrm{rk}_{A\to B,t,1}F_{W,2}(M)^{r_m^{'}}$，$\sigma_{B,2}=\sigma_{A,2}\mathrm{rk}_{A\to B,t,2}$，$\sigma_{B,3}=\sigma_{A,3}\,\mathrm{rk}_{A\to B,t,3}$ 和 $\sigma_{B,4}=\sigma_{A,4}g^{r_m^{'}}$，输出一个对应于身份 ID_B 和 t 的关于 M 的签名 $\sigma_B=(\sigma_{B,1},\sigma_{B,2},\sigma_{B,3},\sigma_{B,4})$。

8) verify 算法

给定身份 ID、消息 M、时间周期 $t\leqslant T$ 和签名 $\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4)$，验证者检查以下等式是否成立。

$$e(\sigma_1,g)=e(g_2,g_1)e(F_{W,1}(\mathrm{ID}),\sigma_2)$$

$$e(v^{'}v^{t},\sigma_3)e(F_{W,2}(M),\sigma_4)$$

如果等式成立，说明 σ 是一个合法的签名，输出 1；否则，输出 0。

9) revoke 算法

给定 BT 上存储被撤销用户身份 ID 的叶子节点 η，当前时间周期 $t\leqslant T$、用户撤销列表 RL 和状态 st，PKG 在 RL 中添加 (η,t)，即 $\mathrm{RL}'=\mathrm{RL}\cup\{(\eta,t)\}$，并输出更新后的用户撤销列表 RL′。

4.2 正确性分析

假设身份 ID_B 的签名密钥 $\mathrm{dk}_{B,t}=(\mathrm{dk}_{B,t,1},\mathrm{dk}_{B,t,2},\mathrm{dk}_{B,t,3})=(g_2^{\alpha}F_{W,1}(\mathrm{ID}_B)^{r_{\theta_B}+r_B}(v^{'}v^{t})^{s_{\theta_B}+s_B},g^{r_{\theta_B}+r_B},g^{s_{\theta_B}+s_B})$，对于消息 M 的重签名 $\sigma_B=(\sigma_{B,1},\sigma_{B,2},\sigma_{B,3},\sigma_{B,4})$，其中

$$\sigma_{B,1}=\sigma_{A,1}\mathrm{rk}_{A\to B,t,1}F_{W,2}(M)^{r_m^{'}}=$$

$$\mathrm{d}k_{A,t,1}F_{W,2}(M)^{r_m}\frac{\mathrm{d}k_{B,t,1}}{\mathrm{d}k_{A,t,1}}F_{W,2}(M)^{r_m^{'}}=$$

$$\mathrm{d}k_{B,t,1}F_{W,2}(M)^{r_m+r_m^{'}}=$$

$$g_2^{\alpha}F_{W,1}(\mathrm{ID}_B)^{r_{\theta_B}+r_B}(v^{'}v^{t})^{s_{\theta_B}+s_B}\ F_{W,2}(M)^{r_m+r_m^{'}}$$

$$\sigma_{B,2}=\sigma_{A,2}\mathrm{rk}_{A\to B,t,2}=$$

$$\mathrm{d}k_{A,t,2}\left(\frac{\mathrm{d}k_{B,t,2}}{\mathrm{d}k_{A,t,2}}\right)=\mathrm{d}k_{B,t,2}=g^{r_{\theta_B}+r_B}$$

$$\sigma_{B,3}=\sigma_{A,3}\mathrm{rk}_{A\to B,t,3}=\mathrm{d}k_{A,t,3}\frac{\mathrm{d}k_{B,t,3}}{\mathrm{d}k_{A,t,3}}=\mathrm{dk}_{B,t,3}=g^{s_{\theta_B}+s_B}$$

$$\sigma_{B,4}=\sigma_{A,4}g^{r_m^{'}}=g^{r_m}g^{r_m^{'}}=g^{r_m+r_m^{'}}$$

则 σ_B 的正确性验证如下

$$e(\sigma_{B,1},g)=e(g_2^{\alpha}F_{W,1}(\mathrm{ID}_B)^{r_{\theta_B}+r_B}(v^{'}v^{t})^{s_{\theta_B}+s_B}$$

