基于无证书的子分组多重签名方案

王宇航[1], 徐哲清[1], 王志伟[1], 刘 峰[2]

[1]南京邮电大学计算机学院 南京 中国 210023

[2]中国科学院信息工程研究所 信息安全国家重点实验室 北京 中国 100093

摘要 无证书密码体制是在传统的基于证书的公钥密码体制和身份基的密码体制的基础上提出的一种新的密码体制。无证书密码体制不仅克服了传统公钥密码体制中的证书管理问题, 更是解决了身份基的密码体制中的密钥托管问题, 在巧妙地克服两者缺点的同时, 将两者的优势相结合。在实际应用中, 高效和安全是大家设计方案所需要追求的目标, 因此如何设计出安全高效的无证书密码算法一直是大家关注的焦点。多重签名用于证明一组签名者已对给定消息进行了签名, 其签名长度与签名者数量无关。在区块链等共识场景中, 使用多重签名算法是一种兼顾安全和效率的解决方案。目前, 多重签名被越来越多地应用在区块链等共识场景下, 其优点在于减少区块的存储消耗、正确性验证时间等。然而在共识场景下应用的多重签名方案中默认签名者为诚实实体, 因此当存在“Byzantine 节点”时, 无法保证多重签名的安全有效。为了将无证书密码体制以及多重签名的优势结合起来, 并提高多重签名方案在共识场景下的鲁棒性, 本文提出了一种基于无证书的子分组多重签名方案。该方案中允许群中任意合法子分组代表群产生多重签名, 并在签名聚合前验证所有单个签名的有效性。在本文中, 我们定义了方案的鲁棒性, 并给出了相应的证明; 在随机预言机模型下, 我们证明了本文方案在适应性选择消息攻击下具有不可伪造性。最后, 由效率分析和仿真实验表明, 本文方案中的多重签名生成效率较高。

关键词 无证书; 子分组; 多重签名; 鲁棒性; CDH 假设

中图法分类号 TP309 DOI 号 10.19363/J.cnki.cn10-1380/tn.2024.09.03

A Certificateless Subgroup Multi-Signature Scheme

WANG Yuhang[1], XU Zheqing[1], WANG Zhiwei[1], LIU Feng[2]

[1] School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China

[2] State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China

Abstract The certificateless cryptosystem is a new cryptosystem proposed on the basis of traditional certificate based public key cryptosystems and identity based cryptosystems. The certificateless cryptosystem not only overcomes the certificate management problem in traditional public key cryptosystems, but also solves the key escrow problem in identity-based cryptosystems. While cleverly overcoming the shortcomings of the two, it combines their advantages. In practical applications, efficiency and security are the goals that everyone needs to pursue when designing solutions, so how to design secure and efficient certificateless password algorithms has always been a focus of attention. Multi-signatures are used to prove that a group of signers have signed a given message, and their signature length is independent of the number of signers. In consensus scenarios such as blockchain, using multi-signature algorithms is a solution that balances security and efficiency. Currently, multi-signature is increasingly being applied in consensus scenarios such as blockchain, with the advantages of reducing block storage consumption and correctness verification time. However, in the multi-signature scheme applied in consensus scenarios, the default signer is an honest entity, so when there is a “Byzantine node”, the security and effectiveness of the multi-signature cannot be guaranteed. In order to combine the advantages of certificateless cryptography and multi-signature, and improve the robustness of multi-signature schemes in consensus scenarios, this paper proposes a sub group multi-signature scheme based on certificateless. In this scheme, any legitimate subgroup within the group is allowed to generate multi-signatures on behalf of the group, and the validity of all individual signatures is verified before signature aggregation. In this article, we define the robustness of the scheme and provide corresponding proof; under the random oracle machine model, we prove that the scheme in this paper is unforgeable under the adaptive selection message attack. Finally, efficiency analysis and simulation experiments show that the multi-signature generation efficiency in this scheme is relatively high.

Key words certificateless; subgroups; multi-signature; robustness; Computational Diffie-Hellman(CDH) assumption

通讯作者: 王志伟, 博士, Email: zhwwang@njupt.edu.cn。

本文受国家自然科学基金项目(No. 62372245)、2022 年信息安全国家重点实验室开放课题项目(No. 2022-MS-5)、江苏省研究生科研与实践创新计划项目(No. KYCX22_0987)资助。

收稿日期: 2023-01-18; 修改日期: 2023-04-20; 定稿日期: 2024-06-14

1 背景

Al-Riyami 和 Paterson[1]在 2001 年提出了无证书密码体制, 这种密码体制介于传统的公钥密码体制和身份基密码体制之间。在这种机制中, 用户的私钥由两个部分组成[2]: 一个是从密钥生成中心(key generation center, KGC)中提取的与用户身份信息有关的身份密钥; 另一个是由用户自己随机构造的密钥对, 并且无法从其中一个部分中计算出另一个部分的相关信息。无证书密码体制的出现, 既解决了传统公钥密码体制的证书管理和维护问题, 也解决了身份基密码体制的密钥托管问题, 在巧妙地克服两者缺点的同时, 将两者的优势相结合。因此无证书密码体制在数字签名应用中具有无证书管理、通信开销较低、强不可抵赖性等众多优点[3]。

多重签名是由 Itakura 和 Nakamura[4]提出的, 指一组签名者合作产生对同一个消息的签名。多重签名验证签名的时间与单个签名验证时间相同, 多重签名的长度与签名者数量无关, 并使得多重签名验证者确信每一个签名者都参与了消息 m 的签名, 具有更加安全和多样化管理的特点。2003 年, Dan 等人[5]基于 BLS 签名提出一个非交互式的多重签名方案; 2004 年, Chunbo 等人[6]结合向量空间秘密共享以及变色龙哈希函数, 提出了一个新的基于双线性的变色龙多重签名方案, 并对方案的安全性进行了分析; Komano 等人[7]在 2006 年给出了多重签名方案的安全定义和安全模型, 并对方案进行了安全性证明。此外, 在区块链等需要协同合作的应用场景下, 使用多重签名是兼顾安全性与实施效率的一种解决方案。在 2018 年, Boneh 等人[8]基于区块链提出了一种紧密的多重签名方案, 在方案中通过对聚合公钥进行处理, 从而使得验证时的运算效率得到了显著的提高。Drijvers M 等人[9]在 2019 年提出了一种可证明安全的前向多重签名方案, 该方案的效率较高, 仅需要一个指数计算和三个双线性配对计算来实现签名的验证; 同年, Gregory 等人[10]基于 Schnorr 签名提出了一种可证明安全的多重签名方案, 并将该方案应用于区块链中, 有效地提高了比特币的性能。在 2021 年, Kojima R 等人[11]在 Gregory 等人方案的基础上提出一个具有密钥聚合的多签名方案,该方案不仅解决了动态签名的聚合, 更是在验证算法中仅使用了单个聚合公钥, 提高了验证算法的效率。但是以上的方案都是基于传统的公钥密码体制设计的, 一旦签名用户不断的增多, 就会存在大量证书的管理和维护问题。

为了将无证书密码体制以及多重签名的优势结合起来, 在 2008 年, 梁红梅等人[12]首次提出无证书多重签名方案的定义以及敌手模型, 并基于双线性配对设计了一个无证书多重签名方案, 方案解决了基于传统公钥签名方案存在的证书管理问题以及基于身份签名方案所存在的密钥管理的问题, 并根据随机预言机模型对方案的安全性进行了证明; 2013 年, 秦艳琳等人[13]提出了一个基于无证书的有序多重签名方案, 该方案计算效率较高, 但是存在安全漏洞; 2015 年, 杜红珍等人[14]对秦的方案进行了改进, 改进后的方案可以有效地抵御任何敌手的攻击, 同时只需要两次双线性配对计算就可以验证签名的正确性, 提高了运算效率。但是上述基于无证书的多重签名方案无法抵御“Byzantine 攻击”, 无法安全高效地应用于分布式环境中。

