软件学报 ISSN 1000-9825, CODEN RUXUEW
2025,36(8):3883−3895 [doi: 10.13328/j.cnki.jos.007255] [CSTR: 32375.14.jos.007255]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

加权门限 SM2 签名方案*

唐长虹[1], 赵艳琦[1], 杨晓艺[1], 冯 琦[2], 禹 勇[3]

[1](西安邮电大学 网络空间安全学院, 陕西 西安 710121)

[2](武汉大学 国家网络安全学院, 湖北 武汉 430072)

[3](陕西师范大学 计算机科学学院, 陕西 西安 710119)

通信作者: 禹勇, E-mail: yuyong@snnu.edu.cn

摘 要: 随着物联网和移动互联网技术的发展, 各类移动终端设备被接入互联网中. 当对移动终端设备进行识别和认证时, 通常需要验证其提交的数字签名. 但移动终端设备本身的计算能力受限, 往往采用软件模块来保存密钥至本地或者智能芯片中, 增加了密钥泄露的风险. 现实应用中多采用门限数字签名来抵抗这一攻击, 借助多方合作来分散风险, 提升设备可用性. SM2 数字签名算法是我国自主研发的椭圆曲线公钥密码算法, 于 2016 年成为国家密码标准, 被广泛应用于政府部门、金融机构、电子认证服务提供商等领域. 设计高可用的门限 SM2 数字签名备受关注, 但这类方案的构造依旧较少, 同时也缺乏对参与者权重的考量. 因此, 提出更加灵活的加权门限 SM2 数字签名方案. 在加权门限 SM2 数字签名中签名者分配不同权重, 之后多个签名者共同生成一个有效的签名. 在方法上, 基于中国剩余定理的加权门限秘密共享将 SM2 数字签名的秘钥进行分割. 参与者不只是单一的达到门限值就可以得到签名密钥, 而需要通过计算参与者权重之和, 并达到相应的秘密门限值 t 和重构门限 T, 才能了解到密钥的部分信息或者恢复出签名密钥. 在秘密分割时, 对 SM2 数字签名算法的签名私钥进行变形, 以完成签名阶段对 SM2 密钥进行求逆的这一操作. 最后, 将所提方案与门限 SM2 签名以及联合 SM2 签名等已有工作进行分析比较, 该算法在提升 SM2 签名方案功能性的同时进一步降低了计算开销.

关键词: SM2; 加权门限; 中国剩余定理; 秘密共享

中图法分类号: TP309

中文引用格式: 唐长虹, 赵艳琦, 杨晓艺, 冯琦, 禹勇. 加权门限SM2签名方案. 软件学报, 2025, 36(8): 3883–3895. http://www.jos.org.cn/1000-9825/7255.htm

英文引用格式: Tang CH, Zhao YQ, Yang XY, Feng Q, Yu Y. Weighted Threshold SM2 Signature Scheme. Ruan Jian Xue Bao/Journal of Software, 2025, 36(8): 3883–3895 (in Chinese). http://www.jos.org.cn/1000-9825/7255.htm

Weighted Threshold SM2 Signature Scheme

TANG Chang-Hong[1], ZHAO Yan-Qi[1], YANG Xiao-Yi[1], FENG Qi[2], YU Yong[3]

[1](School of Cyberspace Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China)

[2](School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China)

[3](School of Computer Science, Shaanxi Normal University, Xi'an 710119, China)

Abstract: As the Internet of Things and mobile Internet technologies continue to advance, a wide range of mobile devices are connected to the Internet. To identify and authenticate these devices, it is necessary to verify the digital signatures they submit. However, many mobile devices have limited computing power and typically use software modules to store keys locally or on smart chips, which increases

* 基金项目: 国家重点研发计划 (2022YFB2701500); 国家自然科学基金 (61872229, U19B2021, 62202375, 62202339); 陕西省杰出青年基金 (2022JC-47); 陕西省科学技术协会青年人才托举计划 (20220134); 陕西省重点研发计划重点产业创新链 (群)(2024GX-ZDCYL-01-09, 2024GX-ZDCYL-01-15); 陕西省教育厅科学研究项目 (22JK0557)
收稿时间: 2024-01-04; 修改时间: 2024-04-03; 采用时间: 2024-07-08; jos 在线出版时间: 2024-12-31
CNKI 网络首发时间: 2025-01-02

the risk of key exposure. To avoid this risk, threshold signatures are commonly employed in real-world applications. These signatures rely on multi-party cooperation to decentralize risks and enhance device availability. The SM2 digital signature algorithm, an elliptic curve public key cryptographic algorithm developed independently by China, was adopted as the national cryptography standard in 2016. It finds extensive use in various sectors including government agencies, financial institutions, and electronic authentication service providers. While there has been interest in constructing SM2 threshold signatures with high availability, there are still limited schemes available, and participant weights have not been adequately considered. This study proposes a flexible SM2 weighted threshold signature scheme. In this scheme, signers are assigned different weights, and multiple signers collaborate to generate a valid signature. The key of the SM2 digital signature is divided based on the weighted threshold secret sharing of the Chinese remainder theorem. Participants do not acquire a signing key only by meeting the threshold value. They have to meet the corresponding secret threshold t and the reconstruction threshold T by calculating the sum of the weights of participants to obtain part of the key information or recover the signing key. During secret segmentation, the private signing key of the SM2 digital signature algorithm is transformed to complete the inversion of the SM2 key during the signing stage. Finally, the proposed scheme is compared with other schemes such as SM2 threshold signatures and joint SM2 signatures. The proposed scheme not only reduces computational overhead but also enhances the functionality of the SM2 signature.

Key words: SM2; weighted threshold; Chinese remainder theorem (CRT); secret sharing

随着物联网和移动互联网技术的发展, 各类移动终端设备被接入互联网中, 给人们学习、工作带来了极大的便利. 当对移动终端设备进行识别和认证时, 通常需要验证其提交的数字签名. 基于椭圆曲线密码体制 (elliptic curve cryptography, ECC)[1]构造的数字签名算法, 如: ECDSA、EdDSA、Schnorr、SM2 等签名由于其计算量小、节省存储空间、低带宽等优势被广泛应用. 其中, SM2[2] 椭圆曲线公钥密码算法是国家密码管理局于 2010 年发布《SM2 椭圆曲线公钥密码算法》(GM/T 0003-2012)[3]中提出的, 并于 2016 年成为国家密码标准. SM2 算法在安全性上采用了符合当前密码学标准的 128 位安全强度, 并提供了数字签名、密钥交换协议和公钥加密等功能, 有效满足了政府机构和金融行业等领域对国产密码算法的需求. 在现实应用中, 移动终端设备本身的计算能力受限, 往往采用软件模块来保存密钥至本地或者智能芯片中, 这增加了密钥泄露的风险. 为防范密钥泄露的风险, 多个基于 SM2 的功能型数字签名被提出来, 其借助多方合作分散风险, 提升了算法可用性.

