标准模型下 CCA 匿名性的失败停止属性基群签名方案

廖东旭 [1], 程小刚 [1,2]

[1] 华侨大学 计算机科学与技术学院 厦门 中国 361021
[2] 华侨大学 厦门市数据安全与区块链技术重点实验室 厦门 中国 361021

摘要 随着网络技术的普及, 个人隐私和信息安全的保护已成为全球关注的问题。群签名技术允许群组中的成员在保持签名者匿名的同时, 代表整个群组进行签名, 并在必要时能够追踪签名者的身份。因此, 该技术在电子投票和匿名认证等领域具有广泛的应用。然而, 现有的群签名方案在成员属性的动态管理、安全性以及抵抗拥有无限计算能力的攻击者方面存在局限性。为了解决这些问题, 本文提出了一种基于属性的失败停止群签名方案。方案通过 Groth-Sahai 证明系统和可验证加密技术在标准模型下实现了 CCA(Chosen Ciphertext Attack)匿名性, 确保即使在强大对手的存在下也能提供高水平的安全性。方案还引入了失败停止签名技术, 进一步增强了安全性, 通过检测和停止未经授权的操作, 防止潜在的漏洞和系统滥用。方案支持成员属性的动态管理和撤销, 允许管理员根据需要高效地更新和撤销成员凭证, 从而保证群签名系统的长期安全。此外, 群组中允许管理员追踪签名属性集, 确保只有授权的属性用于签名过程。方案还使群组成员能够提供证据, 证明在遭受攻击时应停止该方案以防止敌手的进一步的恶意行动。本文还提出了一个简化方案, 该方案仅提供 CPA(Chosen Plaintext Attack)匿名性, 但保留了原方案的核心功能, 提供了安全性和效率之间的平衡, 适用于计算资源有限的设备。最后, 通过与类似方案的比较分析, 展示了本文方案在安全性和功能性方面的优势。本文所提出的方案不仅增强了对拥有无限计算能力的对手的安全性, 还具有动态管理成员属性的功能, 使其在需要兼顾安全性和效率的实际应用场景中具有可行性。

关键词 属性基群签名; 失败停止签名; Groth-Sahai 证明系统; 标准模型; CCA 匿名性; 动态聚合器
中图法分类号 TP309 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2026.01.06

Fail-Stop Attribute-Based Group Signature Scheme with CCA Anonymity in Standard Model

LIAO Dongxu[1], CHENG Xiaogang[1,2]

[1] College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China
[2] Xiamen Key Laboratory of Data Security and Blockchain Technology, Huaqiao University, Xiamen 361021, China

Abstract Due to the proliferation of network technology, the protection of personal privacy and information security has become a global concern. Group signature technology allows a member of a group to sign on behalf of the entire group while maintaining the anonymity of the signer, enabling the traceability of the signer's identity when necessary. Therefore, it has a wide range of applications in fields such as electronic voting and anonymous authentication. However, existing group signature schemes have limitations in terms of dynamic management of member attributes, security, and resistance to attackers with unlimited computational power. To solve these problems, this paper proposes a fail-stop attribute-based group signature scheme. The scheme achieves CCA (Chosen Ciphertext Attack) anonymity under the standard model through the Groth-Sahai proof system and verifiable encryption technology, ensuring that it can provide a high level of security even in the presence of powerful adversaries. It also introduces the fail-stop signature to further enhance security by enabling the detection and halting of unauthorized actions, thus preventing potential breaches and misuse of the system. The scheme supports dynamic management and revocation of member attributes, allowing administrators to efficiently update and revoke member credentials as needed, ensuring the security of the group signature system over time. Additionally, it allows administrators to trace the signature attribute set, ensuring that only authorized attributes are used in the signing process. The scheme also enables group members to provide evidence that the scheme should be stopped in the event of an attack to prevent further malicious actions by the adversary. This paper also proposes a simplified scheme that only provides CPA (Chosen Plaintext Attack) anonymity while retaining the core functionality of the original scheme, offering a balance between security and efficiency, making it suitable for applications with limited computational resources.

通讯作者: 程小刚, 博士, 副教授, Email: cxg@hqu.edu.cn。
本课题得到福建省社会科学规划项目(No. FJ2024B088)、福建省高校以马克思主义为指导的哲学社会科学学科基础理论研究项目(No. FJ2024MGCA028)、教育部人文社会科学研究专项项目(No. 24JD710008)资助。
收稿日期: 2024-05-23; 修改日期: 2024-07-18; 定稿日期: 2025-11-11

Finally, the advantages of this paper's scheme in terms of security and functionality are demonstrated through comparative analysis with similar schemes. The proposed scheme not only enhances the security against opponents with unlimited computing power, but also has the function of dynamically managing member attributes, making it feasible in practical application scenarios that require both security and efficiency.

Key words attribute-based group signatures; fail-stop signatures; Groth-Sahai proof system; standard model; CCA-anonymity; dynamic accumulators

1 引言

在数字化时代的背景下, 网络技术的广泛普及和数据交换的日益频繁使得个人隐私保护与信息安全成为全球性的焦点问题, 特别是在多方参与和身份认证的场景中, 如何在保护用户隐私的同时确保交易和通信的安全, 已成为密码学领域的核心研究议题之一。在此背景下, 群签名技术作为一种重要的密码学工具, 它允许群成员代表整个群组进行签名, 同时保持签名者的匿名性, 同时在必要时可以由特定权威机构揭示签名者的真实身份, 因此群签名在电子投票、匿名认证等领域得到了广泛应用。

随着研究开展, 传统的群签名方案暴露出一些局限性, 方案不支持根据成员属性构建细粒度的签名策略, 从而制约了其应用场景的扩展。在这一背景下, Khader[1]基于属性基密码学的思想提出了属性基群签名, 成员证书由身份证书和属性证书组成, 能较好地制定灵活的签名策略, 但早期的方案并未考虑成员属性的动态变化需求, 并且早期方案普遍构建在随机预言机模型下, 在安全性上存在缺陷, 同时难以有效抵御外部攻击。此外, 随着量子计算机研究的推进, 传统的群签名方案面临着越来越严峻的安全挑战, 目前尚缺乏一种可以有效证明方案不再安全的手段。因此, 提出一个既能适应成员属性动态变化, 又能提供充分安全保障的属性基群签名方案, 成为当务之急。

为应对上述挑战, 本文提出了一种新的属性基群签名方案。该方案在标准模型下采用 Groth-Sahai 证明系统[2]与基于变色龙哈希函数的可验证加密技术[3]进行构造, 实现了 CCA 匿名性, 方案通过动态聚合器[4]支持成员属性的动态管理和撤销。同时, 方案引入了失败停止签名技术[5], 使方案能够抵御无限计算能力敌手的攻击, 在敌手攻击导致方案安全性受到威胁时, 成员能够提供证据证明方案已不再安全, 从而触发失败停止机制, 为群签名方案的安全性和实用性提供了新的保障。

1.1 相关工作

群签名技术最初由 Chaum 和 van Heyst[6]提出, 主要满足中心化匿名性应用场景的需求, 早期的方案普遍存在效率不高和功能单一的问题, 这限制了实际应用范围。在此基础上, Bellare 等[7]首次提出适用于成员动态变更场景的 BSZ05 安全模型, 该模型的安全性质主要围绕匿名性、可追踪性和不可陷害性展开, 通过证明这 3 个安全性质, 可以确认方案的整体安全性。Boyen 和 Waters[8]依托二级签名和子群判定假设, 在标准模型下提出了 BW07 方案, 方案在签名和验证的计算开销方面表现良好, 但仅实现了 CPA 匿名性, 并且不支持成员动态变更。进一步地, Groth[9]基于证书签名和标签加密技术, 构造出了具有 CCA 匿名性的群签名方案, 但在签名阶段需要对成员证书进行随机化处理, 这使得证书管理较为困难, 此外, 该方案还采用了一次签名技术, 每次签名都需要使用新的签名秘钥进行消息签名, 从而增加了签名的开销。为解决上述问题, 岳笑含等[10]利用可验证加密技术对身份证书进行加密, 避免了证书随机化的问题, 并且提出的方案在通信开销和计算开销方面都表现优秀。Cheng 等[4]在标准模型下模拟了随机预言机模型中的动态聚合器, 撤销开销由管理者承担, 成员只需在签名时更新其证书即可, 这种方法具有较好的通用性, 但实现撤销成员的代价是群公钥的长度与撤销列表的大小呈线性关系。

由于传统群签名方案无法适应更复杂的应用需求, Khader[1]提出了属性基群签名方案, 将身份基群签名与属性基密码学相结合, 群管理员(Group Manager, GM)能够根据成员的属性制定出细粒度的签名策略, 仅当成员属性符合该策略时成员才能执行签名, 这种方式可以适应更为复杂的应用场景。按照对哈希函数的定义, 属性基群签名方案可分为两类: 一类是将哈希函数视为理想的随机预言机模型[1,11-12], 在这一模型下, 哈希函数被假设为完全随机且均匀分布的, 实际上并不存在这种哈希函数; 另一类是标准模型[13-16], 在此模型中哈希函数被视为现实可实现的, 即仅满足抗碰撞性。如果按照访问结构进行分类, 又可以分为访问树结构[1,11-14]和访问矩阵[15-16]两类, 与访问树相比, 访问矩阵的优势在于秘密的分享和恢复都在矩阵上进行, 且在恢复根秘密时不需要递归计算, 因此更适合于构建基于属性的密码方案。接下来我们以第一个分类对属性基群签名进

行综述。

在随机预言机模型的方案中, Emura 等[11]着重解决了群成员的动态加入问题和实现 CCA 匿名性, 他们的方案允许成员通过与 GM 执行交互协议成为合法群成员并获得属性证书, 签名时使用 Cramer-Shoup 加密方案对身份证书进行加密从而保证 CCA 匿名性, 方案还提出了动态变更访问结构的方法, 但这种方法仅限于将 (t,n) 门限扩大为 AND 门限, 具有一定的局限性。Patel 和 Jinwala[12]将工作重点放在实现属性匿名性上, 方案中管理员需要在访问树中使用基于 RSA 的不经意签名封装协议对属性进行封装, 从访问树中自底向上计算根秘密, 封装后的属性是不可区分的, 符合访问结构的群成员能够解开封装并恢复出正确的根秘密进行签名, 方案在验证算法上表现良好, 但给 GM 带来了额外负担。在标准模型的方案中, Qian 和 Zhao[13]在 BW07[8]的基础上提出了第一个标准模型下的属性基群签名方案, 考虑了成员的动态加入和撤销, 但方案中计算开销较大且存在难以实现的二次配对映射 $\hat{e}:G_T\times G_T\to\widehat{G_T}$, 因此方案主要具有理论价值。Ali 和 Amberker[14]使用 Groth-Sahai 证明系统[2]提出了一个计算开销和签名大小都是 $O(1)$ 的属性基群签名方案, 并严格证明了方案中与属性相关的安全性质, 但是方案在 BMW03[17]安全模型下构建, 没有考虑到群组动态变化的情景; Li 等[15]根据标签加密在标准模型下提出了具有 CCA 匿名性的方案, 虽然实现了更强的匿名性, 但与 Ali 和 Amberker 的方案存在同样的问题。许玉岚等[16]对追踪签名使用的属性集进行了研究, 方案中追踪属性的代价是在签名阶段存在一定的通信开销及追踪阶段使用到了昂贵的配对计算, 这限制了属性集追踪的效率。

最近, Chen 等[18]在随机预言机模型中引入了失败停止签名技术, 提出了首个失败停止群签名方案, 群成员通过三方握手协议来加入群组, 即使面对无限计算能力的敌手, 敌手伪造出的私钥与实际签名者私钥相等的概率仍是可忽略的, 这意味着该方案为签名者提供了无条件的安全保障, 然而方案实现的功能存在局限性, 并不支持细粒度的签名策略, 通过结合属性基群签名与失败停止签名, 有望满足更为复杂的实际应用场景的需求并实现更好的安全性。

1.2 本文贡献

在上述工作的基础上, 本文的主要贡献可以概括为以下几个方面:

(1) 本文在标准模型下构造了具有 CCA 匿名性的失败停止属性基群签名方案, 利用失败停止签名技术, 本方案为签名者提供无条件安全性, 能在遭遇具有无限计算能力敌手的攻击时提供阻止其进一步恶意行为的证据。

(2) 通过使用线性秘密分享方案进行属性分发, 与访问树结构相比, 我们的方案在属性分发和秘密恢复方面更为高效。此外, 本文方案还在标准模型下使用动态聚合器实现了对成员属性的动态撤销, 以满足群组中成员属性需要动态变更的应用场景的需要。

(3) 采用基于变色龙哈希函数的可验证加密技术实现了 CCA 匿名性, 能够抵抗比 CPA 更强的攻击。考虑到方案的效率和签名大小, 本文还对方案进行了简化, 简化后的方案只实现了 CPA 匿名性, 但展现出较好的性能。