在分布式环境中,现有的多重签名方案机制在生成多重签名之前没有明确地对单个签名的合法性进行验证或者筛选[15-16]。这种默认来自共识节点的单个签名合法的机制是不合理的, 因为如果参与多重签名的节点中存在“Byzantine 节点”[17-18], 那么方案的安全性就无法得到保障。因此, 为了有效地解决上述问题, Alexandra 在 2002 年首次提出了基于子分组的多重签名方案[19], 该方案允许群中任意合法子分组代表群产生多重签名, 因此对于消息 m 来说, 签名者是群组的一个子分组, 而且该子分组是不确定的, 但是该方案对子分组多重签名方案的定义以及安全模型并不完整; 2019 年 Elrond 等[20]利用 BLS 签名以及 bitmap, 提出了一种基于子分组的多重签名方案, 但是缺乏完善的安全性证明, 并且该方案无法抵御流氓密钥攻击; 2020 年 Galindo 等人[21]给出了多重签名方案的鲁棒性定理, 提出了一个基于子分组的多重签名方案, 并对方案的安全性以及鲁棒性进行了完整的证明, 然而该方案使用的是传统公钥密码体制, 需要较大的证书管理和维护的开销; 2022 年田陈等人[22]提出了一个基于子分组的身份基多重签名方案, 并对方案的鲁棒性和不可伪造性进行了详细的证明, 尽管该方案使用的是基于身份的密码体制, 能够很好地解决了证书管理等开销, 但是还会存在密钥托管的问题。

为了更好地解决文献[22]中存在的密钥托管问题, 我们提出了一种基于无证书的子分组多重签名方案, 改进了原方案的签名算法, 该方案在无证书密码体制的基础上, 结合了子分组多重签名方案的

优势, 提高了实际的应用效率[23]; 此外, 本文定义并证明了方案的鲁棒性, 并在随机预言机模型下, 证明了该方案满足适应性选择消息攻击下的不可伪造性。

2 基础知识

2.1 双线性映射

令 q 为大素数, 定义 G 是阶为 q 的循环乘法群, 生成元为 g, G_T是阶为 q 的循环乘法群。假设 G 中的离散对数假设是困难的, 定义映射 $e{:}G\times G\to G_T$, 该映射满足以下性质:

(1) 双线性: 对于$\forall u,v\in G,\forall a,b\in Z_p$, 均有$e(u^a,v^b)=e(u,v)^{ab}$成立

(2) 非退化性: $\exists u,v\in G$, 使得 $e(u,v)\neq 1_{G_T}$

(3) 可计算性: 对于$\forall u,v\in G$, 均可计算 $e(u,v)$

2.2 困难问题

定义 1(离散对数假设)。对于一个阶为 q 的乘法循环群 $G=<g>$, 我们定义敌手 A 的优势 $Adv^{dl}{}_G$为:

$$Adv^{dl}{}_G=Pr[y=g^x:y\xleftarrow{\$}G,x\xleftarrow{\$}A(y)]$$

这里的概率取决于 A 的随机选取, y 的随机选取。如果敌手 A 至多在时间τ内, 攻破方案的优势为 $Adv^{dl}{}_G\geqslant\varepsilon$, 那么就说明敌手 $A(\tau,\varepsilon)$攻破了离散对数假设。如果不存在这样的敌手, 那么就称离散对数假设是(τ,ε)困难的。

定义 2(计算性-Diffie-Hellman(CDH)假设)。对于一个乘法循环群 $G=<g>$, 给定 $g,g^{\alpha},g^{\beta}\in G$, 称计算$g^{\alpha\beta}\in G$ 为计算性-Diffie-Hellman 假设。定义敌手 A 的优势为 $Adv^{CDH}{}_G$:

$$Adv^{CDH}{}_G=Pr[y=g^{\alpha\beta}:(\alpha,\beta)\xleftarrow{\$}Z_q,y\leftarrow A(g^{\alpha},g^{\beta})]$$

这里的概率取决于 A 的随机选取, 以及(α,β)的随机选择。如果敌手 A 至多在时间τ内, 攻破方案的优势为 $Adv^{CDH}{}_G\geqslant\varepsilon$, 那么就称敌手 $A(\tau,\varepsilon)$攻破了 CDH 假设。如果不存在这样的敌手, 则称 CDH 假设是(τ,ε)困难的。

2.3 广义的分叉引理

2000 年, Pointcheval 和 Stem 等人[24]提出了一般签名体制的概念, 并为了证明随机预言机模型下数字签名方案的安全性, 他们提出了包含随机预言机的分叉引理 (The Forking Lemma)。随后, 在 2008 年, Bagherzandi 等人[25]对分叉引理进行了扩展, 使得扩展后的分叉引理也可以用于其他类型数字签名的安全性证明, 下面对扩展的分叉引理进行介绍。

设 A 是一种算法, 它在输入时与随机预言机 $H{:}\{0,1\}^*\to Z_q$交互。令 $f=(\rho,h_1,...,h_{q_H})$是 A 算法的随机输入, 其中ρ是 A 的随机磁带, h_i是 A 对 H 的第 i 个查询的响应, q_H是 A 询问 H 的最大次数。设Ω是所有这类输入f的空间, $f|_i=(\rho,h_1,...,h_{i-1})$。我们定义 $A(in,f)$来表示在算法 A 中输入 in 和随机参数f的过程。如果 $A(in,f)$输出$\left(J,\{out_j\}_{j\in J}\right)$, 则代表成功, 其中$J$是$\{1,...,q_H\}$的一个子集, $\{out_j\}_{j\in J}$是其他输出的集合。如果 A 输出$\emptyset$, 表示算法 A 失败。对于新的随机性 $f\xleftarrow{\$}\Omega$和由输入生成器 IG 生成的 $in\xleftarrow{\$}IG$, 我们定义 $A(in,f)$成功的概率是ε。

对于一个给定的输入 in, 扩展的分叉引理算法 GF_A定义如下:

$GF_A(in)$:

$f=(\rho,h_1,...,h_{q_H})\xleftarrow{\$}\Omega$

$\left(J,\{out_j\}_{j\in J}\right)\leftarrow A(in,f)$

IF $J=\emptyset$ THEN output fail

LET $J=j_1,...j_n$ such that $j_1\leqslant...\leqslant j_n$

FOR $i=1,...,n$ DO

$succ_i\leftarrow 0;k_i\leftarrow 0;$

$kH\ ln(8n/\varepsilon)_{\max}$

REPEAT until $succ_i=1$ or $k_i>k_{\max}$

$f''\xleftarrow{\$}\Omega$ such that $f'|_{j_i}=f|_{j_i}$

LET$f''=(\rho,h1,...,h_{j_i-1},h''_{j_i},h''_{q_H})$

$\left(J'',out''_{j_{\in J''}}\right)\leftarrow A(in,f'')$

IF $h''_{j_i}\neq h_{j_i}$ and $J''\neq\emptyset$ and $j_i\in J''$

$out'_{j_i}\leftarrow out''_{j_i};\ succ_i\leftarrow 1$

IF $succ_i=1$ for all $i=1,...,n$

output $\left(J,\{out_j\}_{j\in J},\{out'_j\}_{j\in J}\right)$

ElSE output fail

此外 Bagherzandi 等人还证明了该分叉算法的下列引理。

定义 1 (扩展的分叉引理)。设 IG 是一个随机算法, A 是在时间τ上运行的随机化算法, 并且该算法至多进行q_H次随机预言机查询, 最终以ε的概率成功。如果 $q>8nq_H/\varepsilon$, 那么 $GF_A(in)$的运行次数至多为 $\tau\cdot 8n^2q_H/\varepsilon\cdot ln(8n/\varepsilon)$, 并且至少会以$\varepsilon/8$ 的概率成功, 此时成功的概率取决于 $in\xleftarrow{\$}IG$ 和 IG。