在 SM2 分布式签名方面, 林璟锵等人[4]设计了两方协同的 SM2 签名方法, 提高密钥的安全性, 同时减少了云计算中通信双方的交互. 随后, 文献 [5–7] 针对不同应用场景, 设计出两方参与以及多方参与的 SM2 分布式签名方法, 但这些方法需要所有参与者同时在线参与, 且任何一方密钥丢失都会导致签名无法成功进行. Zhang 等人[8]借助 Paillier 同态加密设计出 SM2 签名算法的两方协同方案, 但该方案需要复杂的范围证明. 侯红霞等人[9]提出了两方协作 SM2 数字签名算法, 该算法同样使用同态加密的方法, 但公私钥生成阶段耗时过长. 冯琦等人[10]设计轻量级的 SM2 两方协同签名协议, 有效降低计算开销, 提高了签名效率, 但这一方式只适用于两个参与方. 苏吟雪等人[11]提出了基于 SM2 的双方共同签名协议, 在共同签名的过程中只需要一次标量乘法计算, 此过程减少计算开销却增加了通信开销. 唐张颖等人[12]提出了基于同态加密的 SM2 门限签名, 然而签名时需要对线性的乘法份额进行传输, 增加通信开销. Han 等人[13]利用 Beaver 乘法提出的两方 SM2 协议具有更低的计算成本, 提高了算法效率, 但两方需要同时在线参与. 针对需要参与方同时在线的问题, 尚铭等人[14]提出了门限的 SM2 算法, 其利用门限的性质使得签名具有更好的健壮性. 此外, 涂彬彬等人[15]对同态加密乘法运算进行拓展, 设计出 SM2 的分布式签名进一步降低了密钥泄露的风险. 在设计分布式 SM2 签名过程中通常采用基于秘密共享的方法. 秘密共享 (secret sharing) 是一种密钥管理的有效方法, 是门限密码系统的关键技术. 其基本思想是将一个秘密信息分割成多个部分, 分发给不同的参与者, 只有当足够数量的部分被合并时, 才能够还原出原始的秘密[16,17]. 1979 年, Shamir[18]提出一种基于拉格朗日插值多项式的 (t,n) 门限秘密共享方案, Blakley[19]利用映射几何理论提出了另一种 (t,n) 门限秘密共享方案. 1983 年, Asmuth 等人[20]利用一次同余方程组将秘密恢复出来, 提出基于中国剩余定理 (Chinese remainder theorem, CRT)[21]的门限秘密共享方案. 同年, Mignotte 等人[22]提出了基于矩阵变换的秘密共享方案. 在金融机构的交易中通常需要多个管理者的签名才能生效. 或者在公司组织中需要多个高层管理人员对重要文件进行签名确认, 以确保签名的合法性和可信度, 能够提供更高的安全性和防护, 避免了单一签名者的风险. Morillo 等人[23]为了解决现实应用中参与者权利可能不同的问题, 对加权的门限秘密共享开展研究, 要求只有在参与者权重

之和大于或等于选定的门限值时才能恢复出秘密. 之后 Beimel 等人[24,25]利用单调电路和单调公式得到每个加权门限访问结构的秘密共享方案. Iftene 等人[26]在 Asmuth 等人[20]方案的基础上提出了加权门限秘密共享方案, 提高了密钥管理的安全性以及签名的安全性. Chaidos 等人[27]在门限签名方案中考虑了委员会的方法, 但成本高且易受到适应性的腐败攻击. 接着 Garg 等人[28]基于中国剩余定理的秘密共享构造出加权斜坡秘密共享方案 (weighted ramp secret-sharing scheme, WRSS), 并将该方案应用在 MPC、加密和签名.

加权门限数字签名可满足多方参与的资金转账、数字证券交易等金融应用安全, 也可应用于多方参与的区块链共识机制中, 确保区块的验证过程的可信度和安全性. 为进一步推动信息系统的自主可控安全, 本文探索加权门限 SM2 数字签名构造方法, 并给出具体构造, 以期扩展国密 SM2 算法的应用范围. 在方法上, 本文基于文献 [28] 中的高效 WRSS 技术, 将 SM2 数字签名的秘钥进行分割. 参与者不只是单一的达到门限值就可以得到签名密钥, 而需要通过计算参与者权重之和, 并达到相应的秘密门限值 t 和重构门限 T, 才能了解到密钥的部分信息或者恢复出签名密钥. 在秘密分割时, 本文对 SM2 签名算法的签名私钥进行变形, 以完成签名阶段对 SM2 密钥进行求逆的这一操作. 最后, 将本文方案与门限 SM2 签名以及联合 SM2 签名等已有工作进行分析比较, 本文在提升 SM2 签名方案功能性的同时进一步降低了计算开销.

本文第 1 节首先回顾中国剩余定理、基于中国剩余定理的秘密共享方案、加权斜坡秘密共享方案、SM2 数字签名以及本文方案相关协议等基础知识. 第 2 节给出加权门限 SM2 签名方案的模型与安全性定义. 第 3 节给出加权门限 SM2 签名方案的两个具体构造, 并对方案进行正确性分析. 第 4 节对两个方案的安全性进行分析. 第 5 节对方案进行效率和性能分析. 第 6 节介绍加权门限 SM2 签名方案的现实应用. 第 7 节总结本文工作.

1 基础知识

本节介绍中国剩余定理、基于中国剩余定理的秘密共享、SM2 数字签名算法、理想功能函数等相关基础知识的定义.

1.1 中国剩余定理[29]

中国剩余定理通常用于解决模运算问题. 它可以用来解决一组关于同余方程的问题, 即给定一组模数两两互素的同余方程, 可以通过中国剩余定理找到方程的唯一解, 从而简化计算. 具体描述如下.

设 $m_1,m_2,\ldots,m_k$ 是 k 个两两互素的正整数, 则对另一组整数列 $a_1,a_2,\ldots,a_k$, 有同余方程组:

$$\begin{cases} x\equiv a_1 \bmod m_1 \\ x\equiv a_2 \bmod m_2 \\ \quad\vdots \\ x\equiv a_k \bmod m_k \end{cases}.$$

M 为 $m_1,m_2,\ldots,m_k$ 这些所有正整数的和, 方程组在模 $M=\sum_{i=1}^{k}m_i$ 下必有唯一解为:

$$x\equiv\left(a_1M_1M_1^{-1}+a_2M_2M_2^{-1}+\ldots+a_kM_kM_k^{-1}\right) \bmod M,$$

其中, $M_i=M/m_i$ 即 M 中除去 m_i 剩余所有整数的和; 而 M_i^{-1} 为 M_i 在模 m_i 下的逆元, 即满足 $M_i^{-1}M_i=1 \bmod m_i$, 且 $i\in[1,k]$.

1.2 基于中国剩余定理的秘密共享方案[30]

基于中国剩余定理的秘密共享方案用于将秘密信息分割为多个份额, 当足够份额数重新组合时才能还原出原始的秘密, 是基于中国剩余定理的一种扩展应用. 具体如下.

(1) 系统初始化

设 p_0 是域 $\mathbb{F}$ 的阶, 在基于中国剩余定理的秘密共享中, 每个参与者 i 用整数 p_i 表示, 其中 $p_0,p_1,p_2,\ldots,p_n$ 要求互素, 共享秘密 $s\in\mathbb{F}_{p_0}$, 令 $M=p_1p_2\ldots p_n$. 在 $[0,M-1]$ 中选择随机整数 u, 计算 $S=s+up_0$, 则有 $S\in[0,M-1]$.

(2) 产生秘密份额

分发者计算 n 个秘密份额: $s_i = S \bmod p_i$, $i \in [1,n]$, 并把份额分发给每个参与者.

(3) 恢复秘密

对于一个参与者的授权集合 A, 通过唯一的整数 S 来重构秘密 s, 可构造如下同余方程组:

$$\begin{cases} S \equiv s_1 (\bmod p_1) \\ S \equiv s_2 (\bmod p_2) \\ \quad\vdots \\ S \equiv s_i (\bmod p_i) \end{cases}.$$

根据中国剩余定理, 可知该方程组在 $[0, M-1]$ 内有唯一解 S, 再根据式 $S = s + up_0$ 即可恢复出共享秘密 s.

1.3 高效 WRSS[28]

本节介绍高效加权斜坡秘密共享方案 (weighted ramp secret-sharing scheme, WRSS). WRSS 旨在解决传统门限秘密共享方案中所有参与者平等的局限性. WRSS 在秘密共享方案中引入权重, 以反映参与者的不同权力或信任级别. 这种方案的基本思想是将秘密信息分成多个部分, 并且每个部分有一个相关联的权重. 只有当收集到的部分的权重总和达到预设的权重门限时, 才能恢复原始的秘密. 具体结构如下.