(4) 通过对本文方案进行性能分析和同类方案的比较, 结果表明本文方案在功能性和安全性上优于现有同类方案, 具有较高的实用性和可行性。

2 预备知识

2.1 线性秘密分享方案

线性秘密分享方案[19](Linear Secret Sharing Schemes, LSSS)即是对于实体集合 P 上的秘密分享方案 $\prod$, 存在经过特定算法生成的一个 $|\Gamma|\times col$ 的访问矩阵 $\boldsymbol{M}$ 用于描述访问结构, 即具有哪些属性能够恢复秘密, 其中 Γ 是属性全集, col 取决于采用的线性秘密分享算法。设置 s_T 为需要分享的秘密, 随机选取 $col-1$ 个随机整数 $r_i\in_R Z_p^*$, 其中 $i=2,3,...,col$, 得到用于分享秘密向量 $\vec{\boldsymbol{y}}=(s_T,r_2,r_3,...,r_{col})$, 生成子秘密向量 $\vec{s}=\boldsymbol{M}\times\vec{\boldsymbol{y}}^{\mathrm{T}}=(s_1,s_2,...,s_{|\Gamma|})$ 用于颁发子秘密 s_i 给参与者 p_i, 此时存在向量 $\vec{\boldsymbol{\omega}}$ 使符合访问结构的授权集合 $P_j\subseteq P$ 能够计算 $\sum_{p_i\in P_j}\vec{s}_i\cdot\vec{\omega}_i^{\mathrm{T}}=s_T$, 满足 $\hat{\boldsymbol{M}}\cdot\vec{\omega}^{\mathrm{T}}=(1,0,0,...,0)$, 其中 $\hat{\boldsymbol{M}}$ 是由所有参与者 $p_i\in P_j$ 的子秘密 s_i 在访问矩阵 $\boldsymbol{M}$ 中对应的行向量 $\vec{\boldsymbol{M}}_i$ 组成的子矩阵, 这种性质被称为线性重构性质。

2.2 双线性配对

双线性配对即是在阶为大素数 p 的乘法循环群 G_1,G_2,G_T 上, 存在映射 $e:G_1\times G_2\to G_T$ 具有以下性质:

1) 双线性: 对于任意生成元 $g_1\in G_1,g_2\in G_2$, $a,b\in Z_p^*$, 有 $e(g_1^a,g_2^b)=e(g_1,g_2)^{ab}$。

2) 非退化性: 对于 G_1,G_2 的任意生成元 g_1,g_2,

映射不会映射到群 G_T 的单位元上, 即 $e(g_1,g_2)\neq 1\in G_T$。

3) 可计算性: 对于 G_1,G_2 的任意生成元 g_1,g_2, 配对 $e(g_1,g_2)$ 总是存在一个有效的算法进行计算。

对于 $G_1=G_2$ 的配对称为对称配对, 反之则称为非对称配对, 本文方案构建在对称配对上, 此时 $G=G_1=G_2$。对于阶为 p 的乘法循环群 G,G_T, 下文出现的 G,G_T 和 p 均表示相同的含义, 故不再进行赘述。

2.3 Groth-Sahai 证明系统

Groth-Sahai 证明系统[2]是标准模型下第一个高效的无交互零知识证明系统, 被用于构建大量标准模型下的密码方案, 在文中提出了基于线性判定假设的实例, 通过公共引用字符串(Common Reference String, CRS)对变量进行承诺, 承诺与计算得到的证明实现了对配对乘积等式的证明, 并且存在陷门 (α,β) 可以打开承诺提取变量。Groth-Sahai 证明系统的安全性依赖于 CRS 的完全隐藏设置和完全绑定设置在计算上是不可区分的, 能够满足群签名匿名性的要求, 下面简单介绍实例中对变量的承诺过程以及本文方案中使用到的定义:

首先选择生成元 $g\in_R G$, 选择整数 $r,s,\alpha,\beta\in_R Z_p^*$, 生成公共引用字符串 $CRS=(\vec{\mathbf{f}}_1,\vec{\mathbf{f}}_2,\vec{\mathbf{f}}_3)$, 其中 $g_1=g^{\alpha},g_2=g^{\beta},\vec{\mathbf{f}}_1=(g_1,1,g),\vec{\mathbf{f}}_2=(1,g_2,g),\vec{\mathbf{f}}_3=\vec{\mathbf{f}}_1^{\,r}\odot\vec{\mathbf{f}}_2^{\,s}$, 1 是群 G 的单位元, "$\odot$"表示 Hadamard 乘积。

定义 $\iota(X)=(1,1,X)\in G^3$, $Comm(X)=\iota(X)\odot\vec{\mathbf{f}}_1^{\,r_X}\odot\vec{\mathbf{f}}_2^{\,s_X}\odot\vec{\mathbf{f}}_3^{\,t_X}$ 表示对群 G 中的群元素 X 进行承诺, $comm(x)=\vec{\mu}^{x}\odot\vec{\mathbf{f}}_1^{\,r_x}\odot\vec{\mathbf{f}}_2^{\,s_x}$ 表示对群 $Z_p^{\ *}$ 上的群元素 x 进行承诺, 其中 $r_X,s_X,t_X,r_x,s_x\in_R Z_p^*$, $\vec{\mu}=\iota(g)\odot\vec{\mathbf{f}}_3$; 定义双线性映射 $E:G^3\times G^3\to G_T^9$, 令 $\hat{\iota}_T(X)=E(\vec{\mu},\iota(X))$; 定义 $\pi=\mathrm{NIWI}\{(X,Y):e(X,Y)\ =t_T\}$ 表示对配对乘积等式 $e(X,Y)=t_T$ 的非交互式证据不可区分(Non-Interactive Witness-Indistinguishable, NIWI)证明, $\theta=\mathrm{NIZK}\{(x,y,z):g^x=g^{yz}\}$ 表示对多指数等式 $g^x=g^{yz}$ 的非交互式零知识(Non-Interactive Zero-Knowledge, NIZK)证明。更多相关细节详见 GS08[2]。

2.4 基于变色龙哈希函数的可验证加密方案

变色龙哈希函数是一种具有陷门的哈希函数, 对于掌握了陷门的实体而言, 可以在不同的消息上产生相同的哈希值, 而如果没有掌握陷门, 在计算上这是不可行的, 此时满足抗碰撞性, 本文变色龙哈希函数的陷门仅在安全证明中会使用到。Zhang[3]使用变色龙哈希函数基于线性判定假设构造出了 CCA 安全的可验证加密方案, 可以在不解密的前提下完成对密文完整性和正确性的验证, 能够很好地满足群签名匿名性和可追踪性的要求。

2.5 失败停止签名

失败停止签名是一种当签名者遭到无限计算能力的敌手伪造签名时, 签名者可以提供压倒性的证据证明自己遭到伪造以制止敌手进一步行为的签名技术, 能够给签名者提供无条件安全性。本文采用 Pedersen 和 Pfitzmann 提出的失败停止签名方案[5]:

1) KeyGen: 选择生成元 $g\in_R G$, 选择系统私钥 $gmsk=d\in_R Z_p^*$, 设置系统公钥为 $(g,g_d=g^d)$, 颁布签名私钥 $(k_1,k_2,k_3,k_4)\in_R Z_p^*$, 计算签名公钥 $(S_1=g^{k_1}g_d^{k_3},S_2=g^{k_2}g_d^{k_4})$。

2) Sign: 对于消息 $m_{fs}\in Z_p^*$, 计算 $c_1=k_1+m_{fs}k_2,c_2=k_3+m_{fs}k_4$, 输出 $\sigma=(c_1,c_2)$。

3) Verify: 对于消息签名对 (m_{fs},σ), 验证 $S_1S_2^{m_{fs}}=g^{c_1}g_d^{c_2}$ 是否成立。

4) Proof: 对于伪造的消息签名对 (m'_{fs},σ'), 被伪造者使用签名私钥对 m'_{fs} 签名得到 $\sigma=(c_1,c_2)$, 可以计算证据 $d=\log_g g_d=(c_1'-c_1)(c_2-c_2')^{-1}$ 发送给管理员以证明自己被伪造。

2.6 困难性假设

2.6.1 q-HSDH(q-Hidden Strong Diffie-Hellman)假设[8]

给定 3 个生成元. $g,g^{\gamma},h\in G$ 和 $q-1$ 个三元组 $(g^{1/(\gamma+x_i)},g^{x_i},h^{x_i})\in G^3$, 其中 $x_i\in Z_p,i=1,2,\ldots,q-1$, 输出一组与给定元组不同的三元组 $(g^{1/(\gamma+x)},g^x,h^x)\in G^3$ 是困难的。

2.6.2 DLIN(Decision Linear)假设[10]

在阶为 p 的群 G 和 G_T 中, 给定生成元 $g\in_R G$ 和五元组 $g^a,g^b,g^{ac},g^{ad},T\in G^5$, 其中 $a,b,c,d\in_R Z_p^*$, 判断 T 满足 $T=g^{c+d}$ 还是 T 是随机的是困难的。

2.6.3 ODBP(One side Double Pairing)假设[4]

在阶为 p 的群 G 和 G_T 中, 给定随机群元素 $G_r,G_s\in G^2$, 找到非平凡群元素 $R,S\in G(R,S\neq 1\in G)$ 满足 $e(G_r,R)e(G_s,S)=1$ 是困难的。

2.6.4 DL(Discrete Logarithm)假设

在阶为 p 的循环群 G 上给定二元组 (g,g^{α})，输出 α 是困难的，其中 $g\in G,\alpha\in Z_p^*$。

2.6.5 KEA1(Knowledge of Exponent Assumption 1)假设[14]

对于任意敌手 $\mathcal{A}$ 输入 (p,g,g^{α})，其中 p 是群 G 的阶数，g 是群 G 的生成元，$\alpha\in Z_p^*$，返回群 G 的群元素 (h,h^{α})，存在提取器 $\bar{\mathcal{A}}$ 对于同样的输入能够提取 ξ，使 $g^{\xi}=h$ 成立。

2.7 Groth 子协议+

Groth 在 Gro07 方案[9]中提出了一个基于 DL 假设的交互协议，通过该协议，负责发布成员证书的管理员 GM_{issu} 输出 g^x 并对于 x 保持零知识，加入群组的成员 *User* 可以获得 x。为了将其应用在 q-HSDH 假设，本文对其进行了合理的修改使管理员获得私钥 x，而群成员获得 g^x 并对于 x 保持零知识:

1) 记群 G 元素 $g,h,\Omega=h^{\gamma}$ 为公共参数，*User* 选择整数 $t\in_R Z_p^*$，发送 $(ID,T=g^t)$ 给 GM_{issu}。

2) GM_{issu} 判断 *ID* 是否唯一，若唯一则选择整数 $a,r,\eta\in_R Z_p^*$，发送 $(A=g^a,R=g^rT,D=g^{\eta})$ 给 *User*。

3) *User* 选择整数 $b,s\in_R Z_p^*$，发送 $B=g^bD^s$ 给 GM_{issu}。

4) GM_{issu} 选择 $c\in_R Z_p^*$，发送 c 给 *User*。

5) *User* 发送 b,s 给 GM_{issu}。

6) GM_{issu} 判断等式 $B=g^bD^s$ 是否成立，若成立则计算 $z=(b+c)a+r \bmod p, x=a+b+c$，并对 x 签名，有 $K=(K_1=(g^a)^{1/(\gamma+x)},K_2=g^x,K_3=h^x)$，设置 $REG[i]=(ID,x,K)$，发送 (z,η,K) 给 *User*，发送 (K_2,T) 给属性颁布者 GM_{issa}。

7) *User* 验证 $D=g^{\eta},A^{b+c}RT^{-1}=g^z,e(K_1,\Omega K_3)=e(g,h)^a,e(K_2,h)=e(g,K_3)$ 这 4 个等式是否成立，若成立则设置私钥 $usk=K$。

3 方案形式化定义与安全模型

3.1 方案形式化定义

在本节中，我们给出本文方案的形式化定义，方案中我们定义了 6 种实体，分别是身份颁布者 GM_{issu}、打开者 GM_{opener}、属性颁布者 GM_{issa}、属性打开者 GM_{openat}、群成员以及群外成员，通过将管理员的权限划分给 4 种职责不同的实体，可以防止某个管理员与恶意者合谋导致方案安全性受到威胁，并且可以降低单个管理员的工作负载，标准模型下 CCA 匿名性的失败停止属性基群签名方案由以下 10 个算法组成:

1) Setup $(t_{sp})\rightarrow(isk,osk,oask,params)$: 初始化算法，算法由可信机构执行，输入安全参数 t_{sp}，输出身份颁布私钥 isk，打开私钥 osk，属性打开私钥 $oask$，公共参数 $params$。

2) AttGen $(params,s_T,AS,\Gamma)\rightarrow(aisk,gpk)$: 属性产生算法，算法由可信机构执行，输入公共参数 $params$，根秘密 s_T，访问结构 AS，与访问结构对应的属性全集 Γ，输出属性颁发私钥 $aisk$，群公钥 gpk。

3) Join $(gpk,isk,aisk,ID_i,\Gamma_{req,i})\rightarrow usk_i$: 群外成员加入群组算法，算法由群外成员、$GM_{issu}$、$GM_{issa}$ 执行，输入群公钥 gpk，身份颁发私钥 isk，属性颁发私钥 $aisk$，成员 i 的身份 ID_i 以及申请的属性集 $\Gamma_{req,i}$，通过与 GM 执行交互协议加入群组，输出成员 i 的成员私钥 usk_i。

4) Sign $(msg,usk_i,\varphi_i,gpk)\rightarrow\sigma$: 群成员签名算法，算法由合法群成员执行，输入签名的消息 msg，成员私钥 usk_i，本次签名使用的属性集 φ_i，群公钥 gpk，输出该成员在 msg 上的签名 σ.

