

DOI:10.3969/j.issn.1671-0673. 2026. 04. 004

可穿刺的密文策略基于属性加密方案

杨冬梅^{1,2}, 胡学先¹, 魏江宏¹, 巴 阳¹, 陈 越¹

(1.信息工程大学,河南 郑州 450001; 2.河南信息统计职业学院,河南 郑州 450001)

摘要:针对传统前向安全的基于属性加密(ABE)方案在实际可用性与密钥更新效率方面存在的不足,提出一种可穿刺的密文策略基于属性加密(P-CP-ABE)方案。首先,基于可穿刺公钥加密体制思想,给出P-CP-ABE方案的形式化定义与安全模型;随后,基于布隆过滤器,构造一个具体的P-CP-ABE方案。在判定性 l -BDHE (l -Bilinear Diffie-Hellman Exponent)假设下,证明了所提方案在标准模型下满足选择性前向安全性。性能分析与仿真实验结果表明,该方案显著提升了密钥更新效率,实现了更具实用性的细粒度前向安全性,为数据加密与细粒度访问控制提供了高效、安全的新方法。

关键词:私钥泄露;前向安全;基于属性加密;可穿刺加密;布隆过滤器

中图分类号:TP309.7

文献标识码:A

文章编号:1671-0673(2026)04-0406-08

Puncturable Ciphertext-Policy Attribute-Based Encryption Scheme

YANG Dongmei^{1,2}, HU Xuexian¹, WEI Jianghong¹, BA Yang¹, CHEN Yue¹

(1.Information Engineering University, Zhengzhou 450001, China;

2.Henan Vocational College of Information and Statistics, Zhengzhou 450001, China)

Abstract: To address the shortcomings of traditional forward-secure attribute-based encryption (ABE) scheme in terms of practical availability and private key update efficiency, a puncturable ciphertext-policy attribute-based encryption (P-CP-ABE) scheme is proposed. Firstly, the syntax and security model of P-CP-ABE based on the idea of puncturable public-key encryption are presented. Then, a concrete P-CP-ABE scheme based on a Bloom filter is constructed. Under the decisional l -BDHE (l -Bilinear Diffie-Hellman Exponent) assumption, the proposed scheme is proven to achieve selective forward security in the standard model. The performance analysis and simulation results demonstrate that private key update efficiency is significantly improved and more practical fine-grained forward security is realized by using the scheme. An efficient and secure approach for data encryption and fine-grained access control is provided by using the proposed scheme.

Key words: private key leakage; forward security; attribute-based encryption; puncturable encryption; Bloom filter

云计算、物联网等领域的高速发展,持续推动数据共享与交互场景的深度拓展,数据隐私保护与访问安全已成为该领域发展中亟待解决的核心问题。传统公钥加密体制采用一对一密钥管控模式,加密方需针对每个接收方独立生成密文,难以

满足多用户、多场景下的细粒度访问控制需求。Sahai等^[1]提出的基于属性加密(Attribute-Based Encryption, ABE)体制,为破解该难题提供了有效技术路径。ABE技术依托访问结构实现数据加密,仅当用户属性满足密文对应的访问结构时才可正

收稿日期:2026-02-16;修回日期:2026-04-10

基金项目:国家自然科学基金(62572483,62472438)

作者简介:杨冬梅(1977—),女,副教授,博士生,主要研究方向为大数据安全。E-mail:339097597@qq.com

常解密,以此实现灵活的访问控制。此后,Goyal等^[2]将 ABE 体制分为密文策略 ABE (Ciphertext-Policy ABE, CP-ABE) 与密钥策略 ABE (Key-Policy ABE, KP-ABE) 两类,其中 CP-ABE 的用户私钥与自身属性相关、密文与访问结构相关,KP-ABE 则与之相反。围绕丰富访问结构、增强安全性能、提升运行效率等方面,密码学领域相继提出了诸多改进的 ABE 方案^[3-6]。传统 ABE 方案的安全性均建立在用户私钥绝对安全的前提假设之上,而现阶段网络攻击手段日趋多样化、移动电子设备大范围普及,导致私钥泄露事件频发。攻击者可利用泄露的私钥,解密该用户属性权限范围内的所有密文,大幅降低了 ABE 方案在高安全需求场景中的实际适用性。

为解决私钥泄露带来的安全问题,Anderson^[7]首次在公钥密码体制中引入前向安全思想。受该思想启发,研究者相继提出一系列前向安全的签名方案^[8-11]。Canetti 等^[12]结合分层的身份基加密^[13]提出前向安全构造方法,采用二叉树结构对用户密钥与时间周期进行统一管理。该方法将用户私钥的全生命周期划分为 T 个离散时间周期,并映射至一棵完全二叉树中,每个时间周期与树中唯一叶子节点相对应;用户私钥随时间周期动态更新,且及时删除当前周期的私钥,保证不同时间周期内私钥的唯一性,同时保持公钥固定不变,以此实现前向安全性。该方法被广泛应用于前向安全的属性基密码体制的设计中^[14]。基于该方法,文献^[15]和文献^[16]相继设计了前向安全的 CP-ABE 方案和前向安全的身份基签名方案;Oh 等^[17]则提出了全系统前向安全的身份基签名方案。

针对粗粒度前向安全的固有局限性,Green 等^[18]提出可穿刺公钥加密体制。此后,Günther 等^[19]提出具备完全前向安全性的 0 往返时间密钥交换方案,Cini 等^[20]设计了满足自适应选择密文攻击安全的可穿刺密钥封装管理方案。为提升私钥穿刺效率,Derler 等^[21]提出布隆过滤器加密方案,Li 等^[22]基于该方案构造可穿刺签名方案,并将其应用于区块链权益证明协议中。可穿刺加密方法在属性基加密领域亦得到广泛应用,如 Wei 等^[23-24]构造了前向安全的可穿刺身份基加密方案,继而提出可同时抵抗用户私钥与系统主密钥泄露的 ABE 方案^[25],并将其应用于外包存储系统,实现了细粒度数据共享与安全擦除防护。Deler 等^[26]基于允许/拒绝列表加密,设计了支持正负两种形式的可穿刺