设 $\mathbb{F}_{p_0}$ 是一个有限域, 其中 $p_0 \approx 2^{\lambda}$ 为它的阶. 假设要秘密共享的秘密为 $s \in \mathbb{F}_{p_0}$, 此方案基于中国剩余定理, 它是非线性的, 这些秘密份额 s_i, 满足 $0 \leqslant s_i < p_i - 1$, 其中 p_i 是对所有参与者的表示. 下面介绍基于中国剩余定理的 n 个参与方的 WRSS 的构造.

(1) 访问结构. 设在 n 个参与方的 WRSS 中, 每个参与者都有与之对应的权重 w_i. 存在重构门限 T 和秘密门限 t, 如果一组参与者的总体权重 $\geqslant T$, 将获得授权; 如果总体权重 $\leqslant t$, 则表示为未经授权. 在 WRSS 中, 一组总体权重属于 (t,T) 的各参与者可以了解到关于该秘密的部分信息. 其中, 重构门限 T 和秘密门限 t 定义如下.

重构门限 T. 如果 $\sum_{i\in A} w_i \geqslant T$, 则设置集合 $A \in \mathcal{A}$ 是授权的.

秘密门限 t. 如果 $\sum_{i\in B} w_i \leqslant t$, 则设置集合 $\bar{A} \in \bar{\mathcal{A}}$ 是未经授权的.

(2) 参数. 该方案参数为一组整数 $p_1, p_2, \ldots, p_n$ 以及整数 L, U_L 表示 $[0, L-1]$ 上的均匀分布. 要求所有的 p_i(包括 p_0) 都要互素. 这些参数定义了以下两个乘积, 其中, $P_{\max} < P_{\min}$:

$$P_{\max} = \max_{\bar{A}\in\bar{\mathcal{A}}} \left(\prod_{i\in\bar{A}} p_i \right), P_{\min} = \min_{A\in\mathcal{A}} \left(\prod_{i\in A} p_i \right).$$

(3) 共享秘密. 共享秘密 s, 需要选择一个随机整数构成:

$$S = s + p_0 \cdot U_L,$$

其中, U_L 表示 $[0, L-1]$ 上的均匀分布, 第 i 方的秘密份额应为:

$$s_i = S \bmod p_i.$$

(4) 重构秘密. 对于授权的集合 $A \in \mathcal{A}$, 使用中国剩余定理, 可以找到一组拉格朗日系数 $\{\lambda_i\}_{i\in A}$ 使得 $S = \sum_{i\in A} \lambda_i \cdot s_i$. 之后可重构秘密 s 为:

$$s = S \bmod p_0.$$

1.4 SM2 数字签名算法

SM2 数字签名算法是我国自主研发的椭圆曲线公钥密码算法, 于 2016 年成为国家密码标准. SM2 椭圆曲线数字签名算法的系统参数包括: 有限域 F_q; 椭圆曲线 $E(F_q)$ 方程; 有 $a, b \in F_q$; $G = (x_G, y_G)(G \neq O)$ 为 $E(F_q)$ 上的基点, 其中 x_G 和 y_G 为 F_q 中的元素; G 的阶 n. 用户 A 的杂凑值为 $Z_A = H_{256}(ENTL_A \| ID_A \| a \| b \| x_G \| y_G \| x_A \| y_A)$, 其中 H_{256} 为 SM3 密码杂凑算法, ID_A 为用户 A 位长为 $entlen_A$ 的可辨别标识, $ENTL_A$ 为 $entlen_A$ 转换的 $2B$ 数据.

SM2 数字签名算法包含 3 部分, 分别为密钥生成、签名算法和验证算法. 算法具体描述如下.

● 密钥生成

(1) 随机生成的秘密数 d_A 作为用户 A 的私钥, 且满足 $d_A \in [1, q-1]$.

(2) 计算 $P_A = [d_A]G = (x_A, y_A)$ 得到公钥并公开.

● 签名生成

(1) 对待签名的消息 M 进行签名时, 签名者先将消息 M 与自己的信息杂凑值连接得到 $\bar{M} = Z_A \| M$, 并计算 $e = H_v\left(\bar{M}\right)$.

(2) 签名者利用随机数生成器选取随机数 k, 满足 $k \in [1, n-1]$, 计算 $(x_1, y_1) = kG$.

(3) 计算 $r = (e + r_1) \bmod n$, 若 $r = 0$ 或 $r + k = n$, 则需要重新选择 k.

(4) 计算 $s = \left((1 + d_A)^{-1} \cdot (k - r \cdot d_A)\right) \bmod n$, 若 $s = 0$, 则需要重新选择 k; $s \neq 0$, 则 r, s 即为消息 M 的签名, 记为 (r, s).

● 签名验证

(1) 验证者收到消息 M' 及其数字签名 (r', s') 后, 首先检验 $r' \in [1, n-1]$ 和 $s' \in [1, n-1]$ 是否成立, 并得到 $\bar{M}' = Z_A \| M'$; 然后计算 $e' = H_v\left(\bar{M}'\right)$.

(2) 计算 $t = (r' + s') \bmod n$, 如果 $t = 0$, 即为验证失败; 否则, 计算 $(x_1', y_1') = [s']G + [t]P_A$.

(3) 计算 $R = (e' + x_1') \bmod n$, 判断等式 $R = r'$ 是否成立, 若等式成立则签名验证通过; 否则验证失败.

1.5 理想功能函数

本文借鉴文献 [28] 中的理想功能函数对方案进行构造. 分别为生成随机值的份额 F_{Random}、降阶协议 F_{deg}、乘法协议 F_{Mult} 和打开秘密份额 F_{Open}. 其中, $\mathbf{W} = (w_1, \ldots, w_n)$ 为 n 个参与者的总权重, 每个参与者 i 用整数 p_i 表示, 有集合 $\mathbf{P} = (p_0, p_1, \ldots, p_n)$. 设 (T, t) 分别是重构门限和秘密门限. 用 $\{[s]_i\}_{i\in[n]} \leftarrow Share(\mathbf{P}, T, t, s)$ 来表示某个秘密 s 的 WRSS 结果. 下面具体介绍这 4 个理想功能函数[28].

1.5.1 随机值份额 F_{Random}

各方生成随机值的秘密份额. 具体来说, 各方将自己随机生成的值进行秘密分割后发送给其他人, 所有人接收份额后进行本地加法计算得到随机值, 并将各方得到的随机值作为随机值份额, 即为此函数生成的结果. 具体描述如下.

(1) 对于所有的参与者 $i \in [n]$, 第 i 方选取随机值 $r_i \in \mathbb{F}$. 秘密分割 $r_i : \left\{[r_i]_j\right\}_{j\in[n]} \leftarrow Share(\mathbf{P}, T, t, r_i)$, 并将份额发送给各个参与方.

(2) 所有参与者接收到分割份额后, 本地计算 $[r_i] = ([r_1]_i + [r_2]_i + \ldots + [r_n]_i) \bmod p_i$ 作为随机域元素 $r = r_1 + \ldots + r_n \in \mathbb{F}$ 的份额.

在下文介绍中将用 $F_{\text{Random}}\left(r = \sum_{i\in[n]} r_i\right)$ 表示各个参与方生成 r 的随机份额 r_i.

1.5.2 降阶协议 F_{deg}

在本文方案设计中, 通过乘法协议对两个多项式进行乘法运算, 此时运算结果多项式的阶增高. 因此, 需要利用降阶协议对计算结果进行降阶. 具体描述如下.

(1) 输入. 各方选取 x 的秘密份额 $[x]$. 各方还持有随机值 r 的两个秘密份额 $\left\{[r]_i^0\right\}_{i\in[n]}$ 和 $\left\{[r]_i^1\right\}_{i\in[n]}$. 其中 $[r]^0$ 和 $[r]^1$ 都是使用 F_{Random} 函数生成.

(2) 参与者 i 本地计算并广播 $\left([x]_i + [r]_i^1\right) \bmod p_i$ 作为 $x + r$ 的秘密份额.

(3) 所有参与者接收到秘密份额后, 本地重构 $x + r \in \mathbb{F}$, 并从秘密份额 $\left\{[r]_i^0\right\}_{i\in[n]}$ 中减去 $(x + r) \bmod p_i$.