5) Verify $(msg,\sigma,gpk)\rightarrow(1/0)$: 群签名验证算法，算法由任意获得消息签名对和群公钥的实体执行，输入消息签名对 (msg,σ)，群公钥 gpk，如果签名有效则输出 1，否则输出 0。

6) Open $(msg,\sigma,osk,gpk)\rightarrow\big((ID_i,\tau_i)/\perp\big)$: 群签名打开算法，算法由 GM_{opener} 执行，输入需要打开的消息签名对 (msg,σ)，打开私钥 osk，群公钥 gpk，如果该签名能够被打开到具体的群成员则输出身份 ID_i 和用于验证打开是否正确的证据 τ_i，否则输出“$\perp$”表示无法打开到具体的群成员。

7) Judge $(ID_i,\tau_i,gpk)\rightarrow(1/0)$: 判断群签名打开是否正确的算法，算法由 GM_{opener} 执行，输入打开群签名产生的身份 ID_i 和证据 τ_i，群公钥 gpk，如果打开是正确的输出 1，否则输出 0。

8) OpenA $(msg,\sigma,oask,gpk)\to\varphi_i$: 属性打开算法, 算法由 GM_{openat} 执行, 输入消息签名对 (msg,σ), 属性打开私钥 $oask$, 群公钥 gpk, 输出签署该签名的属性集 ϕ_i。

9) RevA $(\varphi_{rev},aisk,gpk)\to(g_{val},List[\hat{\lambda}])$: 属性撤销算法, 算法由 GM_{issa} 执行, 输入被撤销的属性集 φ_{rev}, 属性颁布私钥 $aisk$, 群公钥 gpk, 输出撤销后所有有效属性对应群元素的聚合值 g_{val}, gpk 中列表 $List$ 的新的一项 $List[\hat{\lambda}]$。

10) Fail-Stop $(msg',\sigma',usk_i,\varphi,gpk)\to(1/0)$: 失败停止算法, 算法由被伪造的群成员 i 和 GM_{issu} 执行, 输入伪造的消息签名对 (msg',σ'), 成员私钥 usk_i, 用于在该消息签名的属性集 ϕ, 群公钥 gpk, 成员 i 根据输入计算出失败停止的证据 $proof_{fs}$ 和自己签署的消息签名对 (msg',σ) 并发送给 GM_{issu}, GM_{issu} 根据消息签名对和证据判断当前方案是否需要失败停止, 如果需要则输出 1, 否则输出 0。

3.2 安全模型

本文方案的安全模型和预言机定义总体上遵循了 BSZ05[7]和 AA14[14], 不同的是引入了一些额外的预言机, 本节对安全性定义进行简述, 预言机的详细定义见附录 1, 同时在附录 2 中对所有的安全性定义进行了形式化。

定义 1: 正确性。对于诚实的参与方, 本文方案能够得到正确的结果。

定义 2: 成员匿名性(CCA 匿名性)。在多项式时间内, 给定两个任意成员生成的签名, 在能够进行打开询问的情况下, 即使敌手与其他管理员和腐败的成员合谋, 也无法以不可忽略的优势得到实际签名者的身份(即瞎猜)。

定义 3: 可追踪性。在多项式时间内, 敌手 $\mathcal{A}$ 即使和全部管理员以及腐败成员合谋, 也无法以不可忽略的优势来生成一个无法被追踪的签名。

定义 4: 不可陷害性。在多项式时间内, 敌手 $\mathcal{A}$ 即使和全部管理员以及腐败成员合谋也无法以不可忽略的优势来伪造出一个可以通过验证、被追踪到某个诚实成员的签名。

定义 5: 属性匿名性。在多项式时间内, 敌手 $\mathcal{A}$ 无法区分挑战者在给定消息上使用两个符合访问结构的属性集所做的签名来自哪一个属性集, 即使两个属性集完全相同, 而敌手此时只能以 1/2 的概率猜测实际用于签名的属性集。

定义 6: 属性不可伪造性。在多项式时间内, 敌手 $\mathcal{A}$ 无法生成一个与访问结构不符但是可以通过 Verify 算法的伪造成员属性证书。

定义 7: 属性抗联合攻击性。在多项式时间内, 敌手 $\mathcal{A}$ 即使与任意数量拥有无效成员属性证书的成员合谋也无法生成能够通过 Verify 算法的合法成员属性证书, 即合谋后生成的成员属性证书仍然不符合访问结构。

4 方案构造

1) Setup (t_{sp}): 输入安全参数 t_{sp}, 得到阶为素数 p 且满足双线性映射关系 $e:G\times G\to G_T$ 的乘法循环群 G,G_T, 签名的消息 msg 的长度为 m, 满足 $m=O(t_{sp})$, 定义一个抗碰撞的变色龙哈希函数 $\mathrm{H}:\{0,1\}^*\to Z_p^*$。

选择整数 $\gamma,\alpha,\beta,r,s,a\in_R Z_p^*$, 选择生成元 $g,h\in_R G$, 计算 $\Omega=h^{\gamma},g_1=g^{\alpha},g_2=g^{\beta},\vec{f}_1=(g_1,1,g),\vec{f}_2=(1,g_2,g),\vec{f}_3=\vec{f}_1^{\,r}\odot\vec{f}_2^{\,s}$, 其中 1 表示群 G 的单位元, Groth-Sahai 证明系统的公共引用字符串 $CRS=(\vec{f}_1,\vec{f}_2,\vec{f}_3)$, 计算 $A=e(g,h^a)$; 选择生成元 $w_1,w_2\in_R G$ 用于计算可验证加密[3]的检验和。

为了使追踪属性的私钥与追踪成员身份的私钥相互独立, 选择整数 $\alpha',\beta',r',s'\in_R Z_p^*$, 计算 $g_1'=g^{\alpha'}$, $g_2'=g^{\beta'}$ 用于对成员属性进行承诺与加密, 并生成相应的 $CRS'=(\vec{f}_1',\vec{f}_2',\vec{f}_3')$, 颁发属性打开私钥 $oask=(\alpha',\beta')$ 给 GM_{openat}, GM_{openat} 可访问 REG 中属性相关部分。

选择生成元 $v',v_1,...,v_m\in_R G$, m 表示群组中签名消息的长度, 定义 Waters 函数 $\mathrm{F}(\cdot)=(v'\prod_{j=1}^{m}v_j^{m_j})$ 用于对消息签名, 选择整数 $d\in_R Z_p^*$ 作为方案失败停止的证据, 计算 $g_d=g^d$。

颁布身份颁布者私钥 $isk=(\gamma,g^a,a,d)$ 给 GM_{issu}, 打开者私钥 $osk=(\alpha,\beta)$ 给 GM_{opener}, 初始化注册列表 REG 和撤销列表 $List$ 为空, 其中 REG 由 GM_{issu} 管理, GM_{issa} 可在 REG 中增加属性信息, GM_{opener} 可以访问 REG 中与成员身份相关的信息, $List$ 可被公开访问并由属性颁布者 GM_{issa} 管理, 公布公共参数为 $params=(e,g,g_1,g_2,w_1,w_2,g_1',g_2',g_d,h,\Omega,CRS,CRS'$,

$A,\mathcal{F},\mathcal{H},List$)。

2) AttGen ($params,s_T,AS,\Gamma$): 假定群中定义的属性全集为 $\Gamma=\left\{Att_1,Att_2,...,Att_{|\Gamma|}\right\}$，访问结构为 $AS=(\boldsymbol{M},\rho)$，其中访问矩阵 $\boldsymbol{M}$ 的维度为 $|\Gamma|\times col$，定义判断输入属性集是否满足访问结构 AS 的函数 $F(\cdot)\to\{0\,1\}$。

GM_{issa} 设置主秘密 $s_T\in Z_p^*$，计算 $g_{s_T}=g^{s_T}$，随机选择 $col-1$ 个整数 $r_i\in Z_p^*$,其中 $i=2,3,...,col$，生成向量 $\vec{\mathbf{y}}=(s_T,r_2,r_3,...,r_{col})$；对于 $\forall Att_i\in\Gamma$，有对应的子秘密 $s_i=\overrightarrow{\mathbf{AS_i}}\cdot\vec{\mathbf{y}}^{\mathrm{T}}$，最终得到 $S_{Att}=\{s_1,...,s_{|\Gamma|}\}$。选择整数 $r_{ra}\in_R Z_p^*$，计算 $h_{ra}=h^{r_{ra}}$，选择生成元 $g_{val}\in_R G$，初始化撤销列表为 $List[0]=(stat_0,C_0)$，其中 $stat_0$ 为初始化成功信息，$C_0=1\in G_T$。

最终生成属性颁布私钥 $aisk=(r_{ra},g_{val},s_T,S_{Att})$，公布群公钥 $gpk=(params,h_{ra},g_{s_T},AS,F)$。

3) Join ($gpk,isk,aisk,ID_i,\Gamma_{req,i}$): 成员 i 选择整数 $t,k_1,k_2,k_3,k_4\in_R Z_p^*$，令 $T=g^t$ 并与 GM_{issu} 执行 Gro07+协议, 协议结束后成员 i 加入群组成功, 设置身份证书为 $IdCert_i=(ID_i,K)$，随后发送 $(S_{fs,1}=g^{k_1}g_d^{k_3}=g^{\cdot fs_1},S_{fs,2}=g^{k_2}g_d^{k_4}=g^{\cdot fs_2})$ 给 GM_{issu}，GM_{issu} 将 $(S_{fs,1},S_{fs,2})$ 作为用于验证成员遭到伪造后证明的凭据加入到 $REG[i]$，发送 $(ID_i,K_2,T,\Gamma_{req,i})$ 给 GM_{issa}，其中 $\Gamma_{req,i}$ 是成员 i 申请的属性集。

GM_{issa} 验证成员发送的 (K_2,T) 是否与 GM_{issu} 发送的一致，若一致则计算交集 $\Gamma_i=\Gamma\cap\Gamma_{req,i}$ 为实际可颁发的属性集，对于 $\forall Att_j\in\Gamma_i$，计算对应的群元素 $att_{i,j}=K_2^{s_{i,j}}$，其中 $s_{i,j}$ 代表成员 i 申请的第 j 个属性对应的子秘密，计算用于证明成员属性未被撤销的证据 $W_i=g_{val}^{r_{ra}}$，随后更新 $g_{val}=g_{val}\cdot\prod_{j=1}^{|\Gamma_i|}att_{i,j}$；假定 $List$ 最后一非空项为 $List[\hat{\lambda}-1]$，令 $List[\hat{\lambda}]=(stat_{\hat{\lambda}},\ C_{\hat{\lambda}}=e(g_{val},h_{ra}))$，其中 $stat_{\hat{\lambda}}$ 为成员的加入信息，并不包含隐私信息，同时在注册列表 REG 中加入颁布的属性信息，发送属性证书 $AttCert_i=(W_i,\bigcup_{j=1}^{|\Gamma_i|}att_{i,j},\hat{\lambda})$ 和实际颁布的属性集 Γ_i 给成员 i，其中 $\hat{\lambda}$ 用于表示属性证书所处的时间点。

最后，成员 i 验证 $e(\prod_{j=1}^{|\Gamma_i|}att_{i,j},h_{ra})\cdot e(W_i,h)=C_{\hat{\lambda}}$，若成立则设置成员签名私钥 $usk_i=(IdCert_i,AttCert_i,k_1,k_2,k_3,k_4)$。

4) Sign (msg,usk_i,φ_i,gpk): 首先成员判断是否有 $F(\varphi_i)=1$，若有则假定当前 $List$ 的长度为 $\hat{\lambda}+1$，在进行签名之前，若成员 i 的 $AttCert$ 中的时间点并非最新的，则需要请求 GM_{issa} 对属性证书进行更新，大致过程与 Join 相同，不同的是此时仅计算 W_i 即可，不需要对 $List$ 进行修改，更新后该成员被撤销的属性将从成员证书中剔除。

计算当前成员有效属性的聚合值 $\widetilde{att}=\prod_{j=1}^{|\Gamma_i|}att_{i,j}$，此时有等式

$$e(\widetilde{att},h_{ra})\cdot e(W_i,h)=C_{\hat{\lambda}}\tag{1}$$

成立，为了保证匿名性需要对 $\widetilde{att},W_i$ 进行承诺，得到 $C_1=Comm(\widetilde{att}),C_2=Comm(W_i)$，由于式(1)为线性配对乘积等式，证明大小为 3 个群元素，故有 $\pi_1=\mathrm{NIWI}\{(\widetilde{att},W_i):e(\widetilde{att},h_{ra})\cdot e(W_i,h)=C_{\hat{\lambda}}\}$ 用于证明等式(1)以表示成员 i 的属性集合法。

随后，成员 i 对消息 msg 签名，选择整数 $s_{rnd}\in_R Z_p^*$，计算 $K_3^{'}=K_3\mathcal{F}(msg)^{s_{rnd}}$，此时有等式

$$e(K_2,h)\cdot e(g^{s_{rnd}},\mathcal{F}(msg))=e(g,K_3^{'})\tag{2}$$

$$e(K_1,\Omega K_3)=e(g,h)^a=A\tag{3}$$

成立，记 $(K_1,K_2,K_3^{'},g^{s_{rnd}})=(S_1,S_2,S_3,S_4)$ 对 K_3,S_1,S_2,S_3,S_4 进行承诺，有 $C_3=Comm(K_3),C_4=Comm(S_1),C_5=Comm(S_2),C_6=Comm(S_3),C_7=Comm(S_4)$，由于式(2)为线性配对乘积等式，证明大小为 3 个群元素，而式(3)为非线性配对乘积等式，证明大小为 9 个群元素，得到证明 $\pi_2=\mathrm{NIWI}\{(S_2,S_3,S_4):e(S_2,h)\cdot e(S_4,\mathrm{F}(msg))=e(g,\ S_3)\}$，$\pi_3=\mathrm{NIWI}\{(S_1,K_3):e(S_1,\Omega K_3)=A\}$，用于证明式(3)成立。