加密方案。此外,在属性基加密的工程化应用层面,相关研究结合区块链技术开展了深入探索^[27-28]。

综上所述,为实现 CP-ABE 体制中高效、细粒度的前向安全性,本文设计一种可穿刺的密文策略基于属性加密 (Puncturable Ciphertext-Policy Attribute-Based Encryption, P-CP-ABE) 方案,通过私钥穿刺算法解决传统前向安全 CP-ABE 方案灵活性不足、私钥更新效率较低的问题。首先,给出 P-CP-ABE 方案的形式化与安全性定义;其次,借助布隆过滤器构造具体的 P-CP-ABE 方案,实现私钥的高效穿刺与更新,并基于判定性 l -BDHE (l -Bilinear Diffie-Hellman Exponent) 假设,在标准模型下完成方案的安全性证明;最后,通过性能分析和仿真实验进一步验证方案的实际运行性能。

1 基础知识

定义 1 布隆过滤器^[29]。一种空间效率很高的随机数据结构,它利用位数组简洁地表示一个集合,并能判断一个元素是否属于这个集合,即:令 $U = \{u_1, u_2, \dots, u_n\}$ 表示 n 个元素的集合,对任意一个元素 u ,若 $u \in U$,则以概率 1 返回 1;若 $u \notin U$,则以概率 $(1 - \delta)$ 返回 0,其中 δ 为假阳性概率。布隆过滤器的算法组成如下。

1) BFGen(λ, n) 为生成算法。给定安全参数 λ ,集合中元素个数为 n ,首先,随机选择整数 $\ell, k \in \mathbb{N}$, ℓ 为布隆过滤器位数组长度,且满足 $\delta = (1 - e^{-kn/\ell})^k$;然后选择 k 个哈希函数 $H_1, \dots, H_k$,令 $H = (H_j)_{j \in [k]}$,并初始化位数组 $T = 0^\ell$,最后输出 (H, T) 。

2) BFUpdate(H, T, u) 为更新算法。以 $H = (H_j)_{j \in [k]}$, $T \in \{0, 1\}^\ell$ 和元素 u 为输入。首先令 $T' \leftarrow T$,然后对任意 $j \in [k]$,令 $T'[H_j(u)] = 1$,其中 $T'[i]$ 表示 T' 的第 i 位,最后返回 T' 。

3) BFCheck(H, T, u) 为检验算法。以 $H = (H_j)_{j \in [k]}$, $T \in \{0, 1\}^\ell$ 和元素 u 为输入,然后输出一个比特 $b = \bigwedge_{j \in [k]} T[H_j(u)]$, $\wedge$ 为逻辑与运算符。

定义 2 双线性映射。令 G 和 G_T 是 2 个阶为 p 的乘法循环群,其中, p 是一个长度为 λ 的素数,令 $g \in G$,则双线性映射 $\hat{e}: G \times G \rightarrow G_T$ 满足下述条件。

1) 双线性: 对任意 $x, y \in G$ 和 $a, b \in \mathbb{Z}_p$, 均成立 $\hat{e}(x^a, y^b) = \hat{e}(x, y)^{ab}$ 。

2) 非退化性: $\hat{e}(g, g) \neq 1$ 。

3) 可计算性: 对任意 $x, y \in G$, 存在一个多项式

时间算法能有效计算出 $\hat{e}(x, y)$ 。

定义 3 访问结构。令 P 是系统属性域, 一个定义在 P 上的访问结构 Γ 是指一个由非空属性子集构成的集合, 即 $\Gamma \subseteq 2^P \setminus \{\emptyset\}$ 。访问结构中的属性子集称为授权子集, 其他属性子集称为非授权子集。此外, 对任意属性子集 S 和 T , 若 $S \in \Gamma$ 并且 $S \subseteq T$ 时, 总成立 $T \in \Gamma$, 则称访问结构 Γ 是单调的。

定义 4 线性秘密共享机制 (Linear Secret Sharing Scheme, LSSS)^[30]。令 $\mathcal{U}$ 是系统属性域, p 是一个素数, 定义在 $\mathcal{U}$ 上的访问结构 Γ 的秘密共享机制在 $\mathbb{Z}_p$ 上是线性的, 是指: 1) 共享秘密 s 的所有共享份额构成 $\mathbb{Z}_p$ 上的一个向量; 2) 存在一个 $l \times n$ 的秘密共享机制的共享生成矩阵 M 和一个映射 $\rho: [l] \rightarrow P$, 对于任何一个随机选取的向量 $v = (s, v_2, \dots, v_n) \in \mathbb{Z}_p^n$, s 为共享秘密, 则 $M \cdot v$ 就是利用秘密共享机制得到的关于 s 的 l 个共享份额组成的向量, 其中 $\lambda_i = M_i \cdot v$ 是属于属性 $\rho(i)$ 的共享份额, M_i 是共享生成矩阵 M 的第 i 行。

LSSS 具有线性可重构性。若属性集 S 是访问结构 Γ 的一个授权属性子集, 即 $S \in \Gamma$, 定义共享生成矩阵 M 的一个行标子集 $I = \{i \in [l] \mid \rho(i) \in S\} \subseteq [l]$, 则可在多项式时间内计算出一组常数 $\{w_i \in \mathbb{Z}_p\}$, 使 $\sum_{i \in I} w_i \cdot M_i = (1, 0, \dots, 0)$, 可恢复 $s = \sum_{i \in I} (w_i \cdot M_i) \cdot v$; 若 $S \notin \Gamma$, 则存在向量 $w \in \mathbb{Z}_p^n$, 使 $w \cdot (1, 0, \dots, 0) = -1$, 并对任意 $i \in I$ 成立 $w \cdot M_i = 0$ 。

定义 5 判定性 l -BDHE 假设。令 G 和 G_T 是 2 个阶为 p 的乘法循环群, 其中 p 是一个长度为 λ 的素数, $\hat{e}: G \times G \rightarrow G_T$ 是一个双线性映射, $g \in G$ 是一个生成元, $a, s \in \mathbb{Z}_p$ 是两个随机数。对任意 $k \in [2l]$, $g_k = g^{a^k}$, 向量 $y = (g, g^s, g_1, g_2, \dots, g_l, g_{l+2}, \dots, g_{2l})$, 则 (G, G_T) 上的判定性 l -BDHE 问题是指给一个概率多项式时间算法 B 提供向量 y 和群元素 $T' \in G_T$, 算法 B 判断 $T' = \hat{e}(g^s, g_{l+1})$, 或者 T' 仅是群 G_T 中的一个随机元素 R ; 若 $T' = \hat{e}(g^s, g_{l+1})$, 算法 B 输出 0, 否则输出 1。