1.5.3 乘法协议 F_{Mult}

在方案设计中, 需要对两个多项式进行乘法运算, 这里对乘法协议进行介绍.

参与方本地计算 $([x]_i \cdot [y]_i) \bmod p_i$ 作为 $x \cdot y$ 的秘密份额. 然后使用降阶协议 F_{deg}, 得到 $[z]_i$ 作为 $z = x \cdot y$ 的新份额.

1.5.4 打开秘密份额 F_{Open}

在构造方案时, 各方需要打开输出值, 下面介绍打开秘密份额协议, 生成方式与 F_{Random} 类似.

(1) 输入. 各方持有输出线路的秘密份额 $[out]$. 各方还持有 [0] 的一个秘密份额.

(2) 参与者 i 本地计算并广播 $([0]_i+[out]_i) \bmod p_i$ 作为 $0+out$ 的秘密份额.

(3) 各方接收到份额后, 本地重构 $0+out$ 作为输出的值.

2 方案模型与安全性定义

2.1 加权门限 SM2 签名语法

本文加权门限 SM2 签名方案由 3 个多项式时间算法构成, 定义如下.

(1) $\mathcal{F}_{\text{Gen}}(1^\lambda,T,t)$: 以安全参数 1^λ, 重构门限 T 和隐私门限 t 作为输入. 输出部分密钥份额 $[sk]_i$ 和公钥 vk, 发送给各参与方.

(2) $\mathcal{F}_{\text{Sign}}([sk]_i,vk,M)$: 以密钥份额 $[sk]_i$, 公钥 vk, 明文消息 M 作为输入, 输出签名 $\sigma=(r,s)$.

(3) $\mathcal{F}_{\text{Verify}}(r,s,M,vk)$: 输入签名 (r,s), 签名消息 M, 公钥 vk. 输出 1 或 0, 其中输出 1 表示验证通过, 即签名有效; 反之, 输出 0 即签名无效.

加权门限 SM2 签名的正确性要求对任意的 $([sk]_i,vk)\leftarrow\mathcal{F}_{\text{Gen}}(1^\lambda,T,t)$ 和 $\sigma\leftarrow\mathcal{F}_{\text{Sign}}([sk]_i,vk,M)$, 当且仅当签名者对消息进行了有效签名, 有 $\mathcal{F}_{\text{Verify}}(r,s,M,vk)=1$.

2.2 敌手模型

敌手模型 (adversary model) 是一种理论上的抽象, 用来描述在特定情境下, 系统或协议可能受到的攻击和威胁. 敌手模型可以分为以下几类.

(1) 被动敌手 (passive adversary): 被动敌手可以监听和观察通信过程, 拦截、截获通信数据, 但不能修改或伪造消息. 被动敌手的目标通常是获取信息而不是破坏系统.

(2) 主动敌手 (active adversary): 主动敌手具有比被动敌手更强大的能力, 可截获、修改、删除、伪造消息, 甚至主动发起攻击. 主动敌手的目标通常是破坏系统的安全性或欺骗系统中的用户.

(3) 自适应敌手 (adaptive adversary): 自适应敌手是一种更加智能的敌手模型, 其可根据已有的信息和观察结果来调整攻击策略. 能够在攻击过程中不断学习和适应, 提高攻击成功的可能性.

(4) 静态敌手 (static adversary): 静态敌手其行为在整个攻击过程中保持不变或不随环境或攻击进展而改变. 静态敌手通常采用固定的攻击策略, 无论遇到何种情况都会使用相同的攻击方法.

适当的敌手模型对于设计安全的系统和协议至关重要. 本文针对静态敌手的攻击, 对方案进行安全性分析.

2.3 安全性定义

本节定义加权门限 SM2 签名方案的安全性, 包括数字签名的不可伪造性以及健壮性, 具体定义如下.

定义 1 (不可伪造性). 给定方案系统参数, 在敌手 $\mathcal{A}$ 最多可以腐化 t 个成员的情况下, 可以拥有交互视图, 并允许进行 q 次适应性选择消息签名问询, 而敌手 $\mathcal{A}$ 最终伪造新消息 M 的有效签名的概率是可忽略的.

定义 2 (健壮性). 在敌手 $\mathcal{A}$ 最多可以腐化 t 个参与者的情况下, 方案依然可以成功运行.

3 加权门限 SM2 签名方案构造

本节提出两种加权门限 SM2 签名方案的构造方法. 根据参与者权重之和不同, 获得 WRSS 相应的访问权限, 对 sk 进行密钥分割, 并发送给各参与方. 该过程仅需满足权重之和大于等于重构门限值的这部分签名者, 即可实现密钥分割及重构, 不需要所有签名者同时在线, 增强了 SM2 签名的健壮性. 本文方案设置预签名阶段, 参与者可以离线进行本地操作, 提高了运算效率, 且预签名阶段只进行两轮, 有效降低通信开销. 最后, 参与者通过非交互的签名阶段广播最终签名.

在加权门限 SM2 签名方案中, 其系统参数包括: 有限域 F_q; 椭圆曲线 $E(F_q)$ 方程; 有两个元素 $a,b\in F_q$; $G=(x_G,y_G)(G\neq O)$ 为 $E(F_q)$ 上的基点, x_G 和 y_G 为 F_q 中的元素; G 的阶 p. 假设有 n 个参与者, 每个参与者 i 用整数 p_i 表示, 即有集合 $\mathbf{P}=\{p_1,p_2,\ldots,p_n\}$; $S\in[n]$ 为参与加权门限 SM2 签名的参与者的子集, W 为 S 集合中参与者的总权重.

3.1 加权门限 SM2 签名 (方案 1)

本节提出第 1 种加权门限 SM2 签名方案, 在密钥生成时, 首先选取部分密钥 sk, 根据定义计算得到完整公私钥对. 将部分密钥 sk 进行 WRSS 秘密共享, 参与者们得到部分密钥和完整公钥. 在签名时, 只需要部分密钥就可以得到部分签名 s. 具体运算过程如下.

3.1.1 密钥生成 $\mathcal{F}_{\text{Gen}}(1^{\lambda},T,t)$

生成签名的公私钥. 这里对 SM2 签名算法私钥进行调整为 $u=sk-1$, 获得相应的签名公钥, 方便后续预签名阶段中对 SM2 密钥求逆的这一操作. 密钥生成具体如下.

$\mathcal{F}_{\text{Gen}}$ 以安全参数 1^{λ}, 以及 WRSS 的重构阈值 T 和隐私阈值 t 作为输入. 将执行以下操作.

(1) 随机选择一个部分签名密钥 $sk\leftarrow\mathbb{F}_q$. 令签名密钥 $u=sk-1$, 将验证公钥设置为 $vk=u\times G=(sk-1)\times G$.

(2) 根据 WRSS 的访问结构, 生成部分密钥 sk 的份额: $\{[sk]_i\}\leftarrow Share(\mathbf{P},T,t,sk)$. 然后发送 $(vk,\{[sk]_i\})$ 给各个参与方 i.

3.1.2 签名生成 $\mathcal{F}_{\text{Sign}}([sk]_i,vk,M)$

秘密输入 $[sk]_i$, 公开输入对应的公钥 vk 和消息 M. 采用以下协议进行签名: $(R_{\text{Random}},F_{\text{Mult}},F_{\text{Open}})$.

● 预签名阶段

利用本文介绍的函数, 得到通过预签名获得的签名相关值 $(r,[s^0]_i,[s^1]_i)$, 直接让每个参与方利用乘法函数得到本地份额乘法的结果, 从而使预签名阶段只进行两轮, 有效降低通信开销. 具体过程如下.

(1) 每个参与方 i 利用随机份额生成函数 F_{Random} 产生 γ 的份额: $\{[\gamma]_i\}_{i\in S}\leftarrow F_{\text{Random}}\left(\gamma=\sum_{i\in S}\gamma_i\right)$ 和 k 的份额 $\{[k]_i\}_{i\in S}\leftarrow F_{\text{Random}}\left(k=\sum_{i\in S}k_i\right)$. 每个参与方得到 k_i, 计算并广播 $k_i\times G$.