为了实现 CCA 匿名性，还需要使用可验证加密对 S_1 进行加密，选择整数 $r_1,r_2\in_R Z_p^*$，计算 $u_1=g_1^{r_1}$，$u_2=g_2^{r_2},e=S_1g^{r_1+r_2}$，此时有等式

$$\begin{aligned}u_1^{-1}\cdot C_{4,1}&=g_1^{\hat{r}}\cdot\vec{\mathrm{f}}_{3,1}^{\hat{t}}\\u_2^{-1}\cdot C_{4,2}&=g_2^{\hat{s}}\cdot\vec{\mathrm{f}}_{3,2}^{\hat{t}}\\e^{-1}\cdot C_{4,3}&=g^{\hat{r}+\hat{s}}\cdot\vec{\mathrm{f}}_{3,3}^{\hat{t}}\end{aligned}\tag{4}$$

成立，其中 $\hat{r},\hat{s},\hat{t}$ 可由式(4)计算得出，需要计算承诺 $C_8=comm(\hat{r}),C_9=comm(\hat{s}),C_{10}=comm(\hat{t})$，有证

明 $\theta_1=\mathrm{NIZK}\{(\hat{r},\hat{t}):u_1^{-1}\cdot C_{4,1}=g_1^{\hat{r}}\cdot\vec{\mathrm{f}}_{3,1}^{\hat{t}}\}$ $\theta_2=\mathrm{NIZK}\{(\hat{r},\hat{t}):u_2^{-1}\cdot C_{4,2}=g_2^{\hat{s}}\cdot\vec{\mathrm{f}}_{3,2}^{\hat{t}}\}$ $\theta_3=\mathrm{NIZK}\{(\hat{r},\hat{s},\hat{t}):e^{-1}\cdot C_{4,3}=g^{\hat{r}+\hat{s}}\cdot\vec{\mathrm{f}}_{3,3}^{\hat{t}}\}$，式(4)为线性多指数等式，证明大小均为两个群元素。

为了证明成员属性集满足访问结构，首先需要对属性集 φ_i 减枝使属性集保证最简，减枝后的属性集记为 ϕ_i，此时可以通过函数 $\rho(att_j)$ 获取 $|\phi_i|\times col$ 的矩阵 $\hat{\mathrm{M}}$，其中 $att_j\in\phi_i$ 并且 col 的大小由线性秘密分享方案决定，此时存在向量 $\vec{\boldsymbol{\omega}}$ 使 $\hat{\mathrm{M}}\cdot\vec{\boldsymbol{\omega}}^{\mathrm{T}}=(1,0,0,\ldots,0)^{\mathrm{T}}$。对于 $\forall Att_j\in\phi_i$，选择整数 $r'_{1,j}$，$r'_{2,j}\in_R Z_p^*$，对群元素 att_j 和 $g^{\vec{\omega}_j}$ 使用 CRS' 进行承诺，用 $Comm'(\cdot),comm'(\cdot)$ 表示用 CRS' 承诺相应群元素以与 CRS 进行区分；为了保证 CCA 匿名性，需要使用 (g,g'_1,g'_2) 对 att_j 进行加密，用 $\mathrm{Enc}(att_j,r'_{1,j},r'_{2,j})$ 来描述加密过程，具体过程与 Sign 一致，有 $C_{att_j}=Comm'(att_j),C_{w_j}=Comm'(g^{\vec{\omega}_j}),E_{att_j}=\mathrm{Enc}(att_j,r'_{1,j},r'_{2,j})=(u_{1,att_j},u_{2,att_j},e_{att_j})$。此时同样有线性多指数等式成立，即

$$\begin{aligned}u_{1,att_j}^{-1}\cdot C_{att_j,1}&=g_1^{\hat{r}'_j}\cdot\vec{\mathrm{f}}_{3,1}^{\hat{t}'_j}\\ u_{2,att_j}^{-1}\cdot C_{att_j,2}&=g_2^{\hat{s}'_j}\cdot\vec{\mathrm{f}}_{3,2}^{\hat{t}'_j}\\ e_{att_j}^{-1}\cdot C_{att_j,3}&=g^{\hat{r}'_j+\hat{s}'_j}\cdot\vec{\mathrm{f}}_{3,3}^{\hat{t}'_j}\end{aligned}\tag{5}$$

为了证明式(5)，有承诺 $C_{\hat{r}'_j}=comm'(\hat{r}'_j),C_{\hat{s}'_j}=comm'(\hat{s}'_j),C_{\hat{t}'_j}=comm'(\hat{t}'_j)$ 和大小均为 2 个群元素的证明 $\theta_{att,j,1},\theta_{att,j,2},\theta_{att,j,3}$，记 $\theta_{att_j}=(\theta_{att_j,1},\theta_{att_j,2},\theta_{att_j,3}),\theta_{att}=(\theta_{att_1},\ldots,\theta_{att_{|\phi_i|}}),C_{att}=(C_{att_1},\ \ldots,C_{att_{|\phi_i|}})$，$C_w=(C_{w_1},\ldots,C_{w_{|\phi_i|}})$，$\hat{C}=(\bigcup_{j=1}^{|\phi_i|}(C_{\hat{r}'_j},C_{\hat{s}'_j},C_{\hat{t}'_j}))$，$E_{att}=(E_{att_1},\ldots,E_{att_{|\phi_i|}})$。此时有配对乘积等式

$$\prod_{j=1}^{|\phi_i|}e(att_j,h^{\vec{\omega}_j})=e(g_{s_T},K_3)\tag{6}$$

成立，可以证明签名使用的属性集满足访问结构，证明式(6)需要 9 个群元素，有 $\pi_4=\mathrm{NIWI}\{(\bigcup_{j=1}^{|\phi_i|}(att_j,h^{\vec{\omega}_j}),K_3):\prod_{j=1}^{|\phi_i|}e(att_j,h^{\vec{\omega}_j})=e(g_{s_T},K_3)\}$，注意到由于 att_j 是使用 CRS' 进行承诺的，因此还需要对 K_3 再次进行承诺，否则无法正确验证等式，记为 $C_{11}=Comm'(K_3)$。

计算 $c_1=k_1+\mathcal{H}(msg)k_2,c_2=k_3+\mathcal{H}(msg)k_4$，对于不同的消息，$c_1,c_2$ 是不可链接的。选择整数 $r_3\in_R Z_p^*$，计算变色龙哈希值 $\tilde{t}=\mathcal{H}(\mathcal{F}(msg),C_{1\sim11},\pi_{1\sim4},\theta_{1\sim3},u_1,u_2,e,\theta_{att},C_{att},C_w,\hat{C},E_{att},c_1,c_2)$ $t=\mathrm{H}(g_1^{\tilde{t}}h^{r_3})$，计算检验和 $v_1=(g^tw_1)^{r_1},v_2=(g^tw_2)^{r_2}$ 用于验证密文。

最终得到签名 $\sigma=(C_{1\sim11},C_{att},C_w,\hat{C},\pi_{1\sim4},\theta_{1\sim3},\theta_{att},u_1,u_2,e,E_{att},c_1,c_2,r_3,v_1,v_2,\hat{\lambda})$，签名由 $(68+24|\phi_i|)$ 个群 G 元素和 3 个群 Z_p^* 元素构成。

5) Verify (msg,σ,gpk)：计算哈希值 $\tilde{t}'=\mathcal{H}(\mathcal{F}(msg),C_{1\sim11},\pi_{1\sim4},\theta_{1\sim3},u_1,u_2,e,\theta_{att},C_{att},C_w,\hat{C},E_{att},c_1,c_2)$ 和 $t'=\mathrm{H}(g_1^{\tilde{t}'}h^{r_3})$，验证等式 $e(g_1,v_1)\cdot e(g_2,v_2)=e(u_1,w_1g^t)\cdot e(u_2,w_2g^t)$，随后根据 $List[\hat{\lambda}]$ 中的信息对上述的NIWI证明和NIZK证明进行验证，若验证通过则签名合法。

6) Open (msg,σ,osk,gpk)：GM_{opener} 使用打开私钥 $osk=(\alpha,\beta)$ 打开签名 σ，计算 $K_1=C_{4,3}/(C_{4,1}^{1/\alpha}\cdot C_{4,2}^{1/\beta})$，$\tau_i=C_{5,3}/(g_1^{1/\alpha}\cdot g_1^{1/\beta})$，查询 REG 并输出实际签名成员的身份 ID_i 和 τ_i。注意到对 C_4 的打开等同于 $K_1=e/(u_1^{1/\alpha}\cdot u_2^{1/\beta})$，即对身份证书加密的密文进行解密。

7) Judge (ID_i,τ_i,gpk)：GM_{opener} 根据输入的 (ID_i,τ_i) 查询 $REG[i]$，得到 K_3，如果有 $e(\tau_i,h)=e(g,K_3)$ 成立则输出 1，否则输出 0。

8) OpenA $(msg,\sigma,oask,gpk)$：GM_{openat} 使用打开属性私钥 $oask=(\alpha',\beta')$ 打开签名 σ 中对属性的承诺，对于 $\forall Att_j\in\phi_i$，计算 $att_{i,j}=C_{att_j,3}/(u_{1,j}^{1/\alpha'}\cdot u_{2,j}^{1/\beta'})$，在 REG 中查找所有相应的属性最终输出属性集 ϕ_i。

9) RevA $(\varphi_{rev},aisk,gpk)$：对于要撤销的属性 $Att_j\in\varphi_{rev}$，计算 $att_{rev}=\prod_{j=1}^{|\varphi_{rev}|}att_j$，更新 $g_{val}=g_{val}/att_{rev}$，假定 $List$ 最后一非空项为 $List[\hat{\lambda}-1]$，计算 $List[\hat{\lambda}]=(stat_{\hat{\lambda}},C_{\hat{\lambda}}=e(g_{val},h_{ra}))$，$stat_{\hat{\lambda}}$ 表示撤销信息，此时所有属性集 φ_{rev} 中的属性均被撤销，所有原有的成员进行签名均需要更新属性证书并重新获得证据 W_i；如果需要撤销某个成员的签名能力，则直接撤销颁发给该成员的所有属性即可。注意到，要恢复成员的属性证书，即重新聚合被撤销的属性即可。

10) Fail-Stop $(msg',\sigma',usk_i,\varphi,gpk)$：假设有敌手

攻破本文方案依赖的困难性问题或者窃取了成员证书, 最终成功伪造出了成员 i 的签名 $\sigma'=(...,c_1',c_2',\hat{\lambda})$, 此时打开签名将指向成员 i, 根据 c_1',c_2' 有

$$\begin{aligned} k_1'+\mathrm{H}(msg')k_2'&=c_1'\\ k_3'+\mathrm{H}(msg')k_4'&=c_2'\\ k_1'+dk_3'&=fs_1'\\ k_2'+dk_4'&=fs_2' \end{aligned} \tag{7}$$

成立, 由于式(1.7)中系数矩阵的秩为 3, 此时 $(k_1',k_2',k_3',k_4')=(k_1,k_2,k_3,k_4)$ 的概率为 $1/p$, 因此成员 i 可以对争议消息进行签名, 有 $\sigma=(...,c_1,c_2,\hat{\lambda})$, 根据 c_1',c_2' 和 c_1,c_2 可以计算出 $d=\log_g g_d=(c_1'-c_1)\cdot(c_2-c_2')^{-1}$, 发送 $proof_{fs}=(ID_i,msg',\sigma,d)$ 给 GM_{issu} 作为方案失败停止的证据, GM_{issu} 验证 $S_{fs,1}S_{fs,2}^{\mathrm{H}(msg')}=g^{c_1}g_d^{c_2}$, 并判断 σ,d 的有效性, 随后裁定当前方案是否仍然安全。

5 CPA 匿名性的方案

出于计算开销和签名大小的考虑, 本文对上述方案进行简化, 简化的方案仅实现 CPA 匿名性, 没有可验证加密和 NIZK 证明, 减少了一定的计算开销和签名大小, 简化的方案仅签名和验证阶段不同, 故仅简述签名和验证阶段的步骤。

1) Sign-$(msg,usk_i,\varphi_i,gpk)$:

此时只保证 CPA 匿名性, 此时仅需计算 $C_{1\sim7,11}$, $C_{att},C_w,\pi_{1\sim4},c_1,c_2$ 即可, 对属性和成员证书只进行承诺, 并不使用可验证加密进行加密和 NIZK 证明, 最终生成的签名为 $\sigma=(C_{1\sim7,11},C_{att},C_w,\ \pi_{1\sim4},c_1,c_2,\hat{\lambda})$, 签名大小为 $(48+6|\phi_i|)$ 个群 G 元素和 2 个群 Z_p^* 元素。

2) Verify-(msg,σ,gpk):