$$F_{W,2}(M)^{r_m+r_m^{'}},g)=e(g_2,g_1)e(F_{W,1}(\mathrm{ID}_B),\sigma_{B,2})$$

$$e(v'v^{t},\sigma_{B,3})e(F_{W,2}(M),\sigma_{B,4})$$

上述推导过程证明了本文方案的正确性。

因为 sign 算法生成的原始签名 $\sigma_A=(\sigma_{A,1},\sigma_{A,2},\sigma_{A,3},\sigma_{A,4})$ 包含了 G 中的 4 个元素，resign 算法生成的重签名 $\sigma_B=(\sigma_{B,1},\sigma_{B,2},\sigma_{B,3},\sigma_{B,4})$ 也包含了 G 中的 4 个元素，所以 σ_B 也可以作为 resign 算法的原始签名。代理者利用重签名密钥 $\mathrm{rk}_{B\to C,t}$ 和 resign 算法能将签名 σ_B 转换成新的签名 σ_C，进而实现签名的多次转换，因此本文方案满足多用性。

通过 ID_A 与 ID_B 间的重签名密钥 $\mathrm{rk}_{A\to B,t}$，很容易计算出 ID_B 与 ID_A 间的重签名密钥 $\mathrm{rk}_{B\to A,t}=\dfrac{1}{\mathrm{rk}_{A\to B,t}}$，所以本文方案具有双向性。

4.3 安全性分析

定理 1 如果 CDH 假设成立，则本文方案满足标准模型下自适应性选择身份和消息攻击的存在不可伪造性。

证明 假设攻击者 A 进行了最多 q_{sk} 次秘密密钥询问、q_{uk} 次更新密钥询问、q_{dk} 次签名密钥询问、q_{rk} 次重签名密钥询问、q_{s} 次签名询问和 q_{r} 次撤销询问后，以不可忽略的概率 ε 攻破了本文方案的存在不可伪造性，则存在一个挑战者 C 将利用攻击者 A 的伪造，以不可忽略的概率 $\varepsilon^{'}$ 解决 G 上的 CDH 问题。对于一个 CDH 问题实例 $(g,g^a,g^b)\in G^3$，C 的目标是计算 g^{ab}。

初始化 C 设置参数 $l_u=2(q_{\mathrm{sk}}+q_{\mathrm{dk}}+q_{\mathrm{rk}}+q_{\mathrm{s}})$ 和 $l_m=2q_{\mathrm{s}}$，满足 $l_u(m+1)<p$ 和 $l_m(n+1)<p$。随机选取 2 个整数 $k_u(0\leqslant k_u\leqslant m)$ 和 $k_m(0\leqslant k_m\leqslant n)$，并随机选择 $x_0,x_1,\cdots,x_m\in Z_{l_u}$，$c_0,c_1,\cdots,c_n\in Z_{l_m}$，$v_0,v_1,y_0,y_1,\cdots,y_m,d_0,d_1,\cdots,y_n\in Z_p$。选择一棵具有 N 个叶子节点的二叉树 BT，设置最大时间周期 T、用户撤销列表 RL=∅和状态 st=BT，参数 $g_1=g^a$，$g_2=g^b$，$u_0=g_2^{-l_uk_u+x_0}g^{y_0}$，$u_i=g_2^{x_i}g^{y_i}(1\leqslant i\leqslant m)$，$v^{'}=g^{v_0}$，$v=g^{v_1}$，$w_0=g_2^{-l_mk_m+c_0}g^{d_0}$ 和 $w_j=g_2^{c_j}g^{d_j}(1\leqslant j\leqslant n)$，并发送系统参数 $\mathrm{pp}=(G,G_T,e,p,g,g_1,g_2,u_0,u_1,\cdots,u_m,v,v^{'},w_0,w_1,\cdots,w_n)$ 给 A。