(2) 各方利用乘法函数计算 $\{[\delta]_i\}_{i\in S}=F_{\text{Mult}}\left(\{[\gamma]_i\}_{i\in S},\{[k]_i\}_{i\in S}\right)$ 和 $\{[\theta]_i\}_{i\in S}=F_{\text{Mult}}\left(\{[\gamma]_i\}_{i\in S},\{[sk]_i\}_{i\in S}\right)$, 方便计算签名 $s=sk^{-1}\cdot k+sk^{-1}\cdot r-r$.

(3) 各方利用打开函数计算 $\theta=F_{\text{Open}}(\{[\theta]_i\}_{i\in S})$. 利用通过广播本地保存的 $k_i\times G$ 计算出 $R=k\times G=\sum_{i\in S}k_i\times G$, 并设置曲线点 $R=(r_x,r_y)$.

(4) 各方计算 $e=H_v(M)$, 得到 $r=(e+r_x)\bmod p$.

(5) 各方计算 $[s^0]_i=\theta^{-1}\cdot\delta$ 和 $[s^1]_i=r\cdot\theta^{-1}\cdot[\gamma]_i$. 这里 $[s^0]_i$ 是 $sk^{-1}\cdot k$ 的一个份额, $[s^1]_i$ 是 $sk^{-1}\cdot r$ 的一个份额.

(6) 每个参与方 i 都保存这些值 $(r,[s^0]_i,[s^1]_i)$.

● 签名阶段

参与者将其保存的值进行加法计算得到签名份额并广播, 之后通过打开函数得到消息 M 的签名. 每个参与者广播最终签名, 这是一个非交互的签名阶段. 具体如下.

(1) 每个参与方 i 利用本地保存的值计算并广播 $[s]_i=[s^0]_i+[s^1]_i-r$.

(2) 各方计算 $s=F_{\text{Open}}(\{[s]_i\}_{i\in S})$, 得到 M 的签名是 $\sigma=(r,s)$.

3.1.3 签名验证 $\mathcal{F}_{\text{Verify}}(r,s,M,vk)$

验证者利用公钥对生成的签名进行验证, 计算相关值后代入验证等式, 判断等式是否成立即可验证签名是否有效. 此验证过程与 SM2 数字签名算法验证方法一致, 具体如下.

(1) 验证者收到消息 M' 及其数字签名 (r',s') 后, 首先检验 $r'\in[1,p-1]$ 和 $s'\in[1,p-1]$ 是否成立, 若不成立则验证不通过, 否则进行下一步.

(2) 计算 $e'=H_v(M')$ 和 $t=(r'+s')\bmod p$, 如果 $t=0$, 即为验证失败; 否则, 计算 $\left(r'_x,r'_y\right)=[s']G+[t]vk$, 其中 $vk=u\times G=(sk-1)\times G$.

(3) 计算 $R=(e'+r'_x)\bmod p$, 判断等式 $R=r'$ 是否成立, 若等式成立则签名验证通过; 否则验证失败.

3.1.4 正确性分析

对收到的消息 M' 及其数字签名 (r',s') 进行正确性分析. 由已知的 $t=(r'+s')\bmod p$, 可得:

$$\begin{cases}\left(r'_x,r'_y\right)=[s']G+[t]vk=[s']G+[tu]G=[s']G+[(r'+s')u]G=[(1+u)s'+ur']G\\ \quad=\left\{(1+sk-1)\left[sk^{-1}\cdot(k+r')-r'\right]+(sk-1)r'\right\}G=[(k+r')-sk\cdot r'+(sk-1)r']G=k\times G\\ (r_x,r_y)=k\times G\end{cases}\cdot$$

因此可以通过检验 $R=r'$ 是否成立, 来判断签名验证是否通过.

3.2 加权门限 SM2 签名 (方案 2)

本节给出第 2 种加权门限 SM2 签名具体构造, 相较于方案 1, 此方案在最开始的密钥生成阶段求得 sk 的逆元, 计算得到签名公钥, 从而在预签名阶段只需要调用一次乘法函数即可得到 θ, 并且计算部分签名 $[s^0]_i$ 和 $[s^1]_i$ 不需要再计算 θ 的逆, 这种方法更加高效, 通信开销减少. 方案具体构造如下.

3.2.1 密钥生成 $\mathcal{F}_{\text{Gen}}(1^\lambda,T,t)$

生成签名的公私钥. 这里对 SM2 签名算法私钥进行调整为 $v=sk^{-1}-1$, 获得相应的签名公钥, 方便后续预签名阶段中对 SM2 密钥求逆的这一操作. 密钥生成具体如下.

$\mathcal{F}_{\text{Gen}}$ 以安全参数 1^λ 以及 WRSS 的重构阈值 T 和隐私阈值 t 作为输入. 然后, 它将执行以下操作.

(1) 随机选择一个部分密钥 $sk\leftarrow\mathbb{F}_q$. 然后计算 sk^{-1}, 令签名私钥 $v=sk^{-1}-1$, 将对应的验证公钥设置为 $vk=v\times G=(sk^{-1}-1)\times G$.

(2) 根据 WRSS 的访问结构, 生成部分密钥 sk 的份额: $\{[sk]_i\}\leftarrow Share(\mathbf{P},T,t,sk)$. 然后发送 $(vk,\{[sk]_i\})$ 给各个参与方 i.

3.2.2 签名生成 $\mathcal{F}_{\text{Sign}}([sk]_i,vk,M)$

秘密输入 $[sk]_i$, 公开输入对应的公钥 vk 和消息 M. 采用以下协议进行签名: $(R_{\text{Random}},F_{\text{Mult}},F_{\text{Open}})$.

● 预签名阶段

利用本文介绍的函数, 得到通过预签名获得的签名相关值. 直接让每个参与方利用乘法函数得到本地份额乘法的结果, 从而使预签名阶段只进行两轮, 有效降低通信开销. 具体过程如下.

(1) 每个参与方 i 利用随机份额生成函数 F_{Random} 产生 γ 的份额: $\{[\gamma]_i\}_{i\in S}\leftarrow F_{\text{Random}}\left(\gamma=\sum_{i\in S}\gamma_i\right)$ 和 k 的份额: $\{[k]_i\}_{i\in S}\leftarrow F_{\text{Random}}\left(k=\sum_{i\in S}k_i\right)$. 每个参与方得到 k_i, 计算并广播 $k_i\times G$.

(2) 各方利用乘法函数计算出 $\{[\theta]_i\}_{i\in S}=F_{\text{Mult}}\left(\{[\gamma]_i\}_{i\in S},\{[sk]_i\}_{i\in S}\right)$.

(3) 各方利用打开函数计算 $\gamma=F_{\text{Open}}\left(\{[\gamma]_i\}_{i\in S}\right)$. 利用通过广播本地保存的 $k_i\times G$ 计算出 $R=k\times G=\sum_{i\in S}k_i\times G$, 并设置曲线点 $R=(r_x,r_y)$.

(4) 计算 $e=H_v(M)$, 得到 $r=(e+r_x)\bmod p$.

(5) 各方计算 $[s^0]_i=\gamma^{-1}\cdot\theta\cdot k$ 和 $[s^1]_i=\gamma^{-1}\cdot\theta\cdot r$. 这里的 $[s^0]_i$ 是 $sk\cdot k$ 的一个份额, $[s^1]_i$ 是 $sk\cdot r$ 的一个份额.

(6) 每个参与方 i 都保存这些值 $(r,[s^0]_i,[s^1]_i)$.

● 签名阶段

每个参与者将保存的值通过加法计算, 得到部分签名份额并广播, 然后通过打开函数得到消息 M 的签名. 具体如下.

(1) 每个参与方 i 本地计算 $[s]_i=[s^0]_i+[s^1]_i-r$.