根据 $List[\hat{\lambda}]$ 中的信息对上述的 NIWI 证明进行验证, 若验证通过则签名合法。

6 安全性证明

6.1 正确性

定理 1. 对于诚实的参与者, 本文方案是正确的。

证明. 在 Sign 算法中, 由式(1.1)有 $e(\widetilde{att},h_{ra})\cdot e(W_i,h)=e(\prod_{j=1}^{|\Gamma_i|}att_{i,j},h^{r_{ra}})\cdot e((\prod_{j=1}^{|\Gamma|}att_j/\prod_{j=1}^{|\Gamma_i|}att_{i,j})^{r_{ra}},h)=e(g_{val},h^{r_{ra}})=C_{\hat{\lambda}}$, 由式(1.2)有 $e(K_2,h)\cdot e(g^{s_{rnd}},\mathrm{F}(msg))=e(g,h^{x_i}\cdot\mathrm{F}(msg)^{s_{rnd}})=e(g,K_3')$, 由式(1.3)有 $e(K_1,\Omega K_3)=e((g^a)^{1/(\gamma+x_i)},\ h^{\gamma+x_i})=e(g,h)^a$, 由式(4)有 $u_1^{-1}\cdot C_{5,1}=g_1^{-r_1}\cdot g_1^{r_{C_5}}\cdot\ \vec{\mathrm{f}}_{3,1}^{t_{C_5}}=g_1^{r_{C_5}-r_1}\cdot\vec{\mathrm{f}}_{3,1}^{t_{C_5}}=g_1^{\hat{r}}\cdot\vec{\mathrm{f}}_{3,1}^{\hat{t}}$ 同理证明式(4)和式(5)中的类似式子, 由式(6), 有 $\prod_{j=1}^{|\phi_i|}e(att_j,h^{\tilde{\omega}_j})=e(g^{\sum_{j=1}^{|\phi_i|}\vec{s}_j\tilde{\omega}_j},h^{x_i})=e(g_{s_T},K_3)$, 因此 Sign 是正确的。

在 Verify 算法中, 有 $e(g_1,v_1)\cdot e(g_2,v_2)=e(g_1^{r_1},w_1g^t)\cdot e(g_2^{r_2},w_2g^t)=e(u_1,w_1g^t)\cdot e(u_2,w_2g^t)$, 因此 Verify 是正确的。

在 Open 算法中, $K_1=C_{4,3}/(C_{4,1}^{1/\alpha}\cdot C_{4,2}^{1/\beta})=K_1\cdot g^{r_4+s_4}/(g_1^{r_4/\alpha}\cdot g_2^{r_4/\beta})=K_1\cdot g^{r_4+s_4}/g^{r_4+s_4}=K_1$, 可以正确打开到成员 i 对应的 K_1, 因此 Open 是正确的, 由于对成员属性的承诺是使用另一组 CRS, 打开属性即是打开承诺, 同理 OpenA 也是正确的。

在 Judge 算法中, $e(\tau_i,h)=e(g^x,h)=e(g,h^x)=e(g,K_2)$, 因此 Judge 是正确的。

在 RevA 算法中, 被撤销的成员属性不被包含在 g_{val} 中, 更新后的 W_i 无法使 $e(att_i,h_{ra})\cdot e(W_i,h)=C_{\hat{\lambda}}=e(g_{val},h^{r_{ra}})$ 成立; 同时, 被撤销属性的成员无法使用旧的证据 W_i 在新的时间点进行签名, 但撤销之前进行的签名仍可被正确验证, 因此 RevA 是正确的。

在 Fail-Stop 算法中, 对于证据 $proof_{fs}=(ID_i,msg',\sigma,d)$ 有 $S_{fs,1}S_{fs,2}^{\mathrm{H}(msg')}=g^{k_1}g_d^{k_3}\cdot(g^{k_2}g_d^{k_4})^{\mathrm{H}(msg')}=g^{k_1+\mathrm{H}(msg')k_2}g_d^{k_3+\mathrm{H}(msg')k_4}=g^{c_1}g_d^{c_2}$, 因此 Fail-Stop 算法是正确的。

6.2 成员匿名性

定理 2. 本文方案具有成员匿名性(CCA), 如果存在一个多项式时间的敌手 $\mathcal{A}$ 能够打破本文方案的成员匿名性, 则存在一个模拟器 $\mathcal{S}$ 可以打破哈希函数的抗碰撞性或 DLIN 假设。

证明. 本文使用 Groth-Sahai 证明系统和基于变色龙哈希函数的可验证加密方案来保证本文方案的 CCA 匿名性, 证明思路与 YZW15[10]和 LPY15[20]相似。我们用 *Game* 序列进行证明, 目标是证明敌手 $\mathcal{A}$ 无法以不可忽略的优势打破真实方案的 CCA 匿名性, 用 Pr[i] 表示 *Game* i 中敌手 $\mathcal{A}$ 获胜的概率, Adv_i 表示敌手在游戏中的优势。

Game 1. 即本文中的真实方案, 敌手与模拟器按照真实方案进行游戏:

初始化阶段, 模拟器 $\mathcal{S}$ 按照 CCA 匿名性定义中

设置游戏为真实游戏, 即此时方案与真实方案无差异, $\mathcal{A}$ 获得真实方案中的 gpk, isk。

询问阶段 1, 敌手 $\mathcal{A}$ 被赋予访问预言机 $\mathcal{O}_{Ch}, \mathcal{O}_{SndToH}, \mathcal{O}_{WReg}, \mathcal{O}_{Crpt}, \mathcal{O}_{Usk}, \mathcal{O}_{Open}$ 的权限, 模拟器 $\mathcal{S}$ 回应 $\mathcal{A}$ 对任意签名的 $\mathcal{O}_{Open}$ 预言询问, 返回询问签名的相应身份, 对于其他询问则按照定义返回相应值。

挑战阶段, $\mathcal{A}$ 发起 $\mathcal{O}_{Ch}$ 询问, 发送消息 msg 和两个诚实成员的身份 ID_0, ID_1, 模拟器随机选择一个身份记为 ID_c 并生成挑战签名 σ_c 作为挑战发送给敌手。

询问阶段 2, 此时 $\mathcal{A}$ 不能进行 σ_c 相关的询问, 其余的与询问阶段 1 一致。

输出阶段, 敌手 $\mathcal{A}$ 停止询问并输出自己的猜测 $ID_{\mathcal{A}}$, 如果 $ID_{\mathcal{A}} = ID_c$ 则 $\mathcal{A}$ 赢得该游戏, 记 $\mathcal{A}$ 在 *Game* 1 的优势为 $Adv_1 = |\Pr[1] - 1/2|$。

Game 2. 在 *Game* 1 的基础上, 如果敌手 $\mathcal{A}$ 在 $\mathcal{O}_{Open}$ 询问过程中询问签名的哈希值与挑战签名 σ_c 的哈希值发生碰撞, 此时 $\mathcal{S}$ 模拟终止。$\mathcal{A}$ 应具有打破哈希函数的抗碰撞性或者计算离散对数来使哈希值碰撞的能力, 用 Adv^{CR}, Adv^{DL} 来表示对应的优势, 又由于使用的可验证加密方案基于 DLIN 问题, 因此有 $Adv^{DL} < Adv^{DLIN}$, 即 $|\Pr[2] - \Pr[1]| \leq Adv^{CR} + Adv^{DLIN}$。

Game 3. 对 *Game* 2 进行以下修改, 使 *Game* 2 与 *Game* 3 在敌手看来不可区分:

初始化阶段, 模拟器 $\mathcal{S}$ 拥有变色龙哈希函数的陷门 td, 选择 $\hat{\tau}, \zeta_1, \zeta_2 \in_R Z_p^*$, 令 $h = g^{td}, w_1 = g^{\hat{\tau}} g_1^{\xi_1}$, $w_2 = g^{\hat{\tau}} g_2^{\xi_2}$。

询问阶段, 对于 $\mathcal{O}_{Open}$ 询问, $\mathcal{S}$ 通过计算 $g^{r_1} = (v_1 / e_1^{\xi_1})^{1/(t+\tau)} = (g^t g^{\tau} g^{\alpha\xi_1} / g_1^{\xi_1})^{r_1/(t+\tau)}, g^{r_2} = (v_2 / e_2^{\xi_2})^{1/(t+\hat{\tau})} = (g^t g^{\hat{\tau}} g^{\beta\xi_2} / g_2^{\xi_2})^{r_2/(t+\hat{\tau})}, K_1 = e_3 / (g^{r_1} \cdot g^{r_2})$, 此时不需要 osk 即可打开签名, 其中 t 是签名的哈希值。

挑战阶段, 模拟器 $\mathcal{S}$ 首先计算挑战签名中包含承诺和证明的哈希值的 $\tilde{t}_c$, 随后计算用于验证的哈希值 t_c, 由于模拟器拥有变色龙哈希函数的陷门 td, 因此可以根据 td 找到碰撞值 r_3' 使 $t_c' = \mathrm{H}(g_1^{\hat{t}} h^{r_3'}) = t_c$ 并发送挑战签名 σ_c。

在敌手 $\mathcal{A}$ 看来 *Game* 3 和 *Game* 2 生成的签名在分布上是不可区分的, 因此 Pr[3]=Pr[2]。

Game 4. 对于 NIWI 证明和 NIZK 证明, 在 *Game* 3 中对 Groth-Sahai 证明系统的公共引用字符串 CRS, CRS' 和 $\vec{\mu}$ 进行修改:

初始化阶段, 模拟器 $\mathcal{S}$ 根据 $CRS = (\vec{\mathrm{f}}_1, \vec{\mathrm{f}}_2, \vec{\mathrm{f}}_3)$, 将 CRS 和 $\vec{\mu}$ 设置为"证据不可区分"情景, 此时 $\vec{\mathrm{f}}_1, \vec{\mathrm{f}}_2, \vec{\mathrm{f}}_3$ 线性独立, 有 $\vec{\mathrm{f}}_3 = \vec{\mathrm{f}}_1^{r} \odot \vec{\mathrm{f}}_2^{s} \odot \iota(g), \vec{\mu} = \vec{\mathrm{f}}_1^{r} \odot \vec{\mathrm{f}}_2^{s}$, 对于 CRS' 采用同样的设置方式。在真实方案中 CRS, $CRS', \vec{\mu}$ 是"完全正确"情景下的, 在敌手 $\mathcal{A}$ 看来这两种设置在计算上是不可区分的, 由于 Groth-Sahai 证明系统的 DLIN 实例中对群元素的承诺是进行 DLIN 加密, 敌手 $\mathcal{A}$ 获胜的概率为区分两种设置方式的概率, 即在两个设置中解决 DLIN 问题的概率, 因此有 $|\Pr[4]\text{-}\Pr[3]| = 2 \cdot Adv^{DLIN}$。

Game 5. 在 *Game* 4 的挑战阶段中, 对加密 K_1 过程进行修改:

$\mathcal{S}$ 选择随机比特 $b \in_R \{0,1\}$, 当 $b = 0$ 时, 设置 $e_3 = K_1 g^{r_{rnd}}$, 其中 $r_{rnd} \in_R Z_p^*$; 当 $b = 1$ 时, 设置 $e_3 = K_1 g^{r_1 + r_2}$。输出阶段, 敌手 $\mathcal{A}$ 需要判断 $b = 0$ 还是 $b = 1$, 即解决 DLIN 问题, 因此有 $|\Pr[5]\text{-}\Pr[4]| = Adv^{DLIN}$。

在 *Game* 5 中, Groth-Sahai 证明系统的所有承诺被完美的隐藏在"证据不可区分"情景中, NIWI/NIZK 证明不会揭示任何信息, 而对成员身份证书加密所得到的密文也隐藏了证书信息, 因此签名独立于消息, 有 Pr[5]=1/2。综上所述, $Adv_1 \leq Adv^{CR} + 4Adv^{DLIN}$, 由于哈希函数是抗碰撞的, 在 DLIN 假设下 Adv^{DLIN} 是可忽略的, 因此本文方案具有成员匿名性(CCA)。

6.3 可追踪性

定理 3. 本文方案具有可追踪性, 如果存在一个多项式时间的敌手能够打破本文方案的可追踪性, 则存在一个模拟器 $\mathcal{S}$ 可以打破 q-HSDH 假设。

证明. 敌手 $\mathcal{A}$ 使用身份 $ID^* \notin \{ID_1, ID_2, ..., ID_n\}$ 并成功伪造出群签名 (msg^*, σ^*), n 表示已加入群组的群成员, $n \leq q - 1$, 此时身份 ID^* 不在注册列表中。

初始化阶段, 模拟器 $\mathcal{S}$ 按照方案中 Setup 和 AttGen 生成群公钥 $gpk = (params, h_{ra}, g_{s_T}, AS, F)$ 和管理私钥, 不同的是此时选择 $z', z_1, ..., z_m \in_R Z_p^*$, 有 $v' = g^{z'}, v_1 = g^{z_1}, ..., v_m = g^{z_m}$, 并建立初始为空的列表 REG 用于回应敌手询问, $\mathcal{S}$ 将群公钥和所有管理员私钥发送给敌手 $\mathcal{A}$。

询问阶段, 敌手能够询问预言机 $\mathrm{O}_{SndToC}, \mathrm{O}_{AddU}, \mathrm{O}_{RReg}, \mathrm{O}_{Usk}, \mathrm{O}_{Crpt}$, 对于敌手对 $\mathcal{O}_{SndToC}$ 关于索引 i 的

询问, 如果 $REG[i]$ 存在, 则返回索引为 i 的群成员的成员证书; 如果 $REG[i]$ 不存在, 则随机选择 $x \in_R Z_p^*$, 设置 $REG[i]=(ID,x,K_i,S_1,S_2,\Gamma_i)$ 将其加入群组, 其中 $K_i=(g^{a/(\gamma+x_i)},g^{x_i},h^{x_i})$, 敌手 $\mathcal{A}$ 至多进行 $q-1$ 次加入询问。