由参数 $g_1=g^a$ 可知，系统的主密钥为 a，但 a 对挑战者 C 来说是未知的。为了描述方便，对于长度为 m bit 的用户身份 $\mathrm{ID}=(\mathrm{ID}_1,\cdots,\mathrm{ID}_m)\in\{0,1\}^m$ 和

长度为 n bit 的签名消息 $M=(M_1,\cdots,M_n)\in\{0,1\}^n$，定义下面 4 个函数。

$$F(\mathrm{ID})=x_0-l_uk_u+\sum_{i=1}^{m}x_i\mathrm{ID}_i$$

$$J(\mathrm{ID})=y_0+\sum_{i=1}^{m}y_i\mathrm{ID}_i$$

$$K(M)=c_0-l_mk_m+\sum_{j=1}^{n}c_jM_j$$

$$L(M)=d_0+\sum_{j=1}^{n}d_jM_j$$

于是有 $F_{W,1}(\mathrm{ID})=u_0\prod_{i=1}^{m}(u_i)^{\mathrm{ID}_i}=g_2^{F(\mathrm{ID})}g^{J(\mathrm{ID})}$ 和 $F_{W,2}(M)=w_0\prod_{j=1}^{n}(w_j)^{M_j}=g_2^{K(M)}g^{L(M)}$。

询问　C 回答 A 发起的一系列如下询问。

1) extract-query：为了响应 A 请求的关于身份 ID 的秘密密钥询问，C 维持一个初始化为空的列表 tsk。如果 $F(\mathrm{ID})=0 \bmod p$，C 退出模拟；否则，C 执行如下操作。

① 在 BT 上随机选择一个空的叶子节点 η，并在 η 中保存 ID。

② 对于每个节点 $\theta\in\mathrm{path}(\eta)$，随机选择 $g_\theta\in G$，并在 θ 中保存 g_θ。如果 tsk 中不存在 $(\theta,r_\theta,\mathrm{ID})$，随机选取 $r_\theta\in Z_p$，将 $(\theta,r_\theta,\mathrm{ID})$ 添加到 tsk 中。然后提取 r_θ，计算 $\mathrm{sk}_\theta=(\mathrm{sk}_{\theta,1},\mathrm{sk}_{\theta,2})=(g_\theta g_1^{-\frac{J(\mathrm{ID})}{F(\mathrm{ID})}}F_{W,1}(\mathrm{ID})^{r_\theta}$，$g_1^{-\frac{1}{F(\mathrm{ID})}}g^{r_\theta})$。

③ 发送秘密密钥 $\mathrm{sk}_{\mathrm{ID}}=\{(\theta,\mathrm{sk}_\theta)\}_{\theta\in\mathrm{path}(\eta)}$ 给 A。

2) KeyUp-query：为了响应 A 请求的关于时间周期 t 的更新密钥询问，C 维持一个初始化为空的列表 T_{uk}，如果 $t>T$，输出 $\perp$；否则，C 执行如下操作。

① 对于每个节点 $\theta\in\mathrm{KUNode}(\mathrm{BT},\mathrm{RL},t)$，首先在 θ 中提取 g_θ，然后在 T_{uk} 中提取 s_θ。如果 T_{uk} 中不存在 (θ,s_θ,t)，则选择一个随机数 $s_\theta\in Z_p$，将 (θ,s_θ,t) 添加到 T_{uk} 中，并计算 $\mathrm{uk}_\theta=(\mathrm{uk}_{\theta,1},\mathrm{uk}_{\theta,2})=(g_\theta^{-1}(v'v^t)^{s_\theta},g^{s_\theta})$。

② 将更新密钥 $\mathrm{uk}_t=\{(\theta,\mathrm{uk}_\theta)\}_{\theta\in\mathrm{KUNode}(\mathrm{BT},\mathrm{RL},t)}$ 发送给 A。