3 方案的定义以及安全模型

在本节中, 我们主要介绍了基于无证书的子分组多重签名方案的定义与安全模型。我们通过允许攻击者参与签名聚合的过程来描述鲁棒性的概念。随后, 我们在一个子分组模型中定义了不可伪造性, 最终攻击者输出一个伪造的(J,σ^*), 其中 J 是攻击者选择的签名者子分组的身份集合, σ^*是一个伪造的多重签名。

3.1 方案的定义

基于无证书的子分组多重签名方案主要包括: 密钥生成中心 KGC, n 个群成员 $U=\{u_1,u_2,...,u_n\}$ (对应的身份集合为 $ID_G=\{ID_1,ID_2,...,ID_n\}$)。该方案主要包含以下几个算法: 系统设置算法, 用户密钥生成算法, 部分私钥提取算法, 群公钥生成算法, 聚合公钥生成算法, 群成员签名算法, 聚合签名算法, 多重签名验证算法。

(1) 系统设置算法(Setup): 输入安全参数λ, 输出系统的主私钥 msk, 主公钥 $mpk=g^{msk}$, 以及系统参数 $Params$。

(2) 用户密钥生成算法(KeyGen): 输入安全参数 κ, 生成公私钥对$(sk,pk=g^{sk})$。

(3) 部分私钥提取算法(Extract-Private-Key): 输入群成员u_i的身份 ID_i、系统参数 $Params$ 和主私钥 msk, KGC 计算群成员u_i的身份私钥d_i, 并将d_i发送给群成员u_i。

(4) 群公钥生成算法(GroupSet): 输入群成员身份集合 ID_G, KGC 计算群的唯一标签 $gtag$, 生成群公钥 $gpk=(gtag,ID_G)$。

(5) 群成员签名算法(Sign): 由签名者子分组中的成员执行。输入签名消息 m, 群标签 $gtag$, 群成员的身份私钥d_i, 群成员的私钥 sk_i, 以及系统参数 $Params$, 输出每个签名者u_i对消息 m 的签名$S_i=(S_{1i},S_{2i})$。

(6) 聚合签名算法(Combine): 输入系统参数 $Params$, 签名者公钥集合 $PK=\{pk_j\}_{j\in J}$, 签名者子分组 J, 以及当前签名者子分组产生的签名集合 $\{S_j\}_{j\in J}$, 输出产生的多重签名σ。

(7) 聚合公钥生成算法(KeyAgg): 输入签名者子分组J和签名者公钥集合$PK=\{pk_j\}_{j\in J}$, 输出聚合公钥 $apk=(apk_1,apk_2)$。

(8) 多重签名验证算法(VerifaMul): 输入群公钥 gpk, 签名者子分组 J, 消息 m, 系统主公钥以及多重签名σ。验证者对多重签名的合法性进行验证, 若签名合法, 输出 1; 反之输出 0。

3.2 安全模型

基于无证书的子分组多重签名方案的安全性主要分为正确性, 鲁棒性和不可伪造性。

定义 1(正确性)。按照签名算法正确生成的基于无证书的子分组多重签名一定能够通过验证算法。

定义 2(鲁棒性)。攻击者无法将若干个合法的成员签名聚合成一个非法的多重签名。本方案的鲁棒性由以下三阶段的博弈定义:

(1) 初始阶段(Steup): 挑战者产生系统参数 $Params$, 选定主公钥 y 与挑战身份 ID^*, 生成一组密钥对(sk^*,pk^*), 并将$(Params,ID^*,pk^*)$发给敌手 A。

(2) 查询阶段(Queries): 在任意的群标签 $gtag$ 下, 敌手 A 能够询问预言机 O, 得到身份 ID 对任意消息 m 的签名。

(3) 输出阶段(Output): 最后, 挑战者会接收到攻击者 A 输出的群公钥 gpk, 身份集合 $ID_G=\{ID_1,ID_2,...,ID_n\}$, 消息 m^*, 公钥集合 $PK=\{pk_1,pk_2,...,pk_n\}$, 以及一组来自于$|I|$个不同的签名者生成的签名集合$\xi=S_{i_{i\in I}}$, 其中 $I\subseteq U$。

此时, 规定如果以下三个条件成立, 就认为敌手 A 破坏了方案的鲁棒性:

(1) $gpk\neq\perp$且 $gpk=GroupSet(ID_G)$

(2) $pk^*=pk_k$且 $k\in U, k\notin I$

(3) $VerifyMul(m^*,J,\sigma^*,gpk)=0$, 其中$S^*$是敌手$A$以$ID^*$的身份查询$(m^*,gtag^*)$得到的签名,$J$是签名者子分组, σ^*是$\xi\cup S^*$生成的多重签名, 并且$J\backslash\{k\}\neq\emptyset$。

如果敌手 A 至多运行的时间为τ, 至多进行q_S次签名查询和q_H次随机预言机查询, 并且至少以ε的概率输出 1, 那么就称算法 $A(\tau,q_S,q_H,\varepsilon)$攻破了方案的鲁棒性。如果不存在这样的敌手, 那么本文方案是$(\tau,q_S,q_H,\varepsilon)$鲁棒的。

定义 3(不可伪造性)。敌手伪造出一个能够通过验证算法的基于无证书的子分组多重签名在计算上是不可行的。

在无证书的签名方案中, KGC 可以通过系统设置算法获得系统主密钥, 因此 KGC 能够计算出所有签名者的身份私钥, 无证书签名的安全模型需要考虑恶意但是被动的 KGC 进行的攻击。此外签名者的公钥没有经过可信第三方的认证, 所以在安全模型中也需要考虑攻击者利用他自己选择的不合法的公钥来替换签名者的公钥。因此, 该安全模型中存在两类攻击者[26]。

1. 攻击者A_1: 恶意用户, 能够替换其他用户的公钥, 但是不知道系统主私钥;

2. 攻击者A_2: 恶意的 KGC, 它知道系统主密钥, 但是不能对签名者的公钥进行替换。

对于敌手A_1, 本文方案的不可伪造性通过挑战者 B 和攻击者A_1之间的游戏来定义:

Game1: 挑战者 B 和攻击者A_1执行挑战问询:

(1) 初始阶段: 挑战者 B 产生系统参数 $Params$, 以及系统主私钥 msk, 并将 $Params$ 发给敌手A_1。

(2) $Hash$ 查询: A_1可以访问方案中所有的 $Hash$ 预言机。

(3) 部分私钥提取查询: A_1询问用户 ID_i的部分私钥, B 运行部分私钥提取算法获取身份私钥d_{ID_i}返回给A_1。

(4) 秘密值查询: A_1询问用户 ID_i的秘密值, 最终会获得 B 返回的秘密值x_i, x_i是 B 运行用户密钥生成算法的输出。如果 ID_i的公钥已被执行过公钥替换算法, 则输出⊥。

(5) 替换公钥查询: A_1能用自己选取的公钥 pk'_i 替换 ID_i的公钥 pk_i。

(6) 签名查询: 在任意的群标签 $gtag$ 下, A_1可以询问身份 ID_i对任意消息 m 的签名。

(7) 输出阶段: 攻击者A_1输出伪造的签名σ^*以及对应的群公钥 gpk, 签名者子分组 J 和消息m^*。

规定若敌手A_1的伪造输出满足以下三个条件时, 则认为敌手A_1伪造成功:

(1) $gpk \neq \perp$且 $gpk = GroupSet(ID_G)$;

(2) 签名者身份 ID^*没有同时提交给公钥替换查询和部分私钥提取查询;

(3) 攻击者A_1不能直接询问$(m^*,gtag^*)$的签名, 而伪造的多重签名(J,σ^*)是有效的, 即 $VerifyMul(m^*, J,\sigma^*, gpk)=1$。

对于敌手A_2, 本文方案的不可伪造性通过挑战者 B 和攻击者A_2之间的游戏来定义:

Game2: 挑战者 B 和攻击者A_2执行挑战问询

(1) 初始阶段: 挑战者 B 产生系统参数 $Params$, 以及系统主私钥 msk, 并将 $Params$ 以及 msk 发给敌手A_2。

(2) $Hash$ 查询: A_2可以访问方案中所有的 $Hash$ 预言机。

(3) 秘密值查询: A_2询问用户 ID_i的秘密值, 最终会获得 B 返回的秘密值x_i, x_i是 B 运行用户密钥生成算法的输出。

(4) 签名查询: 在任意的群标签 $gtag$ 下, A_2可以询问身份 ID_i对任意消息 m 的签名。

(5) 输出阶段: 攻击者A_2输出伪造的签名σ^*以及对应的群公钥 gpk, 签名者子分组 J 和消息m^*。

规定若敌手A_2的伪造输出满足以下三个条件时, 则认为敌手A_2伪造成功:

(1) $gpk \neq \perp$且 $gpk = GroupSet(ID_G)$;

(2) 签名者身份 ID^*没有同时提交给公钥替换查询和部分私钥提取查询;

(3) 敌手A_2不能直接询问$(m^*,gtag^*)$的签名, 而伪造的签名(J,σ^*)是有效的, 即 $VerifyMul(m^*, J, \sigma^*, gpk)=1$。

如果敌手A_1在 Game1 中获胜的概率以及敌手A_2在 Game2 中获胜的概率都是可忽略的, 那么该方案在适应性选择消息攻击下具有不可伪造性。

4 基于无证书的子分组多重签名方案

在本节中, 我们提出了基于无证书的子分组多重签名方案的具体构造。该方案是以 sakai 签名[27]以及 BLS 签名[28]为基础, 并基于子分组设计的, 允许任意合法子分组代表群产生签名, 因此对于消息 m 来说, 签名者所在的子分组 J 是不确定的。此外, 为了使得密钥管理更加方便, 高效, 我们使用了无证书的密码体系来构造子分组多重签名方案。

下面给出基于无证书的子分组多重签名方案, 该方案由 8 个多项式时间算法组成:

(1) 系统设置算法(Setup): 令(q,G,G_T,e,g)是素数阶为 q 的双线性群, $g \in G$。假定哈希函数H_1:$\{0,1\}^* \to G$, H_2:$\{0,1\}^* \to Z_q$, H_3:$\{0,1\}^* \to Z_q$, H_4:$\{0,1\}^* \to G$。KGC 选择系统主私钥 $msk \in Z_q$, 计算系统主公钥 $mpk = g^{msk}$。发布系统参数 $Params$=$\{G, G_T, e, q, g, H_1, H_2, H_3, H_4, mpk\}$, 消息空间为 $m \in \{0,1\}^*$。

(2) 用户密钥生成算法(KeyGen):用户随机选择 $sk \xleftarrow{\$} Z_q$, 计算 $pk = g^{sk}$, 随后输出公私钥对(sk,pk)。

(3) 部分私钥提取算法 (Extract-Private-Key): KGC 通过群成员u_i的身份 ID_i, 计算出u_i的身份私钥 $d_i = H_1(ID_i)^{msk}$, 并将d_i发送给群成员u_i。

(4) 群公钥生成算法(GroupSet): 输入群成员的身份集合 ID_G, KGC 计算 $gtag = H_2(ID_G)$, 令 $gpk = (gtag,ID_G)$。

(5) 群成员签名算法(Sign): 群中的签名发起者确定签名者子分组为 $J = \{ID_1,ID_2,...,ID_j\}$, 并向群成员发送$(m, J)$。属于签名者子分组 J 中的成员u_j接收到(m, J)后, 使用自己的身份 ID_j, 以及自己的私钥 sk_j, 执行以下步骤:

a. u_j执行部分私钥提取算法, 生成身份私钥d_j;

b. u_j计算$S_{1j} = H_4(m,gtag)^{r_j + sk_j} \cdot d_j$;

c. u_j选取随机数$r_j \in Z_q$, 计算$S_{2j} = g^{r_j}$;

d. u_j输出对消息 m 的签名$S_j = (S_{1j},S_{2j})$, 再将签名S_j发送给签名聚合者。

(6) 聚合签名算法(Combine): 当签名聚合者接收所有签名者对消息 m 的签名$S_j = (S_{1j},S_{2j})$时, 执行以下步骤生成多重签名:

a. 检验接收到的所有签名S_j是否是合法签名。若下列等式:

$$e(S_{1j},g)\left(H_4(m,gtag),S_{2j}\right)\left(H_4(m,gtag),pk_j\right) \cdot e\left(H_1(ID_j),mpk\right)$$

成立, 则签名S_j是合法的, 否则该签名是非法签名。当所有的签名都是合法的时候, 才会执行下一步的聚合操作, 否则退出聚合签

名算法, 返回群成员签名算法, 此时签名发起者会重新选择子分组进行签名;

b. 计算$a_j = H_3(ID_j,pk_j,ID_G,J)$;

c. 计算$\sigma_1 = \prod_{j\in J} S_{1j}^{a_j}$, $\sigma_2 = \prod_{j\in J} S_{2j}^{a_j}$, 生成多重签名$\sigma = (\sigma_1,\sigma_2)$。

(7) 聚合公钥生成算法(KeyAgg): 输入签名者子分组 J 和签名者公钥集合 PK, 签名聚合者计算 $a_j = H_3(ID_j,pk_j,ID_G,J)$, $apk_1 = \prod_{j\in J} H_1(ID_j)^{a_j}$, $apk_2 = \prod_{j\in J} pk_j^{a_j}$, 生成聚合公钥$apk = (apk_1,apk_2)$。

(8) 多重签名验证算法(VerifyMul): 验证者输入群公钥 gpk, 签名者子分组 J, 消息 m, 系统主公钥以及多重签名$\sigma = (\sigma_1,\sigma_2)$, 通过以下步骤验证多重签名的合法性:

a. 执行 KeyAgg 算法, 生成聚合公钥: $apk = (apk_1,apk_2)$;

b. 验证下列等式是否成立:

$$e(\sigma_1,g) = e(H_4(m,gtag),\sigma_2) \cdot e(H_4(m,gtag),apk_2)\cdot e(apk_1,mpk)$$

若等式成立, 则表示签名合法, 输出 1; 反之输出 0。

5 安全性分析

5.1 正确性

若基于无证书的子分组多重签名是正确的按照签名算法实现的, 则以下两个等式必然成立:

(1) 群标签为 $gtag$ 的情况下, 每个签名者u_j对消息 m 的签名$S_j = (S_{1j},S_{2j})$满足以下验证等式:

$$\begin{aligned} e(S_{1j},g) &= e\left(H_4(m,gtag)^{r_j+sk_j}\cdot d_j,g\right)\\ &= e\left(H_4(m,gtag)^{r_j+sk_j},g\right)\cdot e(d_j,g)\\ &= e\left(H_4(m,gtag),g^{r_j+sk_j}\right)\cdot e\left(H_1(ID_j)^{msk},g\right)\\ &= e\left(H_4(m,gtag),S_{2j}\right)\cdot e\left(H_4(m,gtag),pk_j\right)\\ &\quad\cdot e\left(H_1(ID_j),mpk\right)\end{aligned}$$