算法 B 解决 (G, G_T) 上的判定性 l -BDHE 问题的优势定义如下:

$$A_{\text{adv-}B}^{l\text{-BDHE}} = \left| \Pr \left[B(y, T' = \hat{e}(g^s, g_{l+1})) = 0 \right] - \Pr \left[B(y, T' = R) = 0 \right] \right|.$$

对任意一个概率多项式时间算法 B , 若其解决 (G, G_T) 上的判定性 l -BDHE 问题的优势均是可忽略的, 则称判定性 l -BDHE 假设在 (G, G_T) 上成立。

2 P-CP-ABE 方案

2.1 形式化定义

本文所提 P-CP-ABE 方案的形式化描述如下。

Setup($\lambda, \mathcal{U}$): 系统建立算法。以系统安全参数 λ 和系统属性域 $\mathcal{U}$ 为输入, 输出系统公开参数 R_{pp} 和系统主密钥 R_{MSK} 。

KeyGen(R_{pp}, R_{MSK}, S): 私钥生成算法。以 R_{pp} 、 R_{MSK} 和用户属性集 S 为输入, 输出用户属性集所对应的初始私钥 $K_{S, \phi}^{SK}$ 。

KPunc($R_{pp}, K_{S, P_{i-1}}^{SK}, t_i$): 私钥穿刺算法。以 R_{pp} 、用户的当前私钥 $K_{S, P_{i-1}}^{SK}$ 和标签 t_i 为输入, 输出一个更新的私钥 K_{S, P_i}^{SK} 。

Encrypt(R_{pp}, Γ, t, m): 加密算法。以 R_{pp} 、访问结构 Γ 、标签 t 和明文 m 为输入, 输出密文 C_{CT} 。

Decrypt($R_{pp}, C_{CT}, K_{S, P_i}^{SK}$): 解密算法。以 R_{pp} 、 C_{CT} 和 K_{S, P_i}^{SK} 为输入, 若 $S \in \Gamma$, 且 $t \notin P_i$, 则输出明文 m , 否则输出 $\perp$ 。

一个 P-CP-ABE 方案是正确的, 是指对任意正确生成的系统公开参数和主密钥 $(R_{pp}, R_{MSK}) \leftarrow \text{Setup}(\lambda, \mathcal{U})$ 、任意明文 m 、任意访问结构 Γ 、所有属性集 $S \in \Gamma$ 、任意标签 t 、任意标签集 $P = \{t_1, \dots, t_i\}$ 以及所有正确生成的属性私钥 $K_{S, \phi}^{SK} \leftarrow \text{KeyGen}(R_{pp}, R_{MSK}, S)$, 则下式成立:

$$\Pr [\text{Decrypt}(R_{pp}, \text{Encrypt}(R_{pp}, \Gamma, t, m), K_{S, \phi}^{SK}) = m] = 1.$$

对反复穿刺 q 次后的用户私钥 $K_{S, P_i}^{SK} \leftarrow \text{KPunc}(R_{pp}, K_{S, P_{i-1}}^{SK}, t_i) (i \in [q])$, 若 $t \notin P_i$, 则下式成立:

$$\Pr [\text{Decrypt}(R_{pp}, \text{Encrypt}(R_{pp}, \Gamma, t, m), K_{S, P_i}^{SK}) = m] = 1.$$

2.2 安全性定义

P-CP-ABE 方案的安全性可通过一个挑战者 C 和一个攻击者 A 之间的安全性游戏来刻画, 定义标签空间 $\mathcal{T} = \{0, 1\}^\eta$, 其中 η 为定长比特。具体如下。

初始化: 在该阶段, 攻击者 A 选择一个挑战访问结构 Γ^* 提交给挑战者 C 。

系统建立: 在该阶段, 挑战者 C 运行系统建立算法 $\text{Setup}(\lambda, \mathcal{U}) \rightarrow (R_{pp}, R_{MSK})$, 然后将 R_{pp} 发送给 A , 而自己持有 R_{MSK} 。

询问阶段 1: 在该阶段, A 可以自适应地进行多项式次数的私钥提取询问, C 利用 R_{MSK} 进行响应。

1) 私钥提取询问。 A 向 C 提交一个用户属性集 S , 要求 C 返回相应的私钥 $K_{S, \phi}^{SK}$ 。 C 收到该询问后, 直接执行 $\text{KeyGen}(R_{pp}, R_{MSK}, S) \rightarrow K_{S, \phi}^{SK}$, 然后将私钥 $K_{S, \phi}^{SK}$ 发送给 A 。

2) 穿刺询问。A 向 C 提交一个用户属性集 S , 要求 C 返回被标签 $t \in \mathcal{T}$ 穿刺的用户私钥 K_{S,P_i}^{SK} 。C 仅响应满足 $t \neq t^*$ 的穿刺询问。为此, C 依次调用私钥生成算法 $\text{KeyGen}(R_{\text{pp}}, R_{\text{msk}}, S) \rightarrow K_{S,\phi}^{\text{SK}}$, 私钥穿刺算法 $\text{KPunc}(R_{\text{pp}}, K_{S,P_{i-1}}^{\text{SK}}, t) \rightarrow K_{S,P_i}^{\text{SK}}$, 最后将穿刺的私钥 K_{S,P_i}^{SK} 发送给 A。

挑战阶段: 在该阶段, 攻击者 A 向挑战者 C 提交两个等长的明文消息 (m_0, m_1) , 并要求 A 在询问阶段 1 所进行的任意询问的输入满足下列条件: $S \notin \Gamma^*$ 或 $t \neq t^*$ 。C 选择一个随机比特 $b \in \{0, 1\}$ 和挑战标签 $t^* \in \mathcal{TV}_i$, 利用 Γ^* 对明文消息 m_b 加密, 然后, 将生成的挑战密文 C_{t^*} 发送给 A。

询问阶段 2: 该阶段类似于询问阶段 1, C 仍仅响应满足 $t \neq t^*$ 的穿刺询问。

猜测阶段: 最后, 攻击者 A 输出对 b 的猜测 b' 。若 $b' = b$, 则称 A 赢得了安全性游戏, 并定义其优势为 $A_{\text{adv-A}}^{\text{P-CP-ABE}} = |\Pr[b' = b] - 1/2|$ 。

定义 6 P-CP-ABE 的前向安全性。对任意一个针对 P-CP-ABE 方案的概率多项式时间攻击者 A, 若其赢得上述安全性游戏的优势 $A_{\text{adv-A}}^{\text{P-CP-ABE}}$ 是安全参数 λ 的一个可忽略函数, 则称该 P-CP-ABE 方案具有前向安全性。