3) SKGen-query：为了响应 A 请求的关于 (ID,t) 的签名密钥询问，C 维持一个初始化为空的列表 T_{dk}。C 首先询问关于 ID 的 extract-query 和关于 t 的 KeyUp-query，分别获得相应的秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$ 与更新密钥 uk_t；然后在 T_{dk} 中提取 $(r_{\mathrm{ID}},s_{\mathrm{ID}})$。如果 T_{dk} 中不存在 $(r_{\mathrm{ID}},s_{\mathrm{ID}},\mathrm{ID},t)$，则选择 2 个随机数 r_{ID}、$s_{\mathrm{ID}}\in Z_p$，将 $(r_{\mathrm{ID}},s_{\mathrm{ID}},\mathrm{ID},t)$ 添加到 T_{dk} 中；最后运行算法 SKGen $(\mathrm{pp},\mathrm{sk}_{\mathrm{ID}},\mathrm{uk}_t)$，将生成的签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$ 返回给 A。

4) ReKey-query：对于 A 请求的关于 2 个身份 $(\mathrm{ID}_A,\mathrm{ID}_B)$ 和时间周期 t 的重签名密钥询问，C 首先询问关于 (ID_A,t) 和 (ID_B,t) 的 SKGen-query，分别获得对应的签名密钥 $\mathrm{dk}_{A,t}$ 与 $\mathrm{dk}_{B,t}$；然后运行算法 ReKey($\mathrm{pp},\mathrm{dk}_{A,t},\mathrm{dk}_{B,t}$)，将生成的重签名密钥 $\mathrm{rk}_{A\to B,t}$ 发送给 A。

5) sign-query：对于 A 请求的关于身份 ID、时间周期 $t(t\leqslant T)$ 与消息 M 的签名询问，如果 $F(\mathrm{ID})\neq 0 \bmod p$，$C$ 询问关于 (ID,t) 的 SKGen-query 获得签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$，然后运行算法 sign($\mathrm{pp},\mathrm{dk}_{\mathrm{ID},t},t,M$)，并将输出的关于 M 的签名 σ 返回给 A。如果 $F(\mathrm{ID})=0 \bmod p$，则考虑以下 2 种情况。

① 如果 $K(M)=0 \bmod p$，则 C 退出模拟。

② 如果 $K(M)\neq 0 \bmod p$，则 C 在 T_{sk}、T_{uk}、T_{dk} 中提取 r_θ、s_θ 和 $(r_{\mathrm{ID}},s_{\mathrm{ID}})$，并随机选择 $r_m\in Z_p$，计算 $\sigma_1=F_{W,1}(\mathrm{ID})^{r_\theta+r_{\mathrm{ID}}}(v'v^t)^{s_\theta+s_{\mathrm{ID}}}g_1^{-\frac{L(M)}{K(M)}}F_{W,2}(M)^{r_m}$、$\sigma_2=g^{r_\theta+r_{\mathrm{ID}}}$、$\sigma_3=g^{s_\theta+s_{\mathrm{ID}}}$ 和 $\sigma_4=g_1^{-\frac{1}{K(M)}}g^{r_m}$，然后将关于 M 的签名 $\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4)$ 发送给 A。

6) revoke-query：对于 A 请求的关于 (ID,t) 的撤销询问，C 运行算法 revoke($\mathrm{ID},t,\mathrm{RL},\mathrm{st}$)，并将运行结果返回给 A。

伪造　攻击者 A 最后输出一个对应于身份 ID^* 和时间周期 $t^*\leqslant T$ 的关于消息 M^* 的签名 $\sigma^*=(\sigma_1^*,\sigma_2^*,\sigma_3^*,\sigma_4^*)$。如果 $F(\mathrm{ID}^*)\neq 0 \bmod p$ 或 $K(M^*)\neq 0 \bmod p$，C 退出模拟；否则，C 计算 CDH 值 gab 过程如下

$$\frac{\sigma_1^*}{(\sigma_2^*)^{J(\mathrm{ID}^*)}(\sigma_3^*)^{v_0+v_1t^*}(\sigma_4^*)^{L(M^*)}}=$$