(2) 群标签为 $gtag$ 的情况下, 本文方案生成的多重签名$\sigma = (\sigma_1,\sigma_2)$满足以下等式:

$$\begin{aligned} e(\sigma_1,g) &= e\left(\prod_{j\in J} S_{1j}^{a_j},g\right)\\ &= e\left(\prod_{j\in J} H_4(m,gtag)^{(r_j+sk_j)a_j},g\right)\cdot e\left(\prod_{j\in J} d_j^{a_j},g\right)\\ &= e\left(\prod_{j\in J} d_j^{a_j},g\right)\cdot e\left(H_4(m,gtag),\prod_{j\in J} g^{r_j\cdot a_j}\right)\\ &\quad\cdot e\left(H_4(m,gtag),\prod_{j\in J} g^{sk_j\cdot a_j}\right)\\ &= e(H_4(m,gtag),\sigma_2)\cdot e(H_4(m,gtag),apk_2)\\ &\quad\cdot e(apk_1,mpk)\end{aligned}$$

5.2 鲁棒性

定理 1. 基于无证书的子分组多重签名方案具有鲁棒性。

证明: 假设存在攻击者 $A(\tau,q_S,q_H,\varepsilon)$能够破坏基于无证书的子分组多重签名方案的鲁棒性, 因此该方案生成的多重签名可能为非法签名。根据 3.2 节定义的鲁棒性安全模型可知, 攻击者最后会输出$\left(m^*,\{pk_j\}_{j\in J'},gpk = (gtag^*,ID_G),\{S_j\}_{j\in J}\right)$, 并且挑战公钥 pk^*在公钥集合中被表示为 pk_k。若攻击者成功输出了该结果, 即方案的聚合签名算法生成了非法的多重签名σ^*, 因此以下验证等式: $e(\sigma_1^*,g) = e(H_4(m^*,gtag^*),\sigma_2^*)\cdot e(H_4(m^*,gtag^*),apk_2)\cdot e(apk_1,mpk)$不成立。所以$\{S_j\}_{j\in J}$中一定存在一个非法签名$S_i$, 使得验证等式 $e(g,S_{i1}) = e(H_4(m^*,gtag^*),S_{i2})\cdot e(H_4(m^*,gtag^*),pk_i)\cdot e(H_1(ID_i),mpk)$不成立。此时存在如下两种情形:

(1) 若 $i = k$, 即签名S_k非法。由于$S_{k2} = g^{r_k}$且 $S_{k1} = H_4(m^*,gtag^*)^{r_k+sk_k}\cdot d^*$, $d^* = H_1(ID_k)^{msk}$, 因此, 签名S_k是正确的按照签名算法实现的, 根据方案的正确性可知, 签名S_k一定是一个合法签名, 此时与签名S_k非法矛盾, 因此该情形不成立。

(2) 若 $i \neq k$, 即签名S_i非法。根据方案的聚合签名算法, 生成多重签名之前会验证所有的单个签名S_i的合法性, 这与签名S_i非法矛盾, 因此该情形不成立。

在生成多重签名过程中, 假设参与签名的诚实实体与“Byzantine 叛徒”均在组群中,它们有着自己随机选择的公私钥对, 身份私钥由 KGC 参与生成。此外, 群组中任意成员都能够轻易的验证其他成员所生成的签名的合法性。签名的发起者首先会选择任意子分组代表群生成多重签名, 当签名收集者收到子分组中所有成员的单个签名时, 首先会验证每个签名的合法性, 若含有非法签名, 则终止算法, 签名者重新选择子分组生成多重签名。只有当所有单个签名都是合法时, 才会执行签名聚合算法, 因此可以有效的抵御 “Byzantine 攻击”, 提高基于无证书的子分组多重签名方案的鲁棒性。

综上可知, 不存在这样的攻击者 $A(\tau,q_S,q_H,\varepsilon)$能够破坏基于无证书的子分组多重签名方案的鲁棒性, 因此基于无证书的子分组多重签名方案具有鲁棒性。

5.3 不可伪造性

定理 2. 在随机预言机模型下, 基于无证书的子分组多重签名方案的不可伪造性是依赖于 CDH 假设以及 BLS 签名的不可伪造性。

证明: A_1, A_2是攻击者算法, B 是以A_1, A_2为子程序的算法, 同时 B 也是 BLS 签名算法的攻击者, F

是CDH假设的挑战者, C是BLS签名算法的挑战者。H_1,H_2,H_3,H_4是随机预言机, B给定参数$(G,G_T,q,g,g^{\alpha},g^{\beta})$, 其中$G=<g>$是循环群, 其阶为素数$q$, $\alpha,\beta\xleftarrow{\$}Z_q^*$。挑战者$F$的目标是利用扩展的分叉引理, 运行算法$GF_B$解决CDH假设, 即计算$g^{\alpha\beta}$的值。挑战者$C$的目标是利用算法$B$伪造出一个合法的BLS签名。

对于攻击者A_1, 挑战者B设定α为系统主私钥, $y=g^{\alpha}$为系统主公钥。挑战者B设定挑战身份ID^*, 同时B需要回答A_1的签名询问与$Hash$询问。B选择系统参数$Params=\{G,G_T,e,q,g,y,H_1,H_2,H_3,H_4\}$, 并将系统参数发送给$A_1$。

对于攻击者A_2, 挑战者B设定msk为系统主私钥, $y=g^{msk}$为系统主公钥。B设定挑战身份ID^*, 同时B需要回答A_2的签名询问与$Hash$询问。选择系统参数$Params=\{G,G_T,e,q,g,y,H_1,H_2,H_3,H_4\}$, 发送系统参数以及系统主私钥$msk$给$A_2$。

以下定义B与A_1, A_2之间的问询:

(1) B回答A_1关于H_3的参考随机向量为$\boldsymbol{f}=(\rho,c_1,\ldots,c_{qH})$。

(2) H_1查询: B定义一个列表$L_{H_1}=\{t,c,x,h\}$, 初始化为$\emptyset$。A_1, A_2询问t对应的$Hash$值, 若$(t,c,x,h)\in L_{H_1}$, 则输出h作为回答; 反之先选择随机数$x\in\{0,1\}$, 再选择随机数$c\xleftarrow{\$}Z_q$, 若$x=0$, 则设$h=g^c$, 若$x=1$, 则返回$\perp$, 每次回答后更新$L_{H_1}=L_{H_1}\cup\{(t,c,x,h)\}$。

(3) 针对攻击者A_1的部分私钥提取查询: A_1询问身份ID时, 先调用H_1预言机查询L_{H_1}中的(t,c,x,h)。若$x=0$, 则令$d_{ID}=ID^c$作为身份私钥; 若$x=1$, 则返回$\perp$。

(4) H_2查询: B定义一个列表$L_{H_2}=\{t,r\}$, 初始为$\emptyset$。A_1, A_2询问t对应的$Hash$值, 若$(t,r)\in L_{H_2}$, 则输出r作为回答; 否则选取随机数$r\xleftarrow{\$}Z_q$作为回答; 每次回答后更新$L_{H_2}=L_{H_2}\cup\{(t,r)\}$。

(5) 针对攻击者A_1的H_3查询: B定义一个列表$L_{H_3}=\{t,c\}$, 初始为$\emptyset$。A_1, A_2第i次查询t对应的$Hash$值, 若$(t,c)\in L_{H_2}$, 则输出c作为回答; 否则根据t的内容来决定如何回答A_1, A_2。若$ID_j=ID^*$, $t=(ID_j,pk_j,ID_G,J)$, 且$ID^*\in J$时, 回答$H_3(t)=c_i$; 否则回答$H_3(t)=r$, 其中$r\xleftarrow{\$}Z_q$。每次回答后更新$L_{H_3}=L_{H_3}\cup\{(t,c)\}$。