(2) 各方计算 $s=F_{\text{Open}}(\{[s]_i\}_{i\in S})$, 消息 M 的签名是 $\sigma=(r,s)$.

3.2.3 签名验证 $\mathcal{F}_{\text{Verify}}(r,s,M,vk)$

此签名方案的验证过程与第 3.1.3 节中的方案 1 验证过程类似. 但值得注意的是, 两者验证阶段的第 (2) 步有所不同, 本节方案 2 签名的密钥为 $v=sk^{-1}-1$, 故而在计算椭圆曲线点 $(r'_x,r'_y)=[s']G+[t]vk$ 时, 其中的签名公钥应为: $vk=v\times G=(sk^{-1}-1)\times G$.

3.2.4 正确性分析

对收到的消息 M' 及其数字签名 (r',s') 进行正确性分析. 由已知的 $t=(r'+s')\bmod p$, 可得:

$$\begin{cases}(r'_x, r'_y) = [s']G + [t]vk = [s']G + [tv]G = [s']G + [(r'+s')v]G = [(1+v)s' + vr']G \\ \quad = \left\{\left(1+sk^{-1}-1\right)[sk\cdot(k+r')-r'] + \left(sk^{-1}-1\right)r'\right\}G = \left[(k+r') - sk^{-1}\cdot r' + \left(sk^{-1}-1\right)r'\right]G = k\times G \\ (r_x, r_y) = k\times G\end{cases}.$$

因此可以通过检验 $R = r'$ 是否成立, 来判断签名验证是否通过.

4 方案安全性

根据文献 [31,32] 有以下定理.

定理 1. 如果 SM2 数字签名方案是存在性不可伪造的, 且加权门限 SM2 签名方案是可模拟的, 那么加权门限的 SM2 签名方案满足不可伪造性.

证明: 首先对本文签名方案 1 (方案 2) 进行模拟, 模拟过程如协议 1 所示. 本文基于半诚实模型假设, 输入公钥 $vk = sk - 1$(或 $vk = sk^{-1} - 1$), 消息 M, 消息签名 (r, s). 在预签名阶段, 参与者诚实地执行协议规定的操作, 不会向其他参与者或未经授权的实体透露其私密信息, 不篡改其私密信息或其他参与者的信息. 假设共有 n 个参与者, 敌手 $\mathcal{A}$ 可以控制 t 个参与者, 因此敌手控制份额为 $(sk'_1, sk'_2, \ldots, sk'_t)$, 容易证明通过私钥 sk_i 得到 sk'_i 的过程是安全的.

协议 1. 模拟协议.

输入: 公钥 vk, 其中 vk=sk−1 (vk=sk^{-1}−1), 消息 M, 消息签名 σ=(r, s), 控制份额 $(sk'_1, sk'_2, \ldots, sk'_t)$;

(1) 计算 r^*=sG+$(r$+$s)vk$.

(2) 半诚实参与者执行 F_{Random} 协议得到共享份额 $k_i\times G$, 敌手的份额模拟器是可以监听到的, 因此, 模拟器可以得到所有的份额, 且敌手的共享份额为 $k_1\times G, \ldots, k_t\times G$.

(3) 记 $r_i^* = k'_i G$ $(1 \leqslant i \leqslant t)$, 且满足任意 t+1 个 r_i^* 可以了解到秘密 r^* 的一些信息, T 个 r_i^* 可以恢复出 r^*. 模拟器掌握 t 个 $k_i\times G$, 其可以计算得到剩余的 $r_i^* = k'_i G$ $(t+1 \leqslant i \leqslant n)$, 将得到的 r_i^* $(t+1 \leqslant i \leqslant n)$ 广播给诚实参与者, 就可以恢复出 $r^* = (r_x, r_y)$, 而 $r = (e+r_x) \bmod p$.

(4) 可以计算 $[s^0]_i$ $(1 \leqslant i \leqslant t)$ 和 $[s^1]_i$ $(1 \leqslant i \leqslant t)$, 在 $t+1 \leqslant i \leqslant T$ 范围内随机选取 s'_i, 由 s'_i $(1 \leqslant i \leqslant t)$ 和 $s'_0 = s$ 可以唯一确定一个共享多项式, 故而可以确定其余 i 满足 $(T+1 \leqslant i \leqslant n)$ 的 s'_i.

(5) 向诚实参与者广播得到的 s'_i $(T+1 \leqslant i \leqslant n)$.

从上面的模拟协议可以看出, 模拟器与加权门限 SM2 签名方案中的变量是一致的, 以下具体分析其具有相同的概率分布.

(1) 因为 $k_1\times G, \ldots, k_t\times G$ 是由 F_{Random} 协议生成, 故 $r_i^* = k'_i G$ $(1 \leqslant i \leqslant t)$ 都是均匀随机分布. 又因为其余 r_i^* $(t+1 \leqslant i \leqslant n)$ 是由 r_i^* $(1 \leqslant i \leqslant t)$ 和 r^* 确定的, 因此, r_i^* $(t+1 \leqslant i \leqslant n)$ 和 r_i^* $(1 \leqslant i \leqslant t)$ 具有相同的概率分布.

(2) 同样的, 因为 s'_i $(T+1 \leqslant i \leqslant n)$ 是由 s'_i $(1 \leqslant i \leqslant T)$ 和 $s'_0 = s$ 确定, 所以 s'_i $(T+1 \leqslant i \leqslant n)$ 与 s'_i $(1 \leqslant i \leqslant T)$ 的概率分布相同.

因此, 可以证明加权门限 SM2 签名方案具有不可伪造性. 证毕.

定理 2. 共有 n 个参与者参与加权门限 SM2 签名, 对于权重总和为 t 的参与者被腐化, 如果 $n \geqslant T > t$, 那么加权门限 SM2 签名方案是健壮的.

证明: 假设敌手 $\mathcal{A}$ 控制权重总和为 t 的参与者, 在密钥分发阶段, 根据 WRSS 的访问结构, 参与者的总体权重 $\geqslant T$, 获得授权; 如果一组参与者的总体权重 $\leqslant t$, 则他们是未经授权的. 所以敌手不能了解到关于秘密的有效信息, 并且不能重构秘密. 因此只需要保证 $n \geqslant T > t$, 即可完成秘钥重构并完成签名过程. 证毕.

5 性能分析

5.1 计算开销分析

本节将本文方案与现有的分布式 SM2 签名方案[6]、门限 SM2 签名方案[14]的计算开销进行对比, 主要从密钥

生成阶段、签名生成阶段以及验证阶段进行分析.

由于本文设有预签名阶段, 考虑到各方案总体计算开销对比, 分析了本文方案预签名阶段的计算开销. 对比结果如表 1 所示. 假设有 n 个参与者, 其中 dc 表示点乘, dj 表示点加, mj 表示模加, mn 表示模逆, mc 表示模乘运算. 需要注意的是, 我们对比的是文献 [14] 在密钥生成阶段不存在可信中心的情况下各阶段的计算量. 在密钥生成阶段, 文献 [6] 计算 Q 时需要执行 n 次点乘运算, 计算公钥时执行了 n 次点加运算. 文献 [14] 执行 PM-SS 得到公钥以及执行 Inv-ss 得到私钥份额时需要 n 次点乘、1 次模逆和 $3n$ 次模乘运算. 而本文方案需要在计算公钥和密钥份额时只需要 n 次点乘运算; 在签名生成阶段, 文献 [6] 计算中间变量以及签名时共需要执行 n 次点乘、$(2n+3)$ 次点加、1 次模逆和 $2n$ 次模乘运算. 文献 [14] 在计算 kG 和签名份额时需要执行 n 次点乘、$n+1$ 次点加和 $2n$ 次模乘运算. 本文仅需要 n 次点加运算即可完成. 而在本文方案的签名阶段需要 n 次点乘、1 次点加、1 次模逆和 $4n$ 次模乘运算; 它们的签名验证主要依赖于 SM2 数字签名的验证过程. 因此, 易知在验证阶段它们的计算量是相当的. 由上表可以看出, 文献 [6] 总体的计算量最大, 文献 [14] 的计算量较小, 本文方案的计算量最小.