$$\frac{g_2^aF_{W,1}(\mathrm{ID}^*)^{r_{\mathrm{ID}}^*+r_\theta^*}(v'v^{t^*})^{s_{\mathrm{ID}}^*+s_\theta^*}F_{W,2}(M^*)^{r_m^*}}{(g^{r_{\mathrm{ID}}^*+r_\theta^*})^{J(\mathrm{ID}^*)}(g^{s_{\mathrm{ID}}^*+s_\theta^*})^{v_0+v_1t^*}(g^{r_m^*})^{L(M^*)}}=$$

$$\frac{g_2^a(g_2^{F(\mathrm{ID}^*)}g^{J(\mathrm{ID}^*)})^{r_{\mathrm{ID}}^*+r_\theta^*}(g^{v_0}g^{v_1t^*})^{s_{ID}^*+s_\theta^*}(g_2^{K(M^*)}g^{L(M^*)})^{r_m^*}}{(g^{J(\mathrm{ID}^*)})^{r_{\mathrm{ID}}^*+r_\theta^*}(g^{v_0+v_1t^*})^{s_{\mathrm{ID}}^*+s_\theta^*}(g^{L(M^*)})^{r_m^*}}=$$

$$g_2^a=g^{ab}$$

其中，$F(\mathrm{ID}^*)=K(M^*)=0 \bmod p$。

因此，C 利用 A 的伪造 $(\mathrm{ID}^*,t^*,M^*,\sigma^*)$ 成功求

解了 CDH 问题实例。与文献[3-4]的分析过程相似，C 将以 $\varepsilon' > \dfrac{\varepsilon}{16(m+1)(n+1)q_{\mathrm{s}}(q_{\mathrm{sk}}+q_{\mathrm{dk}}+q_{\mathrm{rk}}+q_{\mathrm{s}})}$ 的概率成功解决 G 上的 CDH 问题。

定理 2 本文方案满足抗签名密钥泄露攻击性。

证明 在 4.1 节的方案中，分别用 g_θ 和 $\tilde{g}_\theta$ 来构造秘密密钥和更新密钥，并且满足 $g_\theta\tilde{g}_\theta = g_2^\alpha$。只有未撤销的用户才能正确恢复出 g_2^α，进而生成合法的签名密钥。如果攻击者获得了未撤销用户 ID 的签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$，则利用公开信道传输的更新密码 uk_t 计算

$$\mathrm{sk}'_{\theta,1} = \frac{\mathrm{dk}_{\mathrm{ID},t,1}}{\mathrm{uk}_{\theta,1}} = \frac{g_2^\alpha F_{W,1}(\mathrm{ID})^{r_\theta+r_{\mathrm{ID}}}(v'v^t)^{s_\theta+s_{\mathrm{ID}}}}{(\tilde{g}_\theta)^\alpha(v'v^t)^{s_\theta}} =$$

$$g_\theta^\alpha F_{W,1}(\mathrm{ID})^{r_\theta+r_{ID}}(v'v^t)^{s_{\mathrm{ID}}}$$

很显然有 $\mathrm{sk}'_{\theta,1} \neq \mathrm{sk}_{\theta,1} = g_\theta^\alpha F_{W,1}(\mathrm{ID})^{r_\theta}$。

由于在本文方案的 SKGen 算法中，用户随机选取 $r_{\mathrm{ID}}, s_{\mathrm{ID}} \in Z_p$ 对秘密密钥和更新密钥进行了随机化处理，因此攻击者即使获得了用户身份 ID 在时间周期 t 的更新密钥 $\mathrm{dk}_{\mathrm{ID},t}$，但仍然无法直接从 $\mathrm{dk}_{\mathrm{ID},t}$ 中计算出对应的秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$。因为用户的签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$ 通过固定的秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$ 和定期变化的更新密钥 uk_t 生成，所以攻击者在 $\mathrm{sk}_{\mathrm{ID}}$ 未知的情况下不能利用 SKGen 算法产生其他时间周期 $\tilde{t} \neq t$ 的合法签名密钥 $\mathrm{dk}_{\mathrm{ID},\tilde{t}}$。

