

支持双向匹配的可穿刺基于属性可搜索加密

陈礼青¹, 郭笑笑¹, 刘志全²

(1. 淮安大学计算机与软件工程学院, 江苏 淮安 223003; 2. 暨南大学网络空间安全学院, 广东 广州 510632)

摘要: 针对车载社交网络 (VSN) 中现有基于属性可搜索加密方案存在单向访问控制与低效撤销机制的问题, 提出了一种支持双向匹配的可穿刺基于属性可搜索加密 (Pun-ABSE-BM), 并对其系统模型、形式化定义及安全模型进行了阐述。构造了一个具体的 Pun-ABSE-BM 方案, 该方案通过双向策略匹配实现了收发双方的权限互控, 并融合穿刺加密技术以支持灵活高效的用户撤销, 同时集成了撤销验证、离线/在线加密及外包解密功能。安全性证明和性能分析表明, 所提方案满足 VSN 的安全需求并兼具优异的效率与实用性。

关键词: 基于属性可搜索加密; 双向匹配; 用户撤销; 车载社交网络

中图分类号: TP309

文献标志码: A

doi: 10.11959/j.issn.1000-436x.TXXB260136

Puncturable attribute-based searchable encryption with bidirectional matchmaking

Chen Liqing¹, Guo Xiaoxiao¹, Liu Zhiquan²

1. School of Computer and Software Engineering, Huai'an University, Huai'an 223003, China

2. College of Cyber Security, Jinan University, Guangzhou 510632, China

Abstract: To address the issues of one-way access control and inefficient revocation mechanism in existing attribute-based searchable encryption schemes for vehicular social network (VSN), a puncturable attribute-based searchable encryption with bidirectional matchmaking (Pun-ABSE-BM) was proposed. The system model, formal definition, and security model of the proposed scheme were described. A concrete Pun-ABSE-BM scheme was constructed, which achieved mutual permission control between senders and receivers through bidirectional policy matching. By incorporating puncturable encryption technology, it enabled efficient user revocation without requiring ciphertext re-encryption or key updates. The scheme also integrated revocation verification, offline/online encryption, and outsourced decryption. Security proof and performance analysis demonstrate that the proposed scheme meets the security requirements of VSN and offers satisfactory efficiency and practicality.

Key words: attribute-based searchable encryption, bidirectional matchmaking, user revocation, vehicular social network

0 引言

在交通强国战略背景下, 智慧交通推动传统车辆演进为智能交互主体, 交通社交化成为重要趋势。车载社交网络 (vehicular social network,

VSN)^[1-2]作为新型数据交互模式, 是连接出行主体与交通系统的关键纽带。VSN 以兴趣驱动数据共享为核心, 将传统“人-车-路-云”架构拓展为动态社交化网络, 融合多元节点并依托车联网通信

收稿日期: 2026-03-20; 修回日期: 2026-05-17

通信作者: 陈礼青, chenlq@hau.edu.cn

基金项目: 国家自然科学基金资助项目 (No.62272195); 江苏省高等学校基础科学 (自然科学) 研究重大基金资助项目 (No.25KJA520003); 江苏省产学研合作基金资助项目 (No.BY20250234)

Foundation Items: The National Natural Science Foundation of China (No.62272195), The Major Program of Basic Science (Natural Science) Research in Higher Education Institutions of Jiangsu Province of China (No.25KJA520003), The Industry-University-Research Collaboration Program of Jiangsu Province of China (No.BY20250234)

技术实现全场景数据交互,为智慧交通精细化管控与协同化服务提供支撑。数据发布者通过VSN上传行驶状态、环境感知及交通预警等信息。数据请求者通过检索共享数据优化出行路径。车载云服务器(vehicle cloud server, VCS)承担海量数据存储与跨域共享任务,是保障VSN高效运行的基础设施。