表 1 本文方案与文献 [6] 和文献 [14] 计算量比较

方案	密钥生成	预签名阶段	签名生成	验证阶段
文献[6]	$(n)dc+(n)dj$	—	$(n)dc+(2n+3)mj+mn+(2n)mc$	$dc+dj$
文献[14]	$(n)dc+mn+(3n)mc$	—	$(n)dc+(n+1)mj+(2n)mc$	$dc+dj$
本文方案	$(n)dc$	$(n)dc+mj+mn+(4n)mc$	$(n)mj$	$dc+dj$

5.2 功能性对比

本节分别从参与方是否加权、健壮性、预签名和半诚实模型这 4 个方面与已有方案进行功能性对比. 其中, × 表示方案不具备该性质或功能; √则表示具有该性质或功能. 功能性对比如表 2 所示.

表 2 本文方案与文献 [6] 和文献 [14] 功能性对比

方案	加权(权重)	健壮性	预签名	半诚实模型
文献[6]	×	×	×	—
文献[14]	×	√	×	√
本文方案	√	√	√	√

相较文献 [6], 本文引入了加权的概念, 在现实应用上更加具备灵活性. 相较于文献 [14] 的 SM2 分布式签名, 本方案不需要所有参与者同时在线以此完成签名密钥的分发. 此外, 本文方案有效抵抗文献 [14] 中部分密钥丢失的攻击, 增强了 SM2 签名的健壮性. 本文保证半诚实模型下安全, 同时方案增加预签名阶段, 参与者可以离线进行本地操作, 提高了运算效率, 有效降低了通信开销.

6 应 用

在传统门限秘密共享方案中, 所有参与者被视为具有相同地位. 然而, 在实际场景中, 参与者的权力可能不均. 因此, 人们提出了加权门限秘密共享方案. 其主要思想是, 只有当参与者的权重总和达到预设的权重门限时, 方可还原秘密; 反之, 则无法还原秘密. 本文结合加权门限 SM2 签名与区块链技术, 针对现实应用中参与者权利不同的问题, 设计出一个基于加权门限 SM2 签名的区块链投票系统. 系统模型如图 1 所示.

系统主要包含: 用户、区块链模块和加权门限 SM2 签名模块. n 个用户提出投票请求, 只有权重之和$\geqslant T$ 的用户才能利用加权门限 SM2 签名模块生成签名, 权重之和在 (t,T) 之间可以了解到签名密钥相关信息, 但不能进行签名, 而权重之和$\leqslant t$ 的用户既不能了解到密钥相关信息也不能进行签名. 用户签名后利用加权门限 SM2 签名模块对签名进行验证, 验证通过后投票才会被认可并通过区块链模块记录在区块链上. 并且任何人都可以在区块链上查询投票结果, 确保透明和公正.

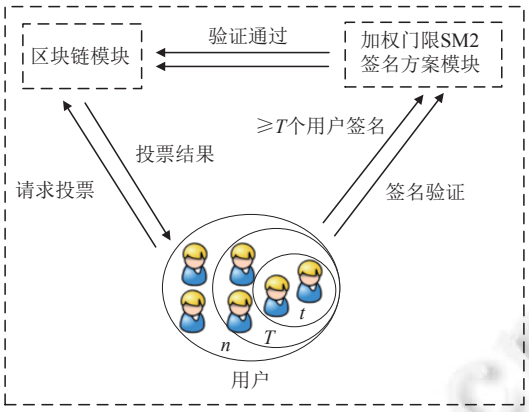

图 1　基于加权门限 SM2 签名的区块链投票系统

7 总　结

本文基于高效的 WRSS 提出了两种能够实现加权功能的门限 SM2 数字签名, 当且仅当所有参与者的权重之和大于或等于一个固定的阈值时, 该秘密才可以被重构. 在秘密分割时, 对 SM2 数字签名算法的签名私钥进行变形, 并对公钥做出调整, 使其在签名阶段恢复密钥进行签名得到简化. 本文进一步给出了方案的正确性和安全性分析, 并与门限 SM2 签名以及联合 SM2 签名的进行计算开销和功能性对比. 结果表明本文方案计算开销最小; 在功能上提供加权的性质, 使得方案更具健壮性. 并且参与者可以执行离线操作的预签名阶段, 提高了运算效率, 有效降低了通信开销. 在现实应用中通常需要综合考虑参与者分配权重, 本文探索基于加权门限 SM2 签名的区块链投票系统, 扩展了国密 SM2 签名的现实应用.

References:

[1] Xu QL, Li DX. Elliptic curve cryptosystems. Journal of Computer Research and Development, 1999, 36(11): 1281–1288 (in Chinese with English abstract).

[2] Wang CH, Zhang ZF. Overview on public key cryptographic algorithm SM2 based on elliptic curves. Journal of Information Security Research, 2016, 2(11): 972–982 (in Chinese with English abstract).

[3] State Cryptography Administration. SM2 elliptic curve cryptographic algorithm. 2010 (in Chinese). https://www.oscca.gov.cn/sca/xxgk/2010-12/17/content_1002386.shtml

[4] Lin JQ, Ma Y, Jing JW, Wang QX, Lei LG, Cai QW, Wang L. Signing and decrypting method and system applied to cloud computing and based on SM2 algorithm. CN: 104243456A, 2014-12-24 (in Chinese).

[5] He DB, Zhang JN, Feng Q, Wang J, Chen MW. Lightweight SM2 two-party collaborative digital signature generation method. CN: 110011803A, 2019-07-12 (in Chinese).

[6] He DB, Feng Q, Wang J, Lin C, Zhang YD, Zhang JN. Method for jointly generating SM2 digital signature by multiple parties. CN: 109547199A, 2019-03-29 (in Chinese).

[7] He DB, Zhang YD, Lin C, Feng Q, Wang J, Zhang JN. A multi-party collaborative method for generating SM2 digital signatures. CN: 109474422A, 2019-03-15 (in Chinese).

[8] Zhang YD, He DB, Zhang MW, Choo KKR. A provable-secure and practical two-party distributed signing protocol for SM2 signature algorithm. Frontiers of Computer Science, 2020, 14(3): 143803. [doi: 10.1007/s11704-018-8106-9]

[9] Hou HX, Yang B, Zhang LN, Zhang MR. Secure two-party SM2 signature algorithm. Acta Electronica Sinica, 2020, 48(1): 1–8 (in Chinese with English abstract). [doi: 10.3969/j.issn.0372-2112.2020.01.001]

[10] Feng Q, He DB, Luo M, Li L. Efficient two-party SM2 signing protocol for mobile Internet. Journal of Computer Research and Development, 2020, 57(10): 2136–2146 (in Chinese with English abstract). [doi: 10.7544/issn1000-1239.2020.20200401]

[11] Su XY, Tian HB. A two-party SM2 signing Protocol and its application. Chinese Journal of Computers, 2020, 43(4): 701–710 (in Chinese with English abstract). [doi: 10.11897/SP.J.1016.2020.00701]

[12] Tang ZY, Wang ZW. A threshold SM2 Signature scheme. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 2022, 42(4): 85–95 (in Chinese with English abstract). [doi: 10.14132/j.cnki.1673-5439.2022.04.012]

[13] Han G, Bai X, Geng SL, Qin BD. Efficient two-party SM2 signing protocol based on secret sharing. Journal of Systems Architecture, 2022, 132: 102738. [doi: 10.1016/j.sysarc.2022.102738]

[14] Shang M, Ma Y, Lin JQ, Jing JW. A threshold scheme for SM2 elliptic curve cryptographic algorithm. Journal of Cryptologic Research, 2014, 1(2): 155–166 (in Chinese with English abstract). [doi: 10.13868/j.cnki.jcr.000015]

[15] Tu BB, Wang XF, Zhang LT. Two distributed applications of SM2 and SM9. Journal of Cryptologic Research, 2020, 7(6): 826–838 (in Chinese with English abstract). [doi: 10.13868/j.cnki.jcr.000409]

[16] Pang LJ, Fei QQ, Li HX, Xu QJ. Secret sharing technology and its applications. Journal on Communications, 2017, 38(3): 183 (in Chinese with English abstract).