综上所述，即使攻击者获得当前时间段的签名密钥 $\mathrm{dk}_{\mathrm{ID},t}$，攻击者无法通过 $\mathrm{dk}_{\mathrm{ID},t}$ 和公开的更新密钥 uk_t 计算出秘密密钥 $\mathrm{sk}_{\mathrm{ID}}$，也不会影响其他时间周期签名密钥 $\mathrm{dk}_{\mathrm{ID},\tilde{t}}$ 的安全性。因此，本文方案能抵抗签名密钥泄露攻击，即满足抗签名密钥泄露攻击性。

4.4 性能分析

文献[4-6]分别提出了 3 个标准模型下安全的基于身份的代理重签名方案（分别将其命名为 Shao 方案[4]、Feng 方案[5]和 Hu 方案[6]），将本文方案与这些方案进行性能比较分析，如表 2 所示。令 N 表示用户总数，R 表示撤销的用户数，则由 KUNode 算法的计算复杂度可知[25]，本文方案中 PKG 更新密钥的开销为 $O(R\log\dfrac{N}{R})$。假设所有方案选择阶为素数 p 的群 G 和 GT，仅考虑计算开销比较大的双线性对和指数运算，不再讨论计算量较小的乘法运算等操作。在表 2 中，符号|G|表示群 G 中一个元素的平均长度，|p|表示有限域 Zp 中一个元素的平均长度，Pa 表示一次双线对运算，exp 表示一次指数运算。

从表 2 可知，与 Shao 方案[4]相比，本文方案的签名长度和重签名长度多了一个群 G 中的元素，并且签名验证算法多了一次双线性对运算和一次指数运算。与 Feng 方案[5]相比，本文方案与该方案有相同的重签名长度，但具有较高的签名验证效率，并满足多用性。与 Hu 方案[6]相比，本文方案具有更短的签名长度和重签名长度，并且签名算法的计算效率优于该方案。然而，对比的 3 种方案均没有考虑用户撤销问题，本文方案实现了用户撤销功能，并且 PKG 撤销用户的工作量随用户总数的增加呈对数增长，具有良好的延展性。

将本文方案、Feng 方案[5]和 Hu 方案[6]进行签名算法的计算时间开销实验对比分析，具体结果如图 2 所示。选取 PBC 库的 a.param 初始化 pairing，实验的硬件环境为：2.5 GHz 英特尔酷睿 i7-6500 处理器，8 GB 的内存，512 GB 的硬盘空间；软件环境为：64 位 Windows 10 操作系统，密码库 PBC-0.4.7-VC。

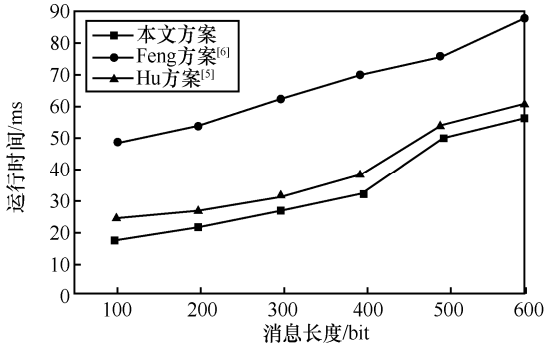

图 2 签名生成的时间开销与消息长度关系

表 2 **4 种方案的计算开销与安全性能比较**

方案	签名长度	重签名长度	签名	重签名	签名验证	抗签名密钥泄露攻击	用户撤销功能
Shao 方案[4]	3|G|	3|G|	2exp	4Pa +2exp	4Pa	否	否
Feng 方案[5]	3|G|	4|G|	3exp	4Pa +5exp	5Pa+3exp	否	否
Hu 方案[6]	4|G|+|p|	4|G|+|p|	6exp	4Pa+6exp	4Pa+4exp	否	否
本文方案	4|G|	4|G|	2exp	5Pa+3exp	5Pa+exp	是	是