3 方案的具体构造

本节在传统 CP-ABE 方案的基础上给出 P-CP-ABE 方案的具体构造, 其关键是实现高效的私钥穿刺操作, 为此, 利用布隆过滤器管理用户私钥。首先, 调用布隆过滤器生成算法得到哈希函数集与初始状态数组 $(H, T) \leftarrow \text{BFGen}(\ell, k)$, 用户私钥由 T 中每一位对应的私钥构件组成, 私钥结构与布隆过滤器状态绑定。解密操作时, 根据目标标签通过哈希函数集 H 定位到 T 中的 k 个位置, 从中选取满足 $T[i'] \neq 1$ 的索引 i' , 然后通过该位置对应的私钥构件进行解密, 保证仅使用未被穿刺的有效私钥构件。当需要用某标签 t 对私钥执行穿刺时, 首先调用布隆过滤器更新算法, 用标签 t 更新布隆过滤器状态 $\text{BFUpdate}(H, T, t) \rightarrow T'$, 然后将 T' 中与 t 映射的所有位置对应的私钥构件设置为 $\perp$, 完成私钥穿刺。穿刺后的私钥将失去对标签 t 对应密文的解密能力, 但不影响其他标签对应密文的正常解密。方案的具体构造由 5 个算法组成。

1) $\text{Setup}(\lambda, U)$ 。给定系统安全参数 λ 和系统属性个数 U , 首先生成双线性群 $(G, G_T, g, p, \hat{e})$ 。然

后, 选择随机整数 $\alpha, a \in \mathbb{Z}_p$, 选择随机群元素 $u_0 \in G$, 以及向量 $\mathbf{h} = (h_1, \dots, h_U) \in G^U$, 选择哈希函数 $\tilde{H}: N \rightarrow G$, 并令 $Z = \hat{e}(g, g)^\alpha$, $g_1 = g^a$, 设置系统主密钥 $R_{\text{msk}} = g^\alpha$, 输出系统公开参数 $R_{\text{pp}} = \{G, G_T, g, p, \hat{e}, g_1, u_0, \mathbf{h}, \tilde{H}, Z\}$ 。

2) $\text{KeyGen}(R_{\text{pp}}, R_{\text{msk}}, S)$ 。给定系统公开参数 R_{pp} 、系统主密钥 R_{msk} 和用户属性集 S 。首先, 调用布隆过滤器生成算法 $(H, T) \leftarrow \text{BFGen}(\ell, k)$, 然后, 选取随机整数 $r_1, r_2 \in \mathbb{Z}_p$ 。对任意 $j \in [\ell]$, 计算私钥构件:

$$R_{\text{sk}}^j = g^\alpha g^{ar_1} \cdot \tilde{H}(j)^{r_2} \quad (1)$$

最后, 输出用户属性集 S 对应的初始私钥:

$$K_{S,\phi}^{\text{SK}} = \{(H, T), \{R_{\text{sk}}^j\}_{j \in [\ell]}, \{K_x = h_x^{r_1} | x \in S\}, K_1 = g^{r_1}, K_2 = g^{r_2}\} \quad (2)$$

3) $\text{KPunc}(R_{\text{pp}}, K_{S,P_{i-1}}^{\text{SK}}, t_i)$ 。给定系统公开参数 R_{pp} 、当前的用户私钥 $K_{S,P_{i-1}}^{\text{SK}}$ 和一个标签 $t_i \in \mathcal{TV}_{i-1}$, 首先, 利用标签 t 调用布隆过滤器更新算法 $T' \leftarrow \text{BFUpdate}(H, T, t)$, 对任意 $j \in [\ell]$, 更新私钥构件如下:

$$R_{\text{sk}}^{j'} = \begin{cases} R_{\text{sk}}^j, & T'[j] = 0; \\ \perp, & T'[j] \neq 0. \end{cases} \quad (3)$$

最后, 更新 $P_i \leftarrow P_{i-1} \cup \{t_i\}$, 输出更新的私钥:

$$K_{S,P_i}^{\text{SK}} = \{(H, T'), \{R_{\text{sk}}^{j'}\}_{j' \in [\ell]}, \{K_x = h_x^{r_1} | x \in S\}, K_1, K_2\} \quad (4)$$

4) $\text{Encrypt}(R_{\text{pp}}, \Gamma, t, m)$ 。给定 R_{pp} 、访问结构 $(M_{l \times n}, \rho)$ 、标签 t 和明文 $m \in G_T$ 。首先, 选取随机向量 $\mathbf{v} = (s, v_2, \dots, v_n) \in \mathbb{Z}_p^n$, 其中, s 是共享秘密。对于任意 $i \in [l]$, 计算属性 $\rho(i)$ 所对应的共享份额 $\lambda_i = \mathbf{v} \cdot \mathbf{M}_i$, 其中, $\mathbf{M}_i$ 是共享矩阵 $\mathbf{M}$ 的第 i 行。然后, 若 $\text{BFCheck}(H, T, t) \rightarrow 0$, 则选取一个指标 $j^* \in \{j_1, \dots, j_k\}$, 其中 $j_y = H_y(t) (y \in [k])$ 。最后, 按如下方式生成密文:

$$C_{\text{CT}} = \{C, C', C'', \{C_i\}_{i \in [l]}\} \quad (5)$$

式中: $C = m \cdot Z^s$; $C' = g^s$; $C'' = \tilde{H}(j^*)^s$; $C_i = g^{a\lambda_i} h_{\rho(i)}^{-s}$ 。

5) $\text{Decrypt}(R_{\text{pp}}, C_{\text{CT}}, K_{S,P_i}^{\text{SK}})$ 。给定 R_{pp} 、密文 C_{CT} 和用户私钥 K_{S,P_i}^{SK} 。若 $S \notin (M_{l \times n}, \rho)$ 或 $t \in P_i$, 则输出 $\perp$; 否则, 令 $I = \{i \in [l] | \rho(i) \in S\}$, 并计算一组常数 $\{w_i \in \mathbb{Z}_p\}_{i \in I}$ 使得 $\sum_{i \in I} w_i \lambda_i = s$; 然后, 从 K_{S,P_i}^{SK} 中提取私钥部件 $R_{\text{sk}}^j, K_x, K_1, K_2$, 计算相应明文如下:

$$m = C \frac{\hat{e}(C'', K_2) \prod_{i \in I} \left(\hat{e}(C_i, K_1) \hat{e}(C', K_{\rho(i)}) \right)^{w_i}}{\hat{e}(C', R_{\text{sk}}^j)} \quad (6)$$