伪造阶段, 敌手 $\mathcal{A}$ 选择一个 $ID^* \notin \{ID_1,...,ID_{q-1}\}$ 输出伪造的签名 σ^* 在消息 msg^* 上, 对于签名 σ^*, Verify 和 Judge 应正确输出 1, 模拟器 $\mathcal{S}$ 可以使用打开私钥 (α,β) 打开签名中的承诺 C_4,C_5,C_6,C_7 得到 S_1,S_2,S_3,S_4, 其中 S_1,S_2 必须是唯一的, 否则表明伪造失败, 有 $(S_1^{1/\alpha},S_2,S_3/(S_4)^{z'+\sum_{j=1}^{m'}z_j})$ 作为 q-HSDH 问题的解, 此时 q-HSDH 问题被解决。

6.4 不可陷害性

定理 4. 本文方案具有不可陷害性, 如果存在一个多项式时间的敌手能够打破本文方案的不可陷害性, 则存在模拟器 $\mathcal{S}$ 能够打破 q-HSDH 假设或者打破失败停止签名。

证明. 对于不可陷害性我们考虑两种敌手, 不难看出本文方案对于消息的签名由二级签名[8]和失败停止签名[5]构成。定义 Type-1 敌手试图伪造合法成员的身份证书, 以在该成员没签署的消息上进行签名, 由于身份证书是管理员 GM_{issu} 在成员身份上的签名, 对消息进行签名则是成员使用成员身份证书对消息进行签名, 因此对于 Type-1 敌手伪造的证明与 BW07[8]一致, 等同于证明二级签名在自适应选择消息攻击下满足存在不可伪造性。Type-2 敌手具有无限的计算能力, 能够打破方案依赖的困难性问题并陷害合法成员, 此时方案应该失败停止。

Type-1: 给定模拟器 $\mathcal{S}$ 一个 q-HSDH 实例 $(g,g^{\gamma},h,(g^{1/(\gamma+x_i)},g^{x_i},h^{x_i})_{i=1,2,...,q-1})$, 敌手 $\mathcal{A}$ 使用身份 $ID^* \in \{ID_1,ID_2,...,ID_n\}$ 并成功伪造出群签名 (msg^*,σ^*), n 表示已加入群组的群成员, $n \le q-1$, 此时身份 ID^* 在注册列表中, 但是 msg^* 是询问阶段没有询问过的消息的签名。

初始化阶段, 模拟器 $\mathcal{S}$ 按照方案中 Setup 和 AttGen 生成群公钥 $gpk=(params,h_{ra},g_{s_T},AS,F)$ 和管理私钥, 不同的是此时选择 $t,z',z_1,...,z_m,b',b_1,...,b_m,\hat{p},\hat{q},\partial \in_R Z_p^*$, 有 $h=g^{\partial}$, 计算 $f=\Omega^{-1}g^t$, 有 $v'=g^{z'}f^{b'-2\hat{p}\hat{q}},v_1=g^{z_1}f^{b_1},...,v_m=g^{z_m}\quad f^{b_m}$, 并建立初始为空的列表 REG 用于回应敌手询问, $\mathcal{S}$ 将群公钥和所有管理员私钥发送给敌手 $\mathcal{A}$, 预先选定一个 $ID^* \in_R \{ID_1,...,ID_n\}$, 敌手最后需要输出 ID^* 的签名。

询问阶段, 敌手 $\mathcal{A}$ 被赋予访问预言机 O_{SndToH}, $O_{WReg},O_{Crpt},O_{HSign},O_{Usk}$ 的权限, 对于敌手提交预言机 $\mathcal{O}_{HSign}$ 关于索引 i 在消息 msg 的签名询问, 如果 $ID_i \neq ID^*$, 若 $REG[i]$ 存在, 则在消息 msg 上使用索引为 i 的群成员的成员证书, 选择属性集 Γ_i 生成签名并返回; 若 $REG[i]$ 不存在, 则设置 $REG[i]=(ID,x,K_i,S_1,S_2,\Gamma_i)$ 将其加入群组, 并返回签名。如果 $ID_i=ID^*$, 定义 $F=y'-2\hat{p}\hat{q}+\sum_{j=1}^{m}y_jm_j$, $J=z'+\sum_{j=1}^{m}z_jm_j$, 如果 $F \equiv 0 \bmod p$, 则模拟停止, 否则选择 $s_{rnd} \in_R Z_p^*$, 有 $(S_1,S_2,S_3,S_4)=(g^{a/t},f,\quad h^{-J/F}\mathrm{F}(msg)^{s_{rnd}},h^{1/F}g^{-s_{rnd}})$, 记 $t=\gamma+x^*,\hat{s}_{rnd}=s_{rnd}-\partial/F$, 有 $S_3=h^{-J/F}(\mathcal{F}(msg))^{s_{rnd}}=h^{-J/F}(\mathcal{F}(msg))^{\hat{s}_{rnd}}$ $(fg^{J/F})^{\partial}=(\Omega^{-1}g^t)^{\partial}(\mathcal{F}(msg))^{\hat{s}_{rnd}}=h^{x^*}(\mathcal{F}(msg))^{\hat{s}_{rnd}}$, $S_4=h^{1/F}g^{-\hat{s}_{rnd}}g^{-\partial/F}=g^{-\hat{s}_{rnd}}$, 因此等价于 $(S_1,S_2,S_3,S_4)=(g^{a/(\gamma+x^*)},g^{x^*},h^{x^*}\mathcal{F}(msg)^{\hat{s}_{rnd}},g^{-\hat{s}_{rnd}})$, 最终返回完整的签名 σ。

伪造阶段, 敌手经过至多 $q-1$ 次加入询问或 s 次签名后, 需要在消息 msg^* 上输出伪造签名 σ^*, 此时满足 $ID^* \in \{ID_1,...,ID_n\}$。如果 $\mathcal{S}$ 使用打开私钥打开承诺 C_5, 有 $S_2 \neq f$, 表明收到不正确的身份, 或者不满足 $F \equiv 0 \bmod p$, 表明收到无效的伪造, 则 $\mathcal{S}$ 中止模拟。最后, 对于正确的伪造 $\mathcal{S}$ 打开承诺 C_4,C_5,C_6,C_7 得到 S_1,S_2,S_3,S_4, 有 $(S_1^{1/\alpha},S_2,S_3\cdot S_4^{J})$ 作为 q-HSDH 问题的解。

Type-2: 敌手 $\mathcal{A}$ 此时拥有无限的计算能力, 能够打破困难性假设从而解决困难性问题, 假定敌手通过成员 i 已有的签名中的 c_1,c_2 推断出了成员 i 签名私钥 usk_i 中的 k_1',k_2',k_3',k_4', 并使用 k_1',k_2',k_3',k_4' 在消息 msg' 上进行签名, 得到 $\sigma'=(...,c_1',c_2',\hat{\lambda})$。此时满足 $k_1'+\mathrm{H}(msg')k_2'=c_1'$, $k_3'+\mathrm{H}(msg')k_4'=c_2'$, 存在一组 $S'_{fs,1},S'_{fs,2}$ 使 $S'_{fs,1}S'^{\mathrm{H}(msg')}_{fs,2}=g^{c_1'}g_d^{c_2'}$ 成立, 即方程组(1.7)成立。令 low_j 表示第 j 个方程, 由于 $low_1+d\cdot low_2-low_3\ -\mathrm{H}(msg')\cdot low_4=0$, 因此系数矩阵秩为

3, 此时由于方程定义在群 Z_p^* 上, 此时有 p 个 k_1', k_2', k_3', k_4' 可以满足方程, 因此 $\Pr[(k_1', k_2', k_3', k_4') = (k_1, k_2, k_3, k_4)] = 1/p$, 成员 i 可以根据伪造的签名来解决离散对数问题 $d = \log_g g_d = (c_1' - c_1) \cdot (c_2 - c_2')^{-1}$ 并作为方案失败停止的证据, 因此失败停止签名不依赖于任何密码学假设, 是无条件安全的。

综上所述, 由于 q-HSDH 问题是困难的, 失败停止签名是无条件安全的, 因此不存在敌手能够进行上述伪造, 本文方案具有不可陷害性。

6.5 属性匿名性

定理 5. 本文方案具有属性匿名性, 如果存在一个多项式时间的敌手能够打破本文方案的属性匿名性, 则存在模拟器能够打破 DLIN 假设。

证明. 本文使用 Groth-Sahai 证明系统和可验证加密方案来保证本文方案的属性匿名性, 属性匿名性的证明与 CCA 匿名性的证明类似, 故不进行赘述, 由于 $K_2^{s_T}$ 并没有揭示任何成员属性集的相关信息, 对签名进行验证仅能验证成员签名的属性集符合访问结构, 因此在 DLIN 假设下, 本文方案具有属性匿名性。

6.6 属性不可伪造性

定理 6. 本文方案具有属性不可伪造性, 如果存在一个多项式时间的敌手能够打破本文方案的属性不可伪造性, 则存在一个模拟器 $\mathcal{S}$ 可以打破 ODBP 假设或 KEA1 假设和 DL 假设。

证明. 下面证明不存在上述敌手, 为了统一表述我们假定敌手缺少子秘密为 s_{rev} 的属性 Att_{rev}, 敌手伪造可以分为 3 种伪造类型打破方案的不可伪造性: Type-1 的伪造是敌手通过某个成员已有的签名来伪造出未颁发给该成员的属性, 使该成员能够使用伪造的属性生成合法的签名; Type-2 的伪造是在 ODBP 假设下, 敌手试图打开 *List* 中的承诺, 使用被撤销的属性进行签名; Type-3 的伪造是在 KEA1 假设和 DL 假设下, 敌手自己伪造出一个未颁发的属性来生成合法签名。上述 3 种类型的敌手都能够在询问阶段访问预言机 $\mathcal{O}_{SndToH}, \mathcal{O}_{Crpt}, \mathcal{O}_{HSign}, \mathcal{O}_{RevA}$。

Type-1: 对于已有的签名, 属性集中的属性通过 Groth-Sahai 证明系统和基于 DLIN 假设的可验证加密方案分别进行承诺和加密, 因此敌手如果能够解决 DLIN 问题, 则对于已有签名中属性证书的承诺和加密仅能够得到 $g^{x_i s_{rev}}$, 因此敌手还需解决 DL 问题, 即采用类似 Type-3 的伪造方法输出伪造属性集签署的签名 σ^*, 后续证明与 Type-3 伪造一致, 故不再进行赘述。

Type-2: 假设敌手的属性 Att_{rev} 在 $\tilde{\lambda}-1$ 时刻被撤销, 此时进入 $\tilde{\lambda}$ 时间点, 有 $C_{\tilde{\lambda}} = e(g_{val,\tilde{\lambda}-1} / g^{x^* s_{rev}}, h_{ra})$, 群元素 $g^{x^* s_{rev}}$ 不再聚合到聚合值中, 其中 $g_{val,\tilde{\lambda}-1}$ 表示 $\tilde{\lambda}-1$ 时间点下所有合法属性群元素的聚合值; 敌手要生成合法签名, 需要使 $e(\prod_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, h_{ra}) \cdot e(W_{i^*}^*, h) = C_{\tilde{\lambda}}$, 即把承诺值 $C_{\tilde{\lambda}}$ 打开到敌手合法属性的聚合值 $\prod_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}$ 中, 从而找到伪造的 W_i^*。

敌手经过询问后用合法的属性证书 $(\bigcup_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, W_i^*)$ 使 $e(\prod_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, h_{ra}) \cdot e(W_{i^*}^*, h) = C_{\tilde{\lambda}}$ 成立, 从而生成伪造签名 σ^*, 模拟器 $\mathcal{S}$ 使用 *osk* 和 *oask* 可以获得 $(\bigcup_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, W_{i^*}^*)$, $\mathcal{S}$ 选择任意一个 $\tilde{\lambda}$ 时间点可以生成合法签名的群成员 *ih* 的成员证书并更新成员证书, 有 $C_{\hat{\lambda}} = e(\prod_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, h_{ra}) \cdot e(W_{i^*}^*, h) = e(\prod_{j=1}^{|\Gamma_{ih}|} att_{ih,j}, h_{ra}) \cdot e(W_{ih}, h)$, 即 $e(\prod_{j=1}^{|\Gamma_{ih}|} att_{ih,j} / \prod_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, h_{ra}) \cdot e(W_{ih} / W_{i^*}^*, h) = 1 \in G_T$, 有 $(\prod_{j=1}^{|\Gamma_{ih}|} att_{ih,j} / \prod_{j=1, j\neq rev}^{|\Gamma_{i^*}|} att_{i^*,j}, W_{ih} / W_{i^*}^*)$ 作为 ODBP 问题的解。

Type-3: 给定模拟器 $\mathcal{S}$ 一个 DL 问题实例 (g, g^{ξ}), 敌手通过伪造出群组中未颁发的属性来生成签名。

初始化阶段, $\mathcal{S}$ 按照 Setup 和 KeyGen 生成群公钥 *gpk* 和管理员私钥, 不同的是, 此时令 $h = g^{\xi}$, 将 *gpk* 和除属性颁发私钥 *aisk* 外的管理员私钥发送给敌手 $\mathcal{A}$。

询问阶段, $\mathcal{S}$ 按照属性不可伪造性定义回应敌手 $\mathcal{A}$ 的所有询问。

伪造阶段, $\mathcal{A}$ 经过询问过后, 成功伪造出属性 Att_j 对应的群元素 $g^{x^* s_{rev}}$ 并生成合法的群签名, 此时有 $\prod_{j=1, j\neq rev}^{|\phi_i|} e(att_j, h^{\vec{\omega}_j}) \cdot e(att_{rev}, h^{\omega_{rev}}) = e(g_{s_T}, K_3)$, 因此敌手的属性集经过伪造后符合访问结构, 即 $F(\Gamma_{i^*}) = 1$。