(6) 针对攻击者A_2的H_3查询: B定义一个列表$L_{H_3}=\{t,r\}$, 初始化为$\emptyset$。A_2第i次查询t对应的$Hash$值, 若$(t,r)\in L_{H_2}$, 则输出r作为回答; 否则根据t的内容来决定如何回答A_2。若$t=(ID_j,pk_j,ID_G,J)$, $ID_j\in J$时, 令$H_3(t)=r$, 其中$r\xleftarrow{\$}Z_q$, 否则返回$\perp$作为回答。每次回答后更新$L_{H_3}=L_{H_3}\cup\{(t,r)\}$。

(7) H_4查询: B定义一个列表L_{H_4}=$\{t,\lambda,H\}$, 初始为$\emptyset$。A_1, A_2第i次查询t对应的$Hash$值, 若$(t,\lambda,H)\in L_{H_4}$, 则输出H作为回答; 否则选择随机数$\lambda\xleftarrow{\$}Z_q$, 计算$H$=$g^{\lambda}$作为回答。每次回答后更新$L_{H_4}$=$L_{H_4}\cup\{(t,\lambda,H)\}$。

(8) 秘密值查询: B定义一个列表$L_{sk}=\{ID_i,sk_i,pk_i\}$, 初始为$\emptyset$。$A_1$, A_2查询ID_i的秘密值, 若$ID_i\neq ID^*$, B选择随机数$sk_i\in Z_q$, 计算$pk_i=g^{sk_i}$, 将sk_i作为回答返回给A_1, A_2; 否则, 如果$ID_i=ID^*$, 将$pk_i=g^{\beta}$, $sk_i=\perp$。每次回答后更新列表$L_{sk}=L_{sk}\cup\{(ID_i,sk_i,pk_i)\}$。

(9) 针对攻击者A_1的替换公钥查询: 当B接收到A_1的替换查询(ID_i,pk_i')时, 若$ID_i=ID^*$, 终止游戏; 否则, B从列表L_{sk}中找到(ID_i,sk_i,pk_i), 将pk_i替换成pk_i', 并令$sk_i=\perp$。

(10) 回答签名查询: A_1, A_2询问t对应的签名, 会根据t的内容来决定如何回应A_1, A_2。若$t=(ID,gtag,m)$, 且$ID\in J$, 当$ID=ID^*$时, 返回$\perp$; 否则查找L_{H_1}, 获取ID对应的回答h, 随后进行部分私钥提取查询, 获取ID对应的身份私钥d_{ID}; 查找L_{sk}, 获取ID对应的用户秘密值sk; 查找L_{H_4}, 获取t对应的回答H。选择随机数$\delta\xleftarrow{\$}Z_q$, 计算$S_1=H^{\delta+sk}\cdot d_{ID}$, $S_2=g^{\delta}$, 令$S=(S_1,S_2)$作为签名回答。

最终, 攻击者A_1, A_2都会输出签名者子分组$J=\{ID_1,ID_2,\ldots,ID_j\}$, 签名者公钥集合$PK=\{pk_1,pk_2,\ldots,pk_j\}$, 群公钥$gpk=(gtag^*,ID_G)$, 伪造的多重签名$\sigma^*$以及消息$m^*$。伪造者$A_1$, A_2不能直接询问(m^*,ID^*)的签名, 而伪造的签名(J,σ^*)是有效的。

对于攻击者A_1, 规定若列表$L_{H_1}=\{t,c,x,h\}$中挑战身份ID^*对应的$x=0$, 则终止算法B。由于x是随机选择的, 因此B成功运行的概率为$\frac{1}{2}$。设$pk^*=pk_k$, j_f是$H_3(ID_j,pk_j,ID_G,J)$在f中的下标, 即$H_3(ID_j,pk_j,ID_G,J)=c_{j_f}$; $a_j=H_3(ID_j,pk_j,ID_G,J)$。因此, 最后$B$的输出表示为$\left(\{j_f\},\left\{\left(\sigma^*,ID_G,J,apk_1,apk_2,\{a_j\}_{j\in J}\right)\right\}\right)$, 此时, B成功输出的概率为$\varepsilon/2$。

挑战者F运行算法GF_B来求解CDH假设, 根据广义的分叉引理算法, 运行GF_B的输出结果为$(\{j_f\},\{out\},\{out'\})$。两次运行$GF_B$使用的随机向量$\boldsymbol{f}$

与$f^{'}$虽不同, 但是仍然满足$f|_{j_f}=f^{'}|_{j_f}$。在GF_B的输出结果中有$out=\left(\sigma,ID_G,J,apk_1,apk_2,\{a_j\}_{j\in J}\right)$, $out^{'}=\left(\sigma^{'},ID_G^{'},J^{'},apk_1^{'},apk_2^{'},\left\{a_j^{'}\right\}_{j\in J^{'}}\right)$,其中$\sigma=(\sigma_1,\sigma_2)$, $\sigma^{'}=\left(\sigma_1^{'},\sigma_2^{'}\right)$。假设两次运行$GF_B$的分叉设置为$a_k=c_{j_f}$、$a_k^{'}=c_{j_f}^{'}$, 因此$a_k\neq a_k^{'}$。而签名者群组是固定的, 因此$ID_G=ID_G^{'}$且$J=J^{'}$。所以, 除了$a_k$以外其他的$j\in J$均满足$a_j=a_j^{'}$, 因此$apk_1/apk_1^{'}=H_1(ID_j)^{a_k-a_k^{'}}$, $apk_2/apk_2^{'}=(pk_k)^{a_k-a_k^{'}}$。

算法GF_B输出的签名σ与$\sigma^{'}$都是合法签名, 故有以下两个等式成立:

$$e(\sigma_1,g)=e\left(H_4(m,gtag),\sigma_2\right)\cdot e\left(H_4(m,gtag),apk_2\right)\cdot e\left(apk_1,mpk\right)$$

$$e\left(\sigma_1^{'},g\right)=e\left(H_4(m,gtag),\sigma_2^{'}\right)\cdot e\left(H_4(m,gtag),apk_2^{'}\right)\cdot e\left(apk_1^{'},mpk\right)$$

根据双线性映射性质, 有:

$$\begin{aligned}e\left(g,\sigma_1/\sigma_1^{'}\right)&=e\left(g^{\lambda^*},\sigma_2/\sigma_2^{'}\right)\cdot e\left(g^{\lambda^*},\left(pk^*\right)^{a_k-a_k^{'}}\right)\cdot e\left(H_1\left(ID^*\right)^{a_k-a_k^{'}},mpk\right)\\&=e\left(g^{\lambda^*},\sigma_2/\sigma_2^{'}\right)\cdot e\left(g^{\lambda^*},g^{\beta\left(a_k-a_k^{'}\right)}\right)\cdot e\left(g^{\beta c\left(a_k-a_k^{'}\right)},g^{\alpha}\right)\\&=e\left(g^{\lambda^*},\sigma_2/\sigma_2^{'}\right)\cdot e\left(g^{\lambda^*+\alpha c},g^{\beta\left(a_k-a_k^{'}\right)}\right)\end{aligned}$$

可得:

$$e\left(g,\left(\sigma_1/\sigma_1^{\prime}\right)\cdot\left(\sigma_2^{\prime}/\sigma_2\right)^{\lambda^*}\right)=e\left(g,g^{\left(a_k-a_k^{\prime}\right)\left(\lambda^*\beta+\alpha\beta c\right)}\right)$$