方案正确性对没有被密文标签 t 穿刺的私钥, 即 $t \notin P_i$, 将密钥构件和密文构件的具体值代入上述等式右边, 即可得:

$$\begin{aligned}
& C \frac{\hat{e}(C', K_2) \prod_{i \in I} \left(\hat{e}(C_i, K_1) \hat{e}(C', K_{\rho(i)}) \right)^{w_i}}{\hat{e}(C', R_{sk}')} = \\
& \frac{m \cdot Z^s \cdot \hat{e}(\tilde{H}(j^*), g^{r_2}) \prod_{i \in I} \left(\hat{e}(g^{a_i} h_{\rho(i)}^{-s}, g^{r_1}) \hat{e}(g^s, h_{\rho(i)}^{r_1}) \right)^{w_i}}{\hat{e}(g^s, g^a g^{ar_1} \cdot \tilde{H}(j)^{r_2})} = \\
& \frac{m \cdot \hat{e}(g, g)^{as} \cdot \hat{e}(\tilde{H}(j^*), g^{r_2}) \prod_{i \in I} \left(\hat{e}(g^{a_i}, g^{r_1}) \right)^{w_i}}{\hat{e}(g^s, g^a g^{ar_1} \cdot \tilde{H}(j)^{r_2})} = \\
& \frac{m \cdot \hat{e}(g, g)^{as} \cdot \hat{e}(g, \tilde{H}(j))^{sr_2} \cdot \hat{e}(g, g)^{ar_1 \sum_{i \in I} w_i \lambda_i}}{\hat{e}(g, g)^{as} \cdot \hat{e}(g, g)^{asr_1} \cdot \hat{e}(g, \tilde{H}(j))^{sr_2}} = m \quad (7)
\end{aligned}$$

4 安全性证明与性能分析

4.1 安全性证明

定理 1 P-CP-ABE 方案的选择性前向安全性。若判定性 l -BDHE 假设在双线性群 $(G, G_T, g, p, \hat{e})$ 上成立, 则 P-CP-ABE 方案在标准模型下满足选择性前向安全性。

证明 在证明过程中, 若存在一个概率多项式时间攻击者 A 以优势 $\text{Adv}_{\text{adv-A}}^{\text{P-CP-ABE}}(\lambda) = \varepsilon$ 攻破 P-CP-ABE 方案的选择性前向安全性, 则构造一个概率多项式时间算法 B 模拟安全性游戏中的挑战者 C, 并让 B 利用 A 攻破 P-CP-ABE 方案的优势解决 $(G, G_T, g, p, \hat{e})$ 上的判定性 l -BDHE 问题。

初始化: 攻击者 A 选择一个挑战访问结构 $(M_{\ell' \times n}^*, \rho^*)$ 和挑战标签 t^* 提交给算法 B。此外, 提供给 B 一个判定性 l -BDHE 问题实例 (y, T') , 其中, 向量 $y = (g, g^s, g_1, g_2, \dots, g_l, g_{l+2}, \dots, g_{2l})$, 对任意 $1 \leq k \leq 2l$ 且 $k \neq l+1$, $g_k = g^{a^k}$ 。

系统建立: 算法 B 选择一个随机整数 $\alpha' \in \mathbb{Z}_p$, 令 $Z = \hat{e}(g_1, g_l) \cdot \hat{e}(g, g)^{\alpha'} = \hat{e}(g, g)^{\alpha' + a^{l+1}}$, 显然, 算法 B 不知道 $(\alpha' + a^{l+1})$ 。对任意 $1 \leq x \leq U$, 选择随机整数 $z_x \in \mathbb{Z}_p$, 若存在整数 $1 \leq i \leq \ell^*$ 使得 $\rho^*(i) = x$, 则令 $h_x = g^{z_x} \cdot \prod_{j=1}^n g_j^{M_{ij}^*}$, 否则, 令 $h_x = g^{z_x}$; 设置系统主密钥 $R_{\text{MSK}} = g^{\alpha' + a^{l+1}}$ 。最后, 算法 B 令系统公开参数 $R_{\text{pp}} = ((G, G_T, g, p, \hat{e}), u_0, u_1, u_2, \dots, u_l, h_1, h_2, \dots, h_U)$ 并发送给攻击者 A。

询问阶段 1: 在该阶段, 允许 A 选择一个属性集 S , 自适应地询问相应的私钥。对于询问中涉及哈希函数 $\tilde{H}$ 的询问, 算法 B 模拟为: 对任意 $i \in [\ell]$, 选择一个随机整数 $t_i \in \mathbb{Z}_p$, 若有 $i \in \{H_j(t^*); j \in [k]\}$,

则令 $\tilde{H}(i) = g^{t_i}$, 否则, 令 $\tilde{H}(i) = g_2 \cdot g^{t_i}$ 。

1) 私钥提取询问。若 $S \notin (M_{\ell' \times n}^*, \rho^*)$, 依据 LSSS 的定义可知, 存在向量 $w = (w_1, \dots, w_n) \in \mathbb{Z}_p^n$ 使得 $w_1 = -1$, 并且对任意 $i \in I = \{i \in [\ell^*] \mid \rho^*(i) \in S\}$, 有 $w M_i = 0$ 。首先, 选择随机整数 $r'_1, r'_2 \in \mathbb{Z}_p$, 令 $r_1 = r'_1 + \sum_{j=1}^n w_j a^{l-j+1}$, 然后, 计算私钥构件:

$$\begin{cases} R_{sk}^i = g^{\alpha'} g^{ar_1} \cdot \tilde{H}(i)^{r_2} = g^{\alpha' + a^{l+1}} g^{a(r'_1 + \sum_{j=1}^n w_j a^{l-j+1})} \cdot \tilde{H}(i)^{r_2} = \\ g^{\alpha'} g_1^{r'_1} \prod_{j=2}^n g_j^{w_j} \cdot \tilde{H}(i)^{r_2} \\ K_1 = g^{r_1} = g^{r'_1} \prod_{j=1}^n g_j^{w_j a^{l-j+1}} = g^{r'_1} \prod_{j=1}^n g_j^{w_j} g_{l-j+1}, K_2 = g^{r_2} \end{cases} \quad (8)$$