由于本文方案是基于 Shao 方案[4]设计的，因此 2 种方案生成签名的计算开销相同，均需要 2 次指数运算。此外，Feng 方案[5]需要 3 次指数运算，Hu 方案[6]需要 6 次指数运算。对于长度相同的签名消息，图 2 表明本文方案生成签名的时间开销低于 Feng 方案[5]和 Hu 方案[6]。

本文方案利用 KUNode 算法[26]来实现用户的撤销，而文献[16]通过 PKG 直接更新未撤销用户的密钥来实现用户的撤销。下面将 2 种方案进行密钥更新的性能比较，结果如图 3 所示。假设用户总数为 32 个，被撤销的用户数分别为 1 个、2 个、4 个、8 个和 16 个。

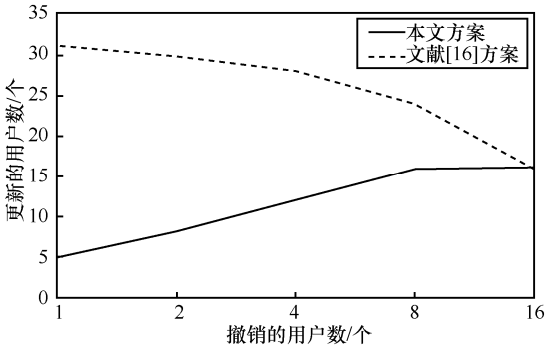

图 3　密钥更新的性能比较

本文方案和文献[16]方案生成每个用户的更新密钥均需要执行 2 次指数运算，所需的时间开销约为 15 ms。由图 3 可知，当被撤销的用户数小于用户总数的一半时，本文方案需要更新密钥的用户个数小于文献[16]方案。因此，本文方案具有更低的用户撤销开销。

5　结束语

本文基于 Shao 方案[4]和 KUNode 算法[26]，构造了一种可撤销的基于身份的代理重签名方案，并在标准模型下证明了其安全性依赖于 CDH 假设。本文方案支持用户撤销功能，满足双向性和多用性，可有效抵抗签名密钥泄露攻击，具有良好的延展性。但本文方案无法抵抗量子计算攻击，下一步的任务是设计格上可撤销的基于身份的代理重签名方案。

参考文献：

[1] YANG T, YU B, WANG H, et al. Cryptanalysis and improvement of Panda-public auditing for shared data in cloud and internet of things[J]. Multimedia Tools and Applications, 2017, 76(19): 19411-19428.

[2] SOOKHAK M, GANI A, KHAN M K, et al. Dynamic remote data auditing for securing big data storage in cloud computing[J]. Information Sciences, 2017, 380: 101-116.

[3] WATERS B. Efficient identity-based encryption without random oracles[C]//The 24th Annual International Conference on The Theory and Application of Cryptographic Techniques. IACR, 2005: 114-127.

[4] SHAO J, CAO Z, WANG L, et al. Proxy re-signature schemes without random oracles[C]//The 8th International Conference on Cryptology. Springer, 2007: 197-209.

[5] FENG J, LAN C, JIA B. ID-based proxy re-signature scheme with strong unforgeability[J]. Journal of Computer Applications, 2014, 34(11): 3291-3294.

[6] HU X, ZHANG Z, YANG Y. Identity based proxy re-signature schemes without random oracle[C]//Computational Intelligence and Security. 2009: 256-259.

[7] SHAO J, WEI G, LING Y, et al. Unidirectional identity-based proxy re-signature[C]//IEEE International Conference on Communications. 2011: 1-5.

[8] HUANG P, YANG X, YAN L I, et al. Identity-based proxy re-signature scheme without bilinear pairing[J]. Journal of Computer Applications, 2015, 35(6):1678-1682.