由于 $\mathcal{S}$ 掌握 *osk*, $\mathcal{S}$ 可以根据 Open 中的方式打开伪造的签名中的承诺 $C_{5,2}$, 从而得到用于签署签名的 $(K_2, K_3) = (g^{x^*}, h^{x^*})$, $\mathcal{S}$ 输入 $(g, K_2) = (g, g^{x^*})$,

$\mathcal{A}$ 隐式返回 $(h,K_3)=(h,h^{x^*})$, 在 KEA1 假设下存在提取器 $\bar{\mathcal{A}}$ 可以提取指数 $\xi=\log_g h$ 作为 DL 问题的解, 即满足 $g^{\xi}=h$, 由此 $\mathcal{S}$ 解决了 DL 问题。

综上所述, 在 ODBP 假设和 DL 假设以及 KEA1 假设下, 实际并不存在上述敌手, 因此本文方案具有属性不可伪造性。

6.7 属性抗联合攻击性

定理 7. 本文方案具有属性抗联合攻击性, 敌手同样被赋予访问预言机 $\mathcal{O}_{SndToH},\mathcal{O}_{SndToC},\mathcal{O}_{AddU},\mathcal{O}_{RReg},\mathcal{O}_{Crpt},\mathcal{O}_{RevA}$ 的权限。如果存在一个多项式时间的敌手能够打破本文方案的属性抗合谋攻击性, 则存在一个模拟器 $\mathcal{S}$ 可以打破 KEA1 假设和 DL 假设。

证明. 下面证明 $\mathcal{A}$ 无法以压倒性优势打破属性抗合谋攻击性。我们考虑敌手 $\mathcal{A}$ 尝试组合不同成员法属性集并自适应的选择合谋伪造的最简属性集 Γ^* 使 $F(\Gamma^*)=1$, 由于 $att_{i,j}=g^{x_i s_j}$, 因此成员并不知道指数 s_j, 假定当前需要 m 个属性, 要组合不同腐败成员的成员属性 $att_{i,j}$, 使 $K_2^{*s_T}=g^{x^* s_T}=\prod_{j=1}^{m-1} att_{i,j}\cdot g^{x^* s^*}$, 此时相当于需要计算出属性 Att_j 对应的指数 s_j, 在 KEA1 假设和 DL 假设下不存在上述敌手, 具体证明过程与属性不可伪造性类似, 因此本文方案具有属性抗联合攻击性。

7 性能分析和方案对比

在本章中给出了本文方案的仿真实验结果并进行性能分析, 同时在最后与同类方案在功能上进行了对比, 结果表明本文方案相比于同类方案整体较优。在仿真实验中, 实验结果为相关算法执行 1000 次运行时间的平均值以消除误差, 我们的实验环境为 Windows10 操作系统、Intel(R) Core(TM) i5-10500 CPU 3.10 GHz、内存 16GB、开发工具为 IntelliJ IDEA, 使用 JPBC 库[21]在默认的 Type-A 曲线下完成相关实验, Type-A 曲线能够提供 1024 bit 的安全性。出于实验考虑, 使用的变色龙哈希函数为 Hugo 和 Tal 提出的方案[22], 线性秘密分享方案为 LW11[19]提出的方案, 涉及属性数量的实验我们仅在 AND 门限下进行实验。

在表 1 中, 给出了本文方案 Setup、AttGen、Open、Judge、RevA、Fail-Stop 这 6 个恒定执行时间算法的时间开销, 由于 Setup 需要进行群元素的预计算以加快其他算法中的计算, 因此开销较高, 但该算法在方案中仅会执行一次, 其余算法整体执行时间较短, 有较好的表现。

在图 1 的(a)、(b)、(c)中, 我们对算法执行时间与属性集大小相关的 Join、Sign、Verify、Sign-、Verify-、OpenA 这 6 个算法进行了实验, 并与同样在标准模型下的 LHZ15[15]和 XCG21[16]方案进行了对比, 实验中所有方案的验证算法都进行了批处理[23]以减少部分计算开销, 其中 LHZ15 是静态方案且没有实现 OpenA 算法, XCG21 是构建在非对称配对下的方案, 此时 $G_1\neq G_2$, 并且群 G_1,G_2 的指数计算开销与对成配对相比较小。图 1(a)中, 通过与 XCG21 对比, 本文方案的 Join 和 OpenA 在属性集较小的情况下有较好的表现, 同时 XCG21 在签名阶段存在 $\sum_{\phi_i\in\hat{\phi}}|\phi_i|G_2+Z_p^*$ 的通信开销以实现 OpenA, 其中 $\hat{\phi}$ 表示签名成员所有可能使用的属性集, 而本文方案在签名阶段仅在需要更新属性证书时会有一个群 G 元素的开销, 综合来看本文方案在属性集大小较小的情况下优于 XCG21 方案。图 1(b)和图 1(c)中, 由于非对称配对指数计算较快, 并且 XCG21 没有通过承诺的方式来打开属性, 因此签名和验证可以保持恒定的开销, 但本文方案在 Verify 算法的性能上优于 XCG21; LHZ15 由于对于属性需要大量的承诺, 因此在签名和验证方面并不理想。综合来看, 本文方案在属性集较小的情况下具有明显的优势。

最后, 我们在表 2 中给出了与同类方案的对比, 其中 BW07[8]是基于身份的群签名方案, 并且方案是静态的, 不适用于较为复杂的应用场景, AA14[14]虽然是基于属性的群签名方案, 但是构建基于 BMW 模型[17], 并不支持成员安全地动态加入和撤销, 同时方案的访问结构是访问树结构, 在恢复根秘密时需要递归树节点, 而本文采用的是 LSSS 访问结构, 恢复秘密并不涉及递归, 因此本文在恢复属性效率和安全性以及功能性上优于上述方案。对于同样实现 CCA 匿名性的 EMO09[11]和 LHZ15[15], EMO09 方案构建基于随机预言机(Random Oracle Model, ROM)模型, 而本文原始方案基于标准模型, 方案中的哈希函数被认为是真实的哈希函数, 因此本文方案在安全性上优于 EMO09 方案; LHZ15 通过标签加密在标准模型下实现了 CCA 匿名性, 而本文原始方案使用的加密方案是可验证加密方案, 可以在验证阶段对密文进行验证, 整体上方案的安全性更强, 并且 LHZ15 没有实现追踪签名使用的属性集的功能, 因此本文方案在安全性和实现的功能上优于 LHZ15。XCG21[16]与本文方案同样在 LSSS 访问结构下实现

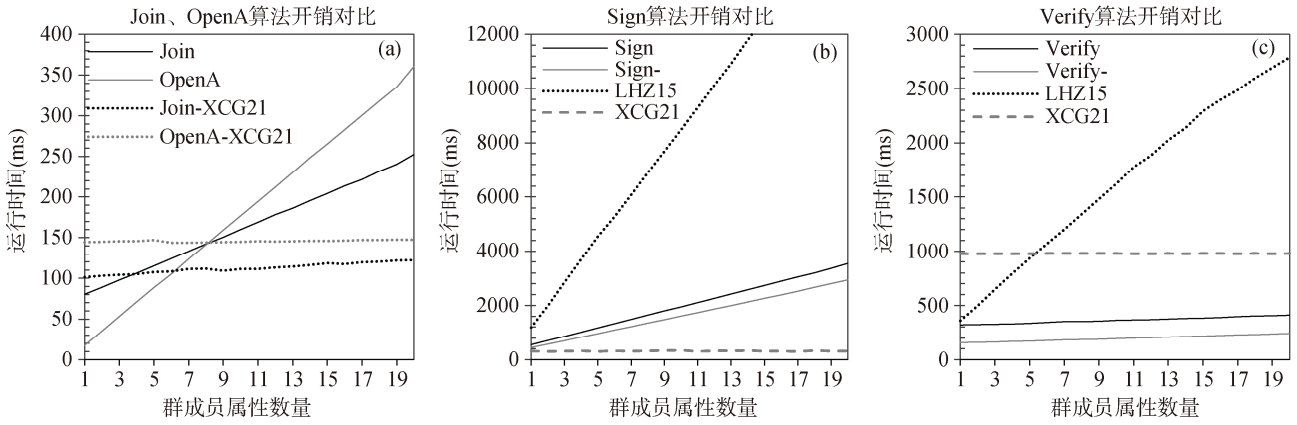

图 1　非恒定执行时间算法开销

Figure 1　Algorithmic overhead for non-constant execution time

表 1　恒定执行时间算法开销

Table 1　Algorithmic overhead for constant execution time

算法	Setup	AttGen	Open	Judge	RevA	Fail-Stop
时间/ms	2023.53	50.03	35.79	4.64	2.58	14.23

表 2　方案对比

Table 2　Comparisons of schemes

方案	匿名性	动态性	属性基	访问结构	打开属性	ROM/SM	失败停止
BW07[8]	CPA	N/A	否	N/A	N/A	SM	否
EMO09[11]	CCA	仅加入	是	访问树	N/A	ROM	否
AA14[14]	CPA	N/A	是	访问树	N/A	SM	否
LHZ15[15]	CCA	N/A	是	LSSS	N/A	SM	否
XCG21[16]	CPA	仅加入	是	LSSS	是	SM	否
Proposed-CCA	CCA	全动态	是	LSSS	是	SM	是
Proposed-CPA	CPA	全动态	是	LSSS	是	SM	是

注: BW07、AA14、LHZ15 都是构建在 BMW03 安全模型下的方案, 因此都是静态方案, 不支持动态加入群成员。

了属性打开功能, 但这是以签名阶段的通信开销为代价实现的, 本文方案在打开阶段仅涉及群元素的指数运算, 打开属性即是打开承诺, 同时 XCG21 并没有实现撤销功能, 因此本文方案与 XCG21 相比表现较优。通过与上述方案相比, 本文方案基于动态聚合器实现了高效的成员属性撤销, 并在标准模型下实现了失败停止功能, 方案能够在遭到无限计算能力的敌手攻击后, 由被伪造的群成员提供证据来证明方案不再安全, 因此本文方案具有更强的安全性。

综上所述, 本文方案具有较好的安全性和功能性, 能够满足成员属性动态变更的应用场景的需要, 整体上优于同类方案。

8　结束语

本文通过 Groth-Sahai 证明系统以及基于变色龙哈希函数的可验证加密方案, 结合了失败停止签名, 实现了在标准模型下具有 CCA 匿名性的失败停止属性基群签名方案, 方案支持高效的成员属性撤销并能够提供证据证明方案当前遭到了无限计算能力的敌手的攻击以实现方案的失败停止。同时方案设置了负责追踪签名属性集的属性打开者, 防止了签名者对属性匿名性的滥用。随后本文出于计算开销和签名大小考虑又对原始方案进行简化, 在简化方案中签名大小和签名以及验证的计算开销大幅减少, 同样能够打开属性和对成员属性动态撤销, 但仅实现了 CPA 匿名性。方案不足之处在于每次撤销和加入都需要所有群成员进行签名时更新属性证书, 更新需要管理员承担计算开销, 增加了管理员的工作量, 并且签名和验证的开销都和属性集大小相关。因此, 提出一个平衡效率和安全性的 CCA 匿名性的方案是接下来的研究目标。

参考文献

[1] Khader D. Attribute based group signatures [EB/OL]. 2007: Cryptology ePrint Archive: 2007/159.

[2] Groth J, Sahai A. Efficient Non-Interactive Proof Systems for Bilinear Groups[C]. *Advances in Cryptology – EUROCRYPT 2008*, 2008: 415-432.

[3] Zhang R. Tweaking TBE/IBE to PKE Transforms with Chameleon Hash Functions[C]. *Applied Cryptography and Network Security*, 2007: 323-339.

[4] Cheng X G, Wang J, Du J X. A new revocation method for standard model group signature[J]. *Journal of Computers*, 2014, 9(5): 1053-1057.

[5] Pedersen T P, Pfitzmann B. Fail-stop signatures[J]. *SIAM Journal*

on Computing, 1997, 26(2): 291-330.

[6] Chaum D, van Heyst E. Group Signatures[M]. Advances in Cryptology — EUROCRYPT '91. Berlin, Heidelberg: Springer, 1991: 257-265.

[7] Bellare M, Shi H X, Zhang C. Foundations of Group Signatures: The Case of Dynamic Groups[C]. *Topics in Cryptology – CT-RSA 2005*, 2005: 136-153.

[8] Boyen X, Waters B. Full-Domain Subgroup Hiding and Constant-Size Group Signatures[C]. *Public Key Cryptography – PKC 2007*, 2007: 1-15.

[9] Groth J. Fully Anonymous Group Signatures without Random Oracles[C]. *Advances in Cryptology – ASIACRYPT 2007*, 2007: 164-180.

[10] Yue X H, Zhou F C, Wang X B. Dynamic group signatures scheme with CCA-anonymity in standard model[J]. *Journal of Chinese Computer Systems*, 2015, 36(1): 138-142.
(岳笑含, 周福才, 王溪波. 一种在标准模型下具有 CCA 匿名性的动态群签名方案[J]. *小型微型计算机系统*, 2015, 36(1): 138-142.)

[11] Emura K, Miyaji A, Omote K. A Dynamic Attribute-Based Group Signature Scheme and Its Application in an Anonymous Survey for the Collection of Attribute Statistics[C]. *2009 International Conference on Availability, Reliability and Security*, 2009: 487-492.

[12] Patel B K, Jinwala D. Anonymity in Attribute-Based Group Signatures[M]. Advanced Computing, Networking and Security. Berlin, Heidelberg: Springer, 2012: 495-504.

[13] Qian Y, Zhao Y M. Strongly Unforgeable Attribute-Based Group Signature in the Standard Model[C]. *2010 IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2010: 843-852.