对任意 $x \in S$, 若不存在整数 $1 \leq i \leq \ell^*$ 使得 $\rho^*(i) = x$, 则令 $K_x = h_x^{r_1} = g^{r_1 z_x} = K_1^{z_x}$; 若存在整数 $1 \leq i \leq \ell^*$ 使得 $\rho^*(i) = x$, 则令

$$\begin{aligned} K_x &= h_x^{r_1} = \left(g^{z_x} \cdot \prod_{j=1}^n g_j^{M_{ij}^*} \right)^{r_1} = K_1^{z_x} g^{\left(r'_1 + \sum_{j=1}^n w_j a^{l-j+1} \right) \sum_{j=1}^n M_{ij}^*} = \\ & K_1^{z_x} \prod_{j=1}^n \left(g_j^{r'_1} \prod_{\gamma=1, \gamma \neq j}^n g_l^{w_j} g_{l-j+\gamma+1} \right)^{M_{ij}^*} \end{aligned} \quad (9)$$

可以看出, 在算法 B 不知道系统主密钥的情况下, 仍可以计算出私钥, 并且与私钥提取算法所产生的私钥是同分布的。最后, B 将计算的私钥 $K_{S, \text{sk}}^{\text{SK}} = \{\{R_{sk}^i\}_{i \in [\ell]}, \{K_x \mid x \in S\}, K_1, K_2\}$ 发送给 A, 同时更新集合 $Q_{sk} \leftarrow Q_{sk} \cup \{S\}$ 。

2) 私钥穿刺询问。若给定的属性集 S 满足挑战访问结构 $(M_{\ell' \times n}^*, \rho^*)$, 此时为 A 生成被标签 t 穿刺的用户私钥, B 选择随机整数 $r_1, r'_2 \in \mathbb{Z}_p$, 令 $r_2 = r'_2 - a$, 则 $K_1 = g^{r_1}$, $K_2 = g^{r_2} = g^{r'_2} \cdot g^{-a}$, 对任意 $x \in S$, 令 $K_x = h_x^{r_1}$, 对任意 $i \in [\ell]$, 若 $i \in \{H_j(t); j \in [k]\}$, 则令 $R_{sk}^{i'} \leftarrow \perp$, 否则计算私钥:

$$\begin{aligned} R_{sk}^{i'} &= g^{\alpha'} g^{ar_1} \cdot \tilde{H}(i)^{r_2} = g^{\alpha' + a^{l+1}} g_1^{r_1} \cdot \tilde{H}(i)^{r'_2} \cdot (g_2 g^{t_i})^{-a} = \\ & g^{\alpha'} g_1^{r_1} \cdot \tilde{H}(i)^{r'_2} \cdot g^{-at_i} \end{aligned} \quad (10)$$

可以看出, 式(10)私钥构件对 B 都是可计算的。

最后, 算法 B 将生成的穿刺私钥 $K_{S, P_i}^{\text{SK}} = \{(H, T'), \{R_{sk}^{i'}\}_{i \in [\ell]}, \{K_x \mid x \in S\}, K_1, K_2\}$ 返回给攻击者 A, 并更新集合 $Q_{\text{punc}} \leftarrow Q_{\text{punc}} \cup \{(S, t)\}$ 和 $P_i \leftarrow P_{i-1} \cup \{t\}$ 。

挑战阶段: 当 A 结束询问阶段之后, 提交两个挑战明文 (m_0, m_1) , 算法 B 随机选择 $b \in \{0, 1\}$ 和挑战标签 $t^* \in \mathcal{T} \setminus P_i$, 然后, 选择随机数 $\theta'_2, \dots, \theta'_n \in \mathbb{Z}_p$, 令 $\theta'_1 = 0$, 向量 $\theta^* = (s, sa + \theta'_2, \dots, sa^{n-1} + \theta'_n)$, 对任意

$i \in [\ell^*]$, 令 $\lambda_i^* = \theta^* \cdot M_i^* = \sum_{j=1}^{n^*} (s\alpha^{j-1} M_{ij}^* + \theta_j')$, 则密

文构件可表示为

$$C_i^* = g^{a\lambda_i^*} h_{\rho^*(i)}^{-s} = g^{\sum_{j=1}^{n^*} (s\alpha^j M_{ij}^* + a\theta_j' M_{ij}^*)} \left(g^{z_{\rho^*(i)}} \prod_{j=1}^{n^*} g_j^{M_{ij}^*} \right)^{-s} =$$

$$\prod_{j=1}^{n^*} g_j^{sM_{ij}^*} \prod_{j=1}^{n^*} g_1^{\theta_j' M_{ij}^*} (g^s)^{-z_{\rho^*(i)}} \prod_{j=1}^{n^*} g_j^{-sM_{ij}^*} =$$

$$\prod_{j=1}^{n^*} g_1^{\theta_j' M_{ij}^*} (g^s)^{-z_{\rho^*(i)}} \quad (11)$$

生成挑战密文如下:

$$C_{C^*} = \left\{ C^*, C'^*, C''^*, \{C_i^*\}_{i \in [l]} \right\} \quad (12)$$

式中: $C^* = m_b \cdot Z^s = m_b T' \hat{e}(g^s, g^{\alpha'})$; $C'^* = g^s$; $C''^* = \tilde{H}(j^*)^s$ 。

最后, B 将生成的挑战密文 C_{C^*} 发送给 A 。

询问阶段2: 该阶段的询问与第1阶段相似。

猜测阶段: 最后, 攻击者 A 输出对 b 的猜测 b' 。

若 $b'=b$, 则 B 输出 0, 即 $T' = \hat{e}(g^s, g_{l+1}) = \hat{e}(g, g)^{\alpha^{l+1}}$; 否则, 输出 1, 即判定 T' 仅是群 G_T 中的一个随机元素。

概率分析, 若 $T' = \hat{e}(g^s, g_{l+1})$, 则 $C^* = m_b T' \hat{e}(g^s, g^{\alpha'}) = m_b \hat{e}(g, g)^{s(\alpha^l + \alpha^{l+1})} = m_b Z$, 即 B 所模拟的安全性游戏与真实的安全性游戏完全相同, 从而 $\Pr[B(y, T') = \hat{e}(g^s, g_l)] = 0 = \frac{1}{2} + A_{\text{adv-A}}^{\text{P-CP-ABE}}$; 若 T' 仅是群 G_T 中的一个随机元素 R , 则从攻击者 A 的角度看, 明文 m_b 的