[9] JIANG M M, HU Y P, WANG B C, et al. Identity-based unidirectional proxy re-signature over lattice[J]. Journal of Electronics & Information Technology, 2014, 36(3): 645-649.

[10] TIAN M M. Identity-based proxy re-signatures from lattices[J]. Information Processing Letters, 2015, 115(4): 462-467.

[11] CANETTI R, GOLDREICH O, HALEVI S. The random oracle methodology, revisited[J]. Journal of the ACM, 2004, 51(4): 557-594.

[12] BONEH D, FRANKLIN M. Identity-based encryption from the weil pairing[C]// Advances in CRYPTO.2001: 213-229.

[13] BOLDYREVA A, GOYAL V, KUMAR V. Identity-based encryption with efficient revocation[C]//ACM Conference on Computer and Communications Security. 2008: 417-426.

[14] LEE K, LEE D H, PARK J H. Efficient revocable identity-based encryption via subset difference methods[J]. Designs, Codes and Cryptography, 2017, 85(1): 39-76.

[15] ZHANG L, SUN Z, MU Y, et al. Revocable hierarchical identity-based encryption over lattice for pay-tv systems[J]. International Journal of Embedded Systems, 2017, 9(4): 379-398.

[16] TSAI T T, TSENG Y M, WU T Y. Provably secure revocable ID-based signature in the standard model[J]. Security and Communication Networks, 2013, 6(10): 1250-1260.

[17] LIU Z, ZHANG X, HU Y, et al. Revocable and strongly unforgeable ID-based signature scheme in the standard model[J]. Security and Communication Networks, 2016, 9(14): 2422-2433.

[18] JIA X, HE D, ZEADALLY S, et al. Efficient revocable ID-based signature with cloud revocation server[J]. IEEE Access, 2017, 5: 2945-2954.

[19] YANG X, YANG P, AN F, et al. Cryptanalysis and improvement of a strongly unforgeable identity-based signature scheme[C]//International Conference on Information Security and Cryptology. Springer. 2017: 196-208.

[20] ZHAO J, WEI B, SU Y. Communication-efficient revocable identity-based signature from multilinear maps[J]. Journal of Ambient Intelligence and Humanized Computing, 2019, 10(1): 1-12.

[21] WEI J, HUANG X, HU X, et al. Revocable threshold attribute-based signature against signing key exposure[C]// International Conference on Information Security Practice and Experience. 2015: 316-330.

[22] ZHENG Q, LI Q, AZGIN A, et al. Data verification in information-centric networking with efficient revocable certificateless signature[C]// IEEE Conference on Communications and Network Security. IEEE, 2017: 1-9.

[23] HUNG Y H, TSENG Y M, HUANG S S. Lattice-based revocable certificateless signature[J]. Symmetry, 2017, 9(10): 242-259.

[24] XU S, YANG G, MU Y. A new revocable and re-delegable proxy signature and its application[J]. Journal of Computer Science and Technology, 2018, 33(2): 380-399.

[25] WEI J, LIU W, HU X. Forward-secure identity-based signature with efficient revocation[J]. International Journal of Computer Mathematics, 2017, 94(7): 1390-1411.

[26] NAOR D, NAOR M, LOTSPIECH J. Revocation and tracing schemes for stateless receivers[C]//The 21st Annual International Cyptology Conference. IACR, 2001: 41-62.

[作者简介]

杨小东（1981− ），男，甘肃甘谷人，博士，西北师范大学副教授，主要研究方向为代理重签名和云计算安全。

李雨潼（1994− ），男，甘肃兰州人，西北师范大学硕士生，主要研究方向为应用密码学与车联网安全。

王晋利（1993− ），女，山西泽州人，西北师范大学硕士生，主要研究方向为信息安全理论与技术。

麻婷春（1992− ），女，甘肃武威人，西北师范大学硕士生，主要研究方向为大数据安全。

王彩芬（1963− ），女，河北安国人，博士，西北师范大学教授、博士生导师，主要研究方向为密码协议和网络编码。