最终可得:

$$g^{\alpha\beta}=\left(\left(\sigma_1/\sigma_1^{\prime}\right)\cdot\left(\sigma_2^{\prime}/\sigma_2\right)^{\lambda^*}/\left(g^{\beta}\right)^{\lambda^*\left(a_k-a_k^{\prime}\right)}\right)^{1/\left(c\left(a_k-a_k^{\prime}\right)\right)}$$

因此, 挑战者F最终能够成功计算出CDH假设的解。但是在多项式时间内, CDH假设是困难的, 此时出现矛盾。因此不存在这样的攻击者A_1, 即基于无证书的子分组多重签名方案对于A_1类敌手, 在适应性选择消息攻击下具有不可伪造性。

对于攻击者A_2, 此时, 设$ID^*=ID_k$,$S^*=S_k$, 对于算法B而言, B可以利用σ^*计算出BLS签名: $S_{BLS}^*=(\sigma_1^*/\prod_{i\in J,i\neq k}S_{1i}^{a_i})^{\frac{1}{a_k}}$, $pk_{BLS}^*=pk_k\cdot(\sigma_2^*/\prod_{i\in J,i\neq k}S_{2i}^{a_i})^{\frac{1}{a_k}}$, 将$\left(S_{BLS}^*,pk_{BLS}^*\right)$发送给挑战者$C$。

显而易见, 如果敌手A_2输出的基于无证书的子分组多重签名σ^*是合法的, 那么计算出来的BLS签名S_{BLS}^*也是一个合法的签名。因此, 本文方案的安全性依赖于BLS签名的安全性, 而BLS签名已在随机预言模型下证明了在适应性选择消息攻击下具有不可伪造性[28], 所以本文的基于无证书的子分组多重签名方案对于A_2类敌手在适应性选择消息攻击下也是不可伪造的。

综上所述, 本文提出的基于无证书的子分组多重签名方案在适应性选择消息攻击下具有不可伪造性。

6 效率分析

本文从理论与实验仿真两个方面对基于无证书的子分组多重签名方案进行效率分析, 并将本文方案与文献[6,10-11]中的多重签名方案以及文献[22]中的基于子分组的身份基多重签名方案进行比较。

6.1 理论分析

表1中列出本文方案与文献[6,10-11,22]中多重签名方案的效率对比, 其中n表示多重签名方案中子分组内成员个数, Mul表示群乘法运算, P表示双线性配对运算, $|G|$表示群上一个点的长度, b表示Z_q群中单个元素的长度, m表示一个schnorr签名的长度。

表1 效率对比

Table 1 Efficiency comparison

方案	私钥长度	公钥长度	签名长度	签名算法开销	验证算法开销	生成多重签名的通信开销
文献[6]	b	$\|G\|$	$\|G\|+b$	$4nP+6nMul$	$2(n+1)P+2nMul$	0
文献[10]	b	$\|G\|$	$\|G\|+b$	$5nMul$	$4nMul$	nm
文献[11]	b	$\|G\|$	$2\|G\|+b$	$8nMul$	$5nMul$	$2nm$
文献[22]	$\|G\|$	$\|G\|$	$2\|G\|$	$3nP+2nMul$	$3P+2nMul$	0
本文方案	$\|G\|+b$	$2\|G\|$	$2\|G\|$	$4nP+4nMul$	$4P+4nMul$	0

分析表中数据, 本文方案与文献[22]中的子分组多重签名方案相比, 尽管本文方案中使用的密钥长度较长, 签名算法以及多重签名验证算法的时间开销也略高, 但是文献[22]中使用的是基于身份的密码体制, 存在密钥托管的问题, 而本文使用的是无证书密码体制, 能够有效地解决密钥托管问题; 本文方案与文献[6,10-11,22]中的多重签名方案相比, 本文方案的签名长度比文献[11]中方案的签名长度更短; 签名算法的开销与验证算法的开销比文献[6]中的开销更小; 并且从通信开销方面来看, 本文方案的开销远低于文献[10-11]中的交互式方案。此外, 由于本文方案使用的是基于子分组的多重签名方案, 能够有效地抵御“Byzantine攻击”, 因而本文方案的

安全性更高。

6.2 实验分析

实验仿真使用的是 PyPBC 密码库来实现本文方案与文献[22]中的方案, 并统计相关的运行时间, 最终结果如图 1～图 2 所示。本次仿真实验的环境如下: CPU 为 Intel i7-12700H, 内存为 8G 的笔记本电脑, 操作系统为 Ubuntu20.0.4, 使用的曲线为 *A* 类曲线。

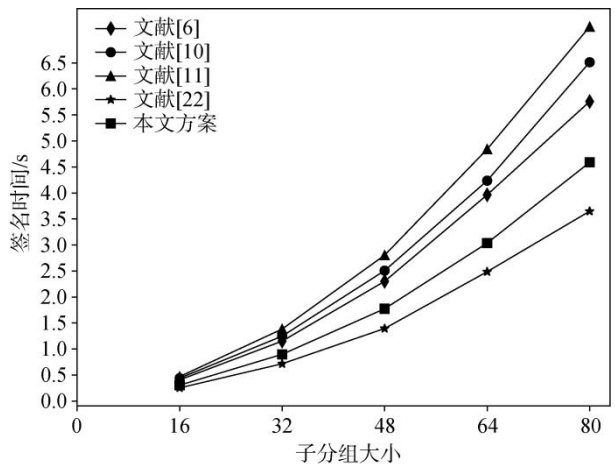

图 1 多重签名生成时间开销对比

Figure 1 Comparison of time cost for multi-signature generation

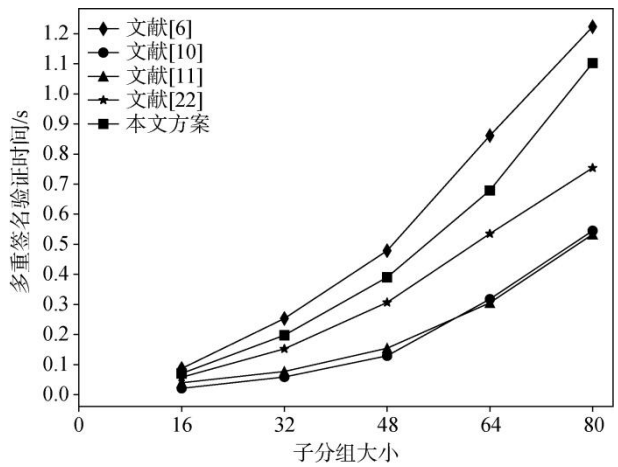

图 2 多重签名验证时间开销对比

Figure 2 Comparison of time cost for multi-signature verification

从图 1 来看, 本文方案的多重签名生成效率较高, 仅次于文献[22]的身份基方案, 优于文献[6,10-11]中的方案。这是由于文献[6]中的方案使用了较多的双线性运算, 生成多重签名时的开销高于本文方案; 文献[10-11]中的交互式方案存在一定的通信开销, 导致方案在生成多重签名时的整体开销要高于本文方案。

从图 2 来看, 本文方案的多重签名验证开销较高, 仅低于文献[6]。这个代价是因为本文的方案在功能上更加完善, 相较于文献[10-11]而言, 本文方案中引入了子分组的概念, 灵活性更高; 相较于文献[22]而言, 本文方案使用了无证书的密码体制, 在实际应用中免除了证书管理的开销。

7 总结

为了提高多重签名在共识机制应用场景下的鲁棒性, 并简化实际应用中证书的管理和维护问题, 本文将无证书的密码体制与子分组多重签名结合起来, 提出了一种基于无证书的子分组多重签名方案, 并对方案的鲁棒性以及不可伪造性进行了证明。本文提出的多重签名方案是以文献[27]中的 sakai 签名以及文献[28]中的 BLS 签名为基础, 并基于子分组设计的。将本文方案与文献[6,10-11,22]中的多重签名方案进行比较, 在实际应用中, 本文的方案相较于文献[6,10-11]而言, 生成多重签名时的开销更小, 并且安全性更高; 与文献[22]中的方案相比较, 尽管本文方案在签名算法以及多重签名验证算法上的计算开销都略高, 但是由于本文使用的是无证书密码体系, 在保证本文方案与文献[22]中的基于子分组的身份基多重签名方案具有相同安全性的同时, 有效地解决了基于身份的密码体制中存在的密钥托管问题, 提高了实际的应用效率。

未来, 我们将对方案的验证效率进行优化, 尝试使用服务器辅助验证的方法提高方案的验证效率, 并将本文的方案实际应用于区块链等共识场景下。

参考文献

[1] Al-Riyami S S, Paterson K G. Certificateless Public Key Cryptography[M]. Advances in Cryptology - ASIACRYPT 2003. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003: 452-473.