信息被完全隐藏, 从而 $\Pr[B(y, T') = 0] = \frac{1}{2}$ 。

综上, 则有:

$$A_{\text{adv-B}}^{\text{q-BDHE}}(\lambda) =$$

$$\left| \Pr[B(y, T') = \hat{e}(g^s, g_l)] - \Pr[B(y, T') = 0] \right| =$$

$$\left| \Pr[b' = b] - \frac{1}{2} \right| \geq A_{\text{adv-A}}^{\text{P-CP-ABE}}(\lambda) = \varepsilon \quad (13)$$

4.2 性能分析

本节从算法效率和存储规模方面与其他相关加密方案进行比较。在比较过程中, 用 E 表示 1 次模指数运算, P 表示 1 次双线性对运算, H 表示 1 次哈希运算, l 为访问结构中属性个数, U 为系统全局属性总数, $|S|$ 表示用户属性集合规模, ℓ 为布隆过滤器数组长度, n 表示标签的数量, k 为布隆过滤器哈希函数个数。分析存储开销时, 以群 G 中 1 个元素的长度为基本单位。

从表 1 可以看出, 本文方案的密钥穿刺算法仅涉及固定次数的哈希运算和位更新, 计算开销几乎可以忽略, 而文献[18]方案的密钥穿刺开销与总的标签数量成线性关系。此外, 本文方案的私钥存储具有较大开销, 这主要是因为布隆过滤器所对应的位数组的每一个位置上都存储一个私钥构件, 以此实现细粒度的前向安全性。此外, 本文方案的公开参数规模和密文长度与文献[4]方案相当, 而文献[18]方案具有更大的公开参数规模。

表 1 相关方案的计算/存储开销对比

方案	计算效率			存储规模		
	加密	解密	私钥穿刺	系统公共参数	私钥	密文
文献[4]	$(2l+2)E$	$2P + lE$	—	$U+3$	$ S +3$	$l+2$
文献[18]	$(l+n)E$	$ Q P + (n+l)E$	nE	$n+U+5$	$3 Q + S +4$	$l+n+3$
本文方案	$(2l+3)E$	$2P + lE$	kH	$U+4$	$(S + \ell + 2)$	$l+3$

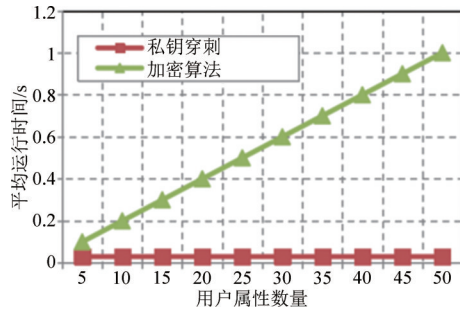
4.3 仿真实验

将本文 P-CP-ABE 方案在 Charm^[31] 框架下进行实现, 以验证其实际性能, 测试环境为具有 16 GB 内存和 i7-1160G7@1.20 GHz CPU 的笔记本电脑。在仿真实验中, 布隆过滤器的假阳性概率设置为 $\delta = 10^{-3}$, 哈希函数个数 $k = 10$, 用户属性数量设置为 5~50, 系统标签最大数量设置为 2~1 024, 哈希函数均为 SHA-256, 椭圆曲线为 SS512 曲线, 所有实验均重复执行 20 次, 取平均值作为最终结果。

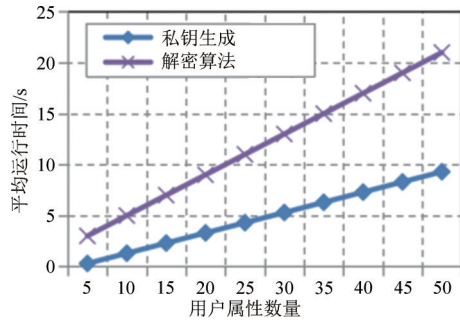
图 1 给出了在不同的用户属性数量下, 本文所提 P-CP-ABE 方案中各个算法的执行效率。可以发现, 私钥生成、加密与解密算法的平均运行时间

随用户属性集规模呈线性增长趋势。这些算法中, 针对每个属性至少需要执行一次指数运算。此外, 本文所提方案的私钥穿刺算法执行效率非常高, 且开销保持常数级。

图 2 展示了系统标签最大数量对所提方案中各算法运行时间的影响。从实验结果可以看出, 私钥生成算法的平均运行时间随系统标签最大数量的增加呈快速增长趋势; 另一方面, 本文所提 P-CP-ABE 方案的加密和解密算法的执行效率受该参数的影响较小。值得注意的是私钥穿刺算法的平均运行时间几乎不受标签数量的影响, 其开销仅与布隆过滤器哈希函数个数相关。

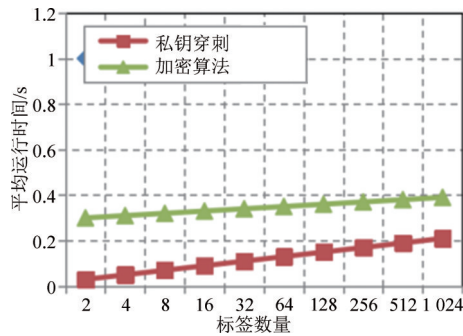


(a) 私钥穿刺算法和加密算法

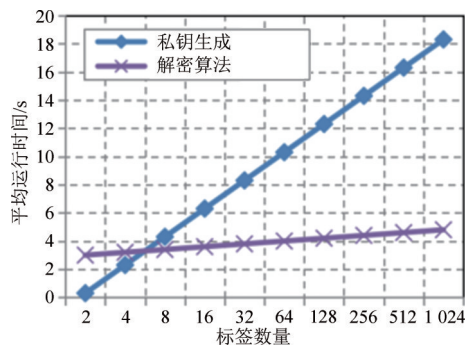


(b) 私钥生成和解密算法

图1 不同用户属性数量下各算法执行效率



(a) 私钥穿刺算法和加密算法



(b) 私钥生成和解密算法

图2 不同标签数量下各算法执行效率(横轴为以2为底对数坐标)

5 结束语

本文以传统的CP-ABE方案为基础,引入可穿刺公钥加密思想,实现了一种可穿刺的密文策略基于属性加密方案,有效解决了传统前向安全CP-ABE方案灵活性不足、私钥更新效率偏低的问题。文中首先给出所提方案的形式化与安全性定义,

随后给出方案的具体构造。方案基于布隆过滤器管理私钥构建和穿刺标签的位置映射,其私钥穿刺过程仅需执行哈希运算与位更新操作,即可实现细粒度的前向安全性与用户私钥的高效更新。基于判定性 I -BDHE假设,本文在标准模型下证明了所提方案满足选择性前向安全性。最后,通过性能分析和仿真实验结果验证了所提方案在计算效率、存储规模方面的优势。