[17] Miao FY, Wang L, Ji YY, Xiong Y. GOMSS: A simple group oriented (*t*, *m*, *n*) multi-secret sharing scheme. Chinese Journal of Electronics, 2017, 26(3): 557–563. [doi: 10.1049/cje.2016.08.014]

[18] Shamir A. How to share a secret. Communications of the ACM, 1979, 22(11): 612–613. [doi: 10.1145/359168.359176]

[19] Blakley GR. Safeguarding cryptographic keys. In: Proc. of the 1979 Int'l Workshop on Managing Requirements Knowledge. New York: IEEE, 1979. 313–313. [doi: 10.1109/MARK.1979.8817296]

[20] Asmuth C, Bloom J. A modular approach to key safeguarding. IEEE Trans. on Information Theory, 1983, 29(2): 208–210. [doi: 10.1109/TIT.1983.1056651]

[21] Ore O. The general Chinese remainder theorem. The American Mathematical Monthly, 1952, 59(6): 365–370. [doi: 10.1080/00029890.1952.11988142]

[22] Mignotte M. How to share a secret. In: Proc. of the 1983 Workshop on Cryptography. Burg Feuerstein: Springer, 1983. 371–375. [doi: 10.1007/3-540-39466-4_27]

[23] Morillo P, Padró C, Sáez G, Villar JL. Weighted threshold secret sharing schemes. Information Processing Letters, 1999, 70(5): 211–216. [doi: 10.1016/S0020-0190(99)00070-8]

[24] Beimel A, Weinreb E. Monotone circuits for monotone weighted threshold functions. Information Processing Letters, 2006, 97(1): 12–18. [doi: 10.1016/j.ipl.2005.09.008]

[25] Beimel A, Tassa T, Weinreb E. Characterizing ideal weighted threshold secret sharing. SIAM Journal on Discrete Mathematics, 2008, 22(1): 360–397. [doi: 10.1137/S0895480104445654]

[26] Iftene S, Boureanu I. Weighted threshold secret sharing based on the Chinese remainder theorem. Technical Report, Iasi: Faculty of Computer Science Iasi, Romania. https://infoscience.epfl.ch/server/api/core/bitstreams/0d0302bf-b0d4-40d9-9215-55291d208bfb/content

[27] Chaidos P, Kiayias A. Mithril: Stake-based threshold multisignatures. In: Proc. of the 2024 Int'l Conf. on Cryptology and Network Security. Cambridge: Springer. 2024. 239–263.

[28] Garg S, Jain A, Mukherjee P, Sinha R, Wang MY, Zhang YN. Cryptography with weights: MPC, encryption and signatures. In: Proc. of the 43rd Annual Int'l Cryptology Conf. on Advances in Cryptology. Santa Barbara: Springer, 2023. 295–327. [doi: 10.1007/978-3-031-38557-5_10]

[29] Chen ZW, Zhang LJ, Wang YM, Huang JW, Huang DR. A group signature scheme based on Chinese remainder theorem. Acta Electronica Sinica, 2004, 32(7): 1062–1065 (in Chinese with English abstract). [doi: 10.3321/j.issn:0372-2112.2004.07.002]

[30] Goldreich O, Ron D, Sudan M. Chinese remaindering with errors. IEEE Trans. on Information Theory, 2000, 46(4): 1330–1338. [doi: 10.1109/18.850672]

[31] Goldwasser S, Micali S, Rivest RL. A digital signature scheme secure against adaptive chosen-message attacks. SIAM Journal on Computing, 1988, 17(2): 281–308. [doi: 10.1137/0217017]

[32] Zhang ZF, Yang K, Zhang J, Cheng C. Security of the SM2 signature scheme against generalized key substitution attacks. In: Proc. of the 2nd Int'l Conf. on Security Standardisation Research. Tokyo: Springer. 2015. 140–153. [doi: 10.1007/978-3-319-27152-1_7]

附中文参考文献:

[1] 徐秋亮, 李大兴. 椭圆曲线密码体制. 计算机研究与发展, 1999, 36(11): 1281–1288.

[2] 汪朝晖, 张振峰. SM2 椭圆曲线公钥密码算法综述. 信息安全研究, 2016, 2(11): 972–982.

[3] 国家密码管理局. 国家密码管理局关于发布《SM2 椭圆曲线公钥密码算法》公告. 2010. https://www.oscca.gov.cn/sca/xxgk/2010-12/17/content_1002386.shtml

[4] 林璟锵, 马原, 荆继武, 王琼霄, 雷灵光, 蔡权伟, 王雷. 适用于云计算的基于 SM2 算法的签名及解密方法和系统. 中国:

104243456A, 2014-12-24.
[5] 何德彪, 张佳妮, 冯琦, 王婧, 陈泌文 . 一种轻量级 SM2 两方协同生成数字签名的方法. 中国: 110011803A, 2019-07-12.
[6] 何德彪, 冯琦, 王婧, 林超, 张语荻, 张佳妮 . 一种多方联合生成 SM2 数字签名的方法. 中国: 109547199A, 2019-03-29.
[7] 何德彪, 张语荻, 林超, 冯琦, 王婧, 张佳妮. 一种多方协同产生 SM2 数字签名的方法. 中国: 109474422A, 2019-03-15.
[9] 侯红霞, 杨波, 张丽娜, 张明瑞. 安全的两方协作 SM2 签名算法. 电子学报, 2020, 48(1): 1–8. [doi: 10.3969/j.issn.0372-2112.2020.01.001]
[10] 冯琦, 何德彪, 罗敏, 李莉. 移动互联网环境下轻量级 SM2 两方协同签名. 计算机研究与发展, 2020, 57(10): 2136–2146. [doi: 10.7544/issn1000-1239.2020.20200401]
[11] 苏吟雪, 田海博. 基于 SM2 的双方共同签名协议及其应用. 计算机学报, 2020, 43(4): 701–710. [doi: 10.11897/SP.J.1016.2020.00701]
[12] 唐张颖, 王志伟. 门限 SM2 签名方案. 南京邮电大学学报 (自然科学版), 2022, 42(4): 85–95. [doi: 10.14132/j.cnki.1673-5439.2022.04.012]
[14] 尚铭, 马原, 林璟锵, 荆继武. SM2 椭圆曲线门限密码算法. 密码学报, 2014, 1(2): 155–166. [doi: 10.13868/j.cnki.jcr.000015]
[15] 涂彬彬, 王现方, 张立廷. 两种分布式 SM2/9 算法应用. 密码学报, 2020, 7(6): 826–838. [doi: 10.13868/j.cnki.jcr.000409]
[16] 庞辽军, 裴庆祺, 李慧贤, 徐启建. 秘密共享技术及其应用. 通信学报, 2017, 38(3): 183.
[29] 陈泽文, 张龙军, 王育民, 黄继武, 黄达人. 一种基于中国剩余定理的群签名方案. 电子学报, 2004, 32(7): 1062–1065. [doi: 10.3321/j.issn:0372-2112.2004.07.002]

唐长虹(2000－), 女, 硕士生, 主要研究领域为国密算法分析与设计, 后门密码学.

冯琦(1994－), 女, 博士, 副研究员, 主要研究领域为应用密码学, 安全协议, 隐私计算.

赵艳琦(1992－), 男, 博士, 副教授, CCF 专业会员, 主要研究领域为公钥密码学, 区块链安全.

禹勇(1980－), 男, 博士, 教授, 博士生导师, 主要研究领域为密码学, 数据安全, 区块链安全.

杨晓艺(1993－), 女, 博士, 讲师, 主要研究领域为隐私计算, 安全多方计算.