[2] Chen Y C, Tso R, Susilo W, et al. Certificateless Signatures: Structural Extensions of Security Models and New Provably Secure Schemes[J]. *IACR Cryptol EPrint Arch*, 2013: 193.

[3] Wang H W, Wang L L, Zhang K, et al. A Conditional Privacy-Preserving Certificateless Aggregate Signature Scheme in the Standard Model for VANETs[J]. *IEEE Access*, 2022, 10: 15605-15618.

[4] ITAKURA K, NAKAMURA K. A public-key cryptosystem suitable for digital multisignatures[J]. *NEC Research and Development*, 1983, 71(71): 474-480.

[5] Boneh D, Gentry C, Lynn B, et al. Aggregate and Verifiably Encrypted Signatures from Bilinear Maps[M]. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003: 416-432.

[6] Ma C B, He D K. A New Chameleon Multi-Signature Based on Bilinear Pairing[C]. *Grid and Cooperative Computing*, 2004: 329-334.

[7] Komano Y, Ohta K, Shimbo A, et al. Formal Security Model of Multisignatures[M]. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006: 146-160.

[8] Boneh D, Drijvers M, Neven G. Compact Multi-Signatures for Smaller Blockchains[C]. *International Conference on the Theory and Application of Cryptology and Information Security*, 2018: 435-464.

[9] Drijvers M, Neven G. Forward-Secure Multi-Signatures[J]. *IACR Cryptol EPrint Arch*, 2019: 261.

[10] Maxwell G, Poelstra A, Seurin Y, et al. Simple Schnorr Multi-Signatures with Applications to Bitcoin[J]. *Designs*, *Codes and Cryptography*, 2019, 87(9): 2139-2164.

[11] Kojima R, Yamamoto D, Shimoyama T, et al. A Novel Scheme of Schnorr Multi-Signatures for Multiple Messages with Key Aggregation[C]. *Broad-Band Wireless Computing, Communication and Application*, 2021: 284-295.

[12] Liang H M, Huang H, Wu C H, et al. A Certificateless Multisignature Scheme[J]. *Journal of Jimei University* (*Natural Science*), 2008, 13(2): 127-131.
(梁红梅, 黄慧, 吴晨煌, 等. 无证书多重签名[J]. *集美大学学报(自然科学版)*, 2008, 13(2): 127-131.)

[13] Qin Y L, Wu X P. Efficient Certificateless Sequential Multi-Signature Scheme[J]. *Journal on Communications*, 2013, 34(7): 105-110.
(秦艳琳, 吴晓平. 高效的无证书有序多重签名方案[J]. *通信学报*, 2013, 34(7): 105-110.)

[14] Du H Z, Wen Q Y. Improved Certificateless Sequential Multi-Signature Scheme[J]. *Journal on Communications*, 2015, 36(10): 56-61.
(杜红珍, 温巧燕. 改进的无证书有序多重签名方案[J]. *通信学报*, 2015, 36(10): 56-61.)

[15] Maxwell G, Poelstra A, Seurin Y, et al. Simple Schnorr Multi-Signatures with Applications to Bitcoin[J]. *Designs*, *Codes and Cryptography*, 2019, 87(9): 2139-2164.

[16] Yu H F, Fu S F, Liu Y X, et al. Certificateless Broadcast Multisignature Scheme Based on MPKC[J]. *IEEE Access*, 2020, 8: 12146-12153.

[17] Shi E. Streamlined Blockchains: A Simple and Elegant Approach (a Tutorial and Survey)[C]. *Advances in Cryptology – ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security*, 2019: 3-17.

[18] ZHAI R, CHEN X B. Research on Blockchain Consensus Mechanism [J]. *Frontiers of Data &Computing,* 2021, 3(3): 86-94.

[19] Boldyreva A. Threshold Signatures, Multisignatures and Blind Signatures Based on the Gap-Diffie-Hellman-Group Signature Scheme[M]. Public Key Cryptography — PKC 2003. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002: 31-46.

[20] TEAM E. Elrond: A highly scalable public blockchain via adaptive state shardingand secure proof of stake [EB/OL]. https://elrond.com/assets/files/elrond-whitepaper.pdf.

[21] Galindo D, Liu J. Robust Subgroup Multi-Signatures for Consensus[C]. *Cryptographers' Track at the RSA Conference*, 2022: 537-561.

[22] Tian C, Wang Z W. Robust Subgroup ID-Based Multi-Signature Scheme[J]. *Computer Science*, 2022, 49(12): 346-352.
(田陈, 王志伟. 基于子分组的身份基多重签名方案[J]. *计算机科学*, 2022, 49(12): 346-352.)

[23] Shamir A. Identity-Based Cryptosystems and Signature Schemes[M]. Advances in Cryptology. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007: 47-53.

[24] Pointcheval D, Stern J. Security Arguments for Digital Signatures and Blind Signatures[J]. *Journal of Cryptology*, 2000, 13(3): 361-396.

[25] Bagherzandi A, Cheon J H, Jarecki S. Multisignatures Secure under the Discrete Logarithm Assumption and a Generalized Forking Lemma[C]. *The 15th ACM conference on Computer and communications security*, 2008: 449-458.

[26] Thorncharoensri P, Susilo W, Baek J. Aggregatable Certificateless Designated Verifier Signature[J]. *IEEE Access*, 2020, 8: 95019-95031.

[27] SAKAI R, OHGISHI K, KASAHARA M. Cryptosystems based on pairing [C]. *The 2000 Symposium on Cryptography and Information Security.*2000:354-368.

[28] Boneh D, Lynn B, Shacham H. Short Signatures from the Weil Pairing[J]. Journal of Cryptology, 2004, 17(4): 297-319.

王宇航 于 2021 年在海南大学信息安全专业获得学士学位。现在南京邮电大学网络空间安全专业攻读硕士学位。研究兴趣包括: 多重签名、聚合签名、区块链。Email: 2510923748@qq.com

徐哲清 于 2021 年在南京邮电大学应用物理专业获得学士学位。现在南京邮电大学网络空间安全专业攻读硕士学位。研究兴趣包括: 区块链、安全多方计算。Email: 1021041521@njupt.edu.cn

王志伟 于 2009 年在北京邮电大学密码学专业获得博士学位。现任南京邮电大学计算机学院, 软件学院, 网络空间安全学院教授。CCF 高级会员, 研究领域为: 云/雾计算安全、区块链、密码协议等。Email: zhwwang@njupt.edu.cn

刘峰 于 2009 年在中科院软件所获得博士学位。现任中国科学院信息工程研究所研究员, 博士生导师。研究领域为: 信息安全体系与战略、网络攻防演化理论、视觉安全理论与技术。Email: fengliu.cas@gmail.com