[14] Ali S T, Amberker B B. Attribute-based group signature without random oracles with attribute anonymity[J]. *International Journal of Information and Computer Security*, 2014, 6(2): 109-132.

[15] Li B H, Huang Y Y, Zhao Y L. Fully adaptive attribute-based group signature in standard model[J]. *Journal of the Chinese Institute of Engineers*, 2015, 38(2): 200-207.

[16] Xu Y L, Chen Y L, Gao S Y. Research on attribute-based group signature scheme supporting LSSS access structure[J]. *Computer Technology and Development*, 2021, 31(9): 92-98.
(许玉岚, 陈燕俐, 高诗尧. 支持 LSSS 访问结构的属性基群签名方案的研究[J]. *计算机技术与发展*, 2021, 31(9): 92-98.)

[17] Bellare M, Micciancio D, Warinschi B. Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions[C]. *Advances in Cryptology — EUROCRYPT 2003*, 2003: 614-629.

[18] Chen J J, Chiang Y Y, Hsu W H, et al. Fail-stop group signature scheme[J]. *Security and Communication Networks*, 2021, 2021(1): 6693726.

[19] Lewko A, Waters B. Decentralizing Attribute-Based Encryption[C]. *Advances in Cryptology – EUROCRYPT 2011*, 2011: 568-588.

[20] Libert B, Peters T, Yung M. Short Group Signatures *via* Structure-Preserving Signatures: Standard Model Security from Simple Assumptions[C]. *Advances in Cryptology— CRYPTO 2015*, 2015: 296-316.

[21] de Caro A, Iovino V. JPBC: Java Pairing Based Cryptography[C]. *2011 IEEE Symposium on Computers and Communications*, 2011: 850-855.

[22] Krawczyk H, Rabin T. Chameleon Hashing and Signatures [EB/OL]. 1998: Cryptology ePrint Archive: 1998/010.

[23] Blazy O, Fuchsbauer G, Izabachène M, et al. Batch Groth–Sahai[M]. Applied Cryptography and Network Security. Berlin, Heidelberg: Springer, 2010: 218-235.

附录 1. 预言机定义

预言机用于安全性证明, 敌手可以根据安全模型中的定义访问部分预言机与模拟器进行交互, 下面给出本文的预言机。

$\mathcal{O}_{AddU}(i,\Gamma_{req,i})$: 如果 i 已经存在于 REG 则返回"⊥", 否则将索引为 i 的群成员执行 Join 算法加入群组并加入诚实成员集合 HU, 此时 $REG[i]$ 有相应信息。

$\mathcal{O}_{RReg}(i)$: 读取 $REG[i]$ 中的成员证书, 如果该项为空则返回"⊥"。

$\mathcal{O}_{WReg}(i,info)$: 在 $REG[i]$ 中写入或修改信息 $info$。

$\mathcal{O}_{CrptU}(i)$: 将诚实的群成员腐化为敌手 $\mathcal{A}$ 可以操纵的腐败的群成员, 从 HU 中移出索引为 i 的群成员, 并将其加入腐败成员集合 CU 中。

$\mathcal{O}_{SndToH}(i)$: 将索引为 i 的腐败成员通过 Join 加入诚实成员集合 HU 中, 此时敌手 $\mathcal{A}$ 仍可获得该成员的签名私钥 usk_i。

$\mathcal{O}_{SndToC}(i)$: 将索引为 i 的腐败成员通过 Join 加入腐败成员集合 CU 中, 此时敌手 $\mathcal{A}$ 仍可获得该成员的签名私钥 usk_i。

$\mathcal{O}_{\_Usk}(i)$: 将索引为 i 的成员的签名私钥 usk_i 作为返回。

$\mathcal{O}_{HSign}(i,msg)$: 签名预言机允许敌手询问索引为 i 的诚实成员在消息 msg 上的签名, 如果该成员的属性集 Γ_i 不符合访问结构, 返回"⊥", 否则返回签名 σ_i。

$\mathcal{O}_{Open}(msg,\sigma)$: 按照 Open 算法打开消息签名对 (msg,σ), 如果无法打开到具体的群成员或者打开的签名是来自 $\mathcal{O}_{Ch}$ 的, 则返回“⊥”。

$\mathcal{O}_{OpenA}(msg,\sigma)$: 按照 OpenA 算法打开消息签名对 (msg,σ), 如果无法打开到具体的属性或者打开的签名是来自 $\mathcal{O}_{ChA}$ 的, 则返回“⊥”。

$\mathcal{O}_{Ch}(i_0,i_1,msg)$: 挑战预言机用于敌手攻击方案的成员匿名性, 敌手输入两个诚实成员的索引 i_0,i_1 和待签名的消息 msg, 这两个成员应具有能够进行签名的属性集, 选择随机比特 $b\in_R\{0,1\}$, 使用成员签名私钥 usk_b 生成索引 i_b 在消息 msg 的签名 σ_b, σ_b 无法再被打开预言机 O_{Open} 询问, 将 σ_b 返回。

$\mathcal{O}_{ChA}(\Gamma_0,\Gamma_1,i,msg)$: 属性挑战预言机用于敌手攻击方案的属性匿名性, 敌手输入索引为 i 的群成员的两个合法的属性集 Γ_0,Γ_1 和待签名的消息 msg, 这两个属性集可以完全相同或不同, 选择随机比特 $b\in_R\{0,1\}$, 选择属性集 Γ_b 在消息 msg 由成员 i 生成签名 σ_b, σ_b 无法再被打开预言机 O_{OpenA} 询问, 将 σ_b 返回。

$\mathcal{O}_{RevA}(i,\Gamma_{rev})$: 撤销 *REG* 列表中索引为 i 的群成员的属性集 Γ_{rev}, 撤销的属性被加入集合 *RA*。

附录 2. 安全模型形式化定义

本文采用实验 Exp 的形式对安全模型进行形式化, 敌手在实验中可以根据安全模型的定义访问部分预言机。

实验 1. 正确性

$\mathrm{Exp}_{\mathrm{A}}^{corr}(t_{sp})$:
$(gpk,isk,osk,oask,iask)\leftarrow\mathrm{Setup}(t_{sp}),\mathrm{AttGen}(params,s_T,$
$AS,\Gamma);\ CU\leftarrow\varnothing,\ HU\leftarrow\varnothing;$
$(i,msg,msg')\leftarrow\mathrm{A}(gpk:\mathrm{O}_{AddU},\mathrm{O}_{RReg});$
If $i\notin HU$ or $usk[i]=\perp$ return 0;
$\sigma\leftarrow\mathrm{Sign}(msg,usk_i,\varphi_i,gpk);$
$(j,\tau)\leftarrow\mathrm{Open}(msg,\sigma,osk,gpk);$
If $\mathrm{Verify}(msg,\sigma,gpk)=0$ return 1;
If $i\neq j$ or $\mathrm{Judge}(i,\tau,gpk)=0$ return 1;
If the following are all true then return 1 else return 0:
1)$\exists\sigma'$ belongs to i on msg', which is legal but not signed by i;
2)$\mathrm{Fail\text{-}Stop}(msg',\sigma',usk_i,\varphi_i,gpk)=0$.

实验 2. 成员匿名性

$\mathrm{Exp}_{\mathcal{A}}^{anon-u}(t_{sp})$:
$(gpk,isk,osk,oask,iask)\leftarrow\mathrm{Setup}(t_{sp}),\mathrm{AttGen}(params,s_T,$
$AS,\Gamma);\ CU\leftarrow\varnothing,\ HU\leftarrow\varnothing;$
$i_{\mathcal{A}}\leftarrow\mathcal{A}(gpk:\mathcal{O}_{Ch},\mathcal{O}_{SndToH},\mathcal{O}_{WReg},\mathcal{O}_{Crpt},\mathcal{O}_{Usk},\mathcal{O}_{Open});$
If $i_{\mathcal{A}}=b$ return 1 else return 0. // $b\in_R\{0,1\}$

实验 3. 可追踪性

$\mathrm{Exp}_{\mathcal{A}}^{trace}(t_{sp})$:
$(gpk,isk,osk,oask,iask)\leftarrow\mathrm{Setup}(t_{sp}),\mathrm{AttGen}(params,s_T,$
$AS,\Gamma);\ CU\leftarrow\varnothing,HU\leftarrow\varnothing;$
$(msg,\sigma)\leftarrow\mathcal{A}(gpk,isk,osk:\mathcal{O}_{SndToC},\mathcal{O}_{AddU},\mathcal{O}_{RReg},\mathcal{O}_{Usk},\mathcal{O}_{Crpt});$
If $\mathrm{Verify}(msg,\sigma,gpk)=0$ then return 0;
$(i,\tau)\leftarrow\mathrm{Open}(msg,\sigma,osk,gpk);$
If $i=\perp$ or $\tau=\perp$ return 1;
If $\mathrm{Judge}(i,\tau,gpk)=0$ return 1 else return 0.

实验 4. 不可陷害性

$\mathrm{Exp}_{\mathcal{A}}^{nf}(t_{sp})$:
$(gpk, isk, osk, oask, iask) \leftarrow \mathrm{Setup}(t_{sp}), \mathrm{AttGen}(params, s_T, AS, \Gamma);\ CU \leftarrow \varnothing, HU \leftarrow \varnothing;$
$(msg, \sigma) \leftarrow \mathcal{A}(gpk, isk, osk : \mathcal{O}_{SndToH}, \mathcal{O}_{WReg}, \mathcal{O}_{Crpt}, \mathcal{O}_{HSign}, \mathcal{O}_{Usk});$
If $\mathrm{Verify}(msg, \sigma, gpk) = 0$ then return 0;
$(i, \tau) \leftarrow \mathrm{Open}(msg, \sigma, osk, gpk);$
If the following are all true then return 1 else return 0:
1) $i \in HU \wedge usk[i] \neq \perp \wedge \mathrm{Judge}(i, \tau, gpk)=1$;
2) $\mathcal{A}$ did not query $\mathcal{O}_{Usk}(i)$ or $\mathcal{O}_{HSign}(i, msg)$;
3) $\mathrm{Fail\text{-}Stop}(msg, \sigma, usk_i, \varphi_i, gpk)=0$.

实验 5. 属性匿名性

$\mathrm{Exp}_{\mathcal{A}}^{anon-A}(t_{sp})$:
$(gpk, isk, osk, oask, iask) \leftarrow \mathrm{Setup}(t_{sp}), \mathrm{AttGen}(params, s_T, AS, \Gamma); CU \leftarrow \varnothing,\ HU \leftarrow \varnothing;$
$i_{\mathcal{A}} \leftarrow \mathcal{A}(gpk : \mathcal{O}_{ChA}, \mathcal{O}_{SndToH}, \mathcal{O}_{WReg}, \mathcal{O}_{Crpt}, \mathcal{O}_{Usk}, \mathcal{O}_{OpenA}, \mathcal{O}_{RevA});$
If $i_{\mathcal{A}} = b$ return 1 else return 0. // $b \in_R \{0,1\}$

实验 6. 属性不可伪造性

$\mathrm{Exp}_{\mathcal{A}}^{nf\text{-}A}(t_{sp})$:
$(gpk, isk, osk, oask, iask) \leftarrow \mathrm{Setup}(t_{sp}), \mathrm{AttGen}(params, s_T, AS, \Gamma); CU \leftarrow i,\ HU \leftarrow \varnothing, RA \leftarrow att_{i,rev};$
If $F(\Gamma_i)$=1 return 0;
$(msg, \sigma) \leftarrow \mathcal{A}(gpk, isk, osk : \mathcal{O}_{SndToH}, \mathcal{O}_{Crpt}, \mathcal{O}_{HSign}, \mathcal{O}_{RevA});$
If $\mathrm{Verify}(msg, \sigma, gpk) = 0$ then return 0;
$\Gamma_j \leftarrow \mathrm{OpenA}(msg, \sigma, gpk);$ If $F(\Gamma_j)$=0 return 1;
If $Att_{i,rev} \in \Gamma_j$ or $Att_k \in \Gamma_j \wedge Att_k \notin \Gamma$ return 1 else return 0.

实验 7. 属性抗联合攻击性

$\mathrm{Exp}_{\mathcal{A}}^{cr\text{-}A}(t_{sp})$:
$(gpk, isk, osk, oask, iask) \leftarrow \mathrm{Setup}(t_{sp}), \mathrm{AttGen}(params, s_T, AS, \Gamma); CU \leftarrow i,\ HU \leftarrow \varnothing, RA \leftarrow \varnothing;$
If $\mathrm{F}(\Gamma_i)$=1 return 0;
$(msg, \sigma) \leftarrow \mathcal{A}(gpk, isk, osk : \mathcal{O}_{SndToH}, \mathcal{O}_{SndToC}, \mathcal{O}_{Crpt}, \mathcal{O}_{HSign}, \mathcal{O}_{RevA});$
If $\mathrm{Verify}(msg, \sigma, gpk) = 0$ then return 0;
$\Gamma_j \leftarrow \mathrm{OpenA}(msg, \sigma, gpk);$
If $F(\Gamma_j)$=1 return 1 else return 0.

廖东旭 于 2022 年在武夷学院小学教育专业获得学士学位。现在华侨大学计算机技术专业攻读硕士学位。研究领域为信息安全。Email: ldxedum@163.com

程小刚 于 2016 年在南京航空航天大学计算机应用技术专业获得博士学位。现任华侨大学计算机科学与技术学院副教授。研究领域为应用密码学、量子密码学。Email: cxg@hqu.edu.cn