参考文献:

- [1] Sahai A, Waters B. Fuzzy identity-based encryption[C]// Advances in cryptology-EUROCRYPT 2005. Berlin, Germany: Springer, 2005:457-473.
- [2] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data[C]// Proceedings of the 13th ACM conference on computer and communications security. New York, USA: ACM, 2006: 89-98.
- [3] Chen S B, Li J G, Zhang Y C, et al. Efficient revocable attribute-based encryption with verifiable data integrity[J]. IEEE Internet of Things Journal, 2024, 11 (6) : 10441-10451.
- [4] Waters B. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization [C] //Public key cryptography-PKC 2011. Berlin, Germany: Springer, 2011:53-70.
- [5] Delerablée C, Gouriou L, Pointcheval D. Key-policy ABE with switchable attributes[C]//Security and cryptography for networks. Cham, Switzerland: Springer, 2022: 147-171.
- [6] Luo F C, Wang H Y, Yan X F, et al. Key-policy attribute-based encryption with switchable attributes for fine-grained access control of encrypted data[J]. IEEE Transactions on Information Forensics and Security, 2024, 19: 7245-7258.
- [7] Anderson R. Two remarks on public key cryptography [C]// Proceedings of the invited lecture at the 4th ACM conference on computer and communications security. New York, USA: ACM, 1997:1-5.
- [8] Bellare M, Miner S K. A forward-secure digital signature scheme [C]//Advances in cryptology-CRYPTO' 99. Berlin, Germany: Springer, 1999:431-448.
- [9] Itkis G, Reyzin L. Forward-secure signatures with optimal signing and verifying [C]//Advances in cryptology-CRYPTO 2001. Berlin, Germany: Springer, 2001:332-354.
- [10] Wu B, Wang C F, Li Y H, et al. A efficient lattice-based forward-secure linearly homomorphic signature scheme for network coding[J]. Telecommunication Systems, 2024,

- 87(4):1129-1145.
- [11] Libert B, Quisquater J J, Yung M. Forward-secure signatures in untrusted update environments: efficient and generic constructions [C] // Proceedings of the 14th ACM conference on computer and communications security. New York, USA : ACM, 2007: 266-275.
- [12] Canetti R, Halevi S, Katz J. A forward-secure public-key encryption scheme [C] // Advances in cryptology-EUROCRYPT 2003. Berlin, Germany: Springer, 2003: 255-271.
- [13] Gentry C, Silverberg A. Hierarchical ID-based cryptography [C] // Advances in cryptology-ASIACRYPT 2002. Berlin, Germany: Springer, 2002: 548-566.
- [14] Yu J, Kong F Y, Cheng X G, et al. Forward-secure identity-based public-key encryption without random oracles [J]. *Fundamenta Informaticae*, 2011, 111(2): 241-256.
- [15] 魏江宏, 刘文芬, 胡学先. 前向安全的密文策略基于属性加密方案 [J]. *通信学报*, 2014, 35(7): 38-45.
- [16] Wei J H, Liu W F, Hu X X. Forward-secure identity-based signature with efficient revocation [J]. *International Journal of Computer mathematics*, 2017, 94(7): 1390-1411.
- [17] Oh H, Kim J, Shin J S. Forward-secure ID based digital signature scheme with forward-secure private key generator [J]. *Information Sciences*, 2018, 454: 96-109.
- [18] Green M D, Miers I. Forward secure asynchronous messaging from puncturable encryption [C] // Proceedings of the 2015 IEEE symposium on security and privacy. Piscataway, USA; IEEE, 2015: 305-320.
- [19] Günther F, Hale B, Jager T, et al. 0-RTT key exchange with full forward secrecy [C] // Advances in cryptology-EUROCRYPT 2017. Cham, Switzerland: Springer, 2017: 519-548.
- [20] Cini V, Ramacher S, Slamanig D, et al. CCA-secure (puncturable) KEMs from encryption with non-negligible decryption errors [C] // Advances in cryptology-ASIACRYPT 2020. Cham, Switzerland: Springer, 2020: 159-190.
- [21] Derler D, Jager T, Slamanig D, et al. Bloom filter encryption and applications to efficient forward-secret 0-RTT key exchange [C] // Advances in cryptology-EUROCRYPT 2018. Cham, Switzerland: Springer, 2018: 425-455.
- [22] Li X Y, Xu J, Fan X, et al. Puncturable signatures and applications in proof-of-stake blockchain protocols [J]. *IEEE Transactions on Information Forensics and Security*, 2020, 15: 3872-3885.
- [23] Wei J H, Chen X F, Wang J F, et al. Forward-secure puncturable identity-based encryption for securing cloud emails [C] // Computer security-ESORICS 2019. Cham, Switzerland: Springer, 2019: 134-150.
- [24] Wei J H, Chen X F, Wang J F, et al. Enabling (end-to-end) encrypted cloud emails with practical forward secrecy [J]. *IEEE Transactions on Dependable and Secure Computing*, 2022, 19(4): 2318-2332.
- [25] Wei J H, Chen X F, Wang J F, et al. Securing fine-grained data sharing and erasure in outsourced storage systems [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2023, 34(2): 552-566.
- [26] Derler D, Ramacher S, Slamanig D, et al. Fine-grained forward secrecy: allow-list/deny-list encryption and applications [C] // Financial cryptography and data security. Berlin, Germany: Springer, 2021: 499-519.
- [27] 巴阳, 陈越, 胡学先, 等. 基于智能合约与 ABAC 的访问控制方案 [J]. *信息工程大学学报*, 2022, 23(5): 608-616.
- [28] 张敏, 李军, 王浩. 基于区块链与属性基加密的数据共享方案 [J]. *信息工程大学学报*, 2025, 26(1): 35-42.
- [29] Bloom B H. Space/time trade-offs in hash coding with allowable errors [J]. *Communications of the ACM*, 1970, 13(7): 422-426.
- [30] Beimel A. Secure schemes for secret sharing and key distribution [D]. Haifa, Israel: Israel Institute of Technology, Technion, 1996: 75-108.
- [31] Akinyele J A, Garman C, Miers I, et al. Charm: a framework for rapidly prototyping cryptosystems [J]. *Journal of Cryptographic Engineering*, 2013, 3(2): 111-128.

(编辑: 邵 华)