VSN的开放社交属性与分布式架构使其数据共享面临隐私保护、权限管控和实时性挑战^[3]。一方面,车联网敏感数据通过无线开放信道传输,存在隐私泄露与非授权访问风险。为此,研究者引入基于属性加密(attribute-based encryption, ABE)^[4],提出了面向车载场景的隐私保护与细粒度访问控制方案^[5-8],实现一对多数据共享的灵活权限管控。另一方面,传统加密共享模式需下载解密完整密文方能筛选信息,难以满足VSN的实时交互需求。

可搜索加密(searchable encryption, SE)^[9]支持对密文进行关键词检索,不需要解密密文数据即可实现目标内容筛选,是VSN场景加密数据高效检索的核心技术。Zheng等^[10]首次结合ABE与SE,提出了基于属性关键词搜索(attribute-based keyword search, ABKS),并分别基于密文策略(ciphertext-policy, CP)和密钥策略(key-policy, KP)给出CP-ABKS与KP-ABKS方案。后续研究进一步提出了基于属性可搜索加密(attribute-based searchable encryption, ABSE)方案,其能在实现访问控制的同时支持加密数据检索,为VSN场景下的安全高效数据共享提供了理想的技术框架。国内外学者围绕该技术持续优化,并提出了一系列功能丰富和性能优异的ABSE方案^[11-14]。

现有ABSE方案多采用单向访问控制,即访问策略与属性的绑定方向唯一。这类方案虽能满足传统数据共享场景的基本需求,但在VSN环境中,对访问控制的灵活性与安全性提出了更高要求:用户不仅需要满足发送方策略,还希望对发送方进行筛选,仅解密具备特定属性的发送方生成的密文。针对此问题,文献^[15]提出了基于属性的匹配加密(matchmaking encryption, ME),发送方同时使用自身属性与接收方属性加密数据,由雾节点验证发送方策略,接收者验证接收方策略。文献^[16]提出了基于属性的访问控制加密,发送方采用CP-ABE实现对接收方的细粒度访问控制,边缘服务器通过预定义策略保障信息的有效性。仅当双方属性均满足

策略时,通信方可进行。

当车辆作为社交节点接入VSN时,其访问权限常因退出社交群组、权限过期或跨区域移动而发生动态变更。例如,车辆退出某路况共享群组后,其对群组内敏感数据的访问权限必须被即时撤销。若缺乏有效的权限撤销机制,已失去权限的车辆仍可凭借留存的密钥搜索并访问加密数据,从而引发数据隐私泄露,严重威胁VSN数据共享的安全性与可信度。针对该问题,国内外学者提出了多种支持用户撤销的方案。现有撤销机制主要分为两类:一是直接撤销^[6,17-20],由数据拥有者(data owner, DO)在加密阶段嵌入被撤销用户列表,使列表内用户的解密权限直接失效;二是间接撤销^[21-23],通过授权机构定期为未被撤销的合法用户更新密钥,同步由云端服务器完成密文重加密操作。

1 相关工作

ABE由Sahai等^[4]提出的模糊基于身份加密发展而来,是实现加密数据细粒度访问控制的关键技术。根据密钥与访问策略的关联方式,ABE可分为KP-ABE^[24]和CP-ABE^[25]。针对不同应用场景,研究者提出了多种改进方案。Feng等^[6]设计了支持并行外包解密的树访问结构ABE方案,以适应边缘智能车联网。为解决传统ABE密钥托管问题,Wang等^[7]提出了一种适用于VSN的可穿刺注册ABE方案。Li等^[26]针对虚假策略攻击,提出了带策略认证的可追踪可撤销CP-ABE方案。文献^[27]面向机密文档安全共享,设计了抵抗量子攻击的可穿刺CP-ABE方案。

单纯的ABE难以满足加密数据检索需求,Zheng等^[10]提出了ABKS,在细粒度访问控制基础上支持关键字检索,降低了本地存储与通信开销。后续研究进一步扩展:Zhang等^[11]基于区块链实现匿名ABSE,支持策略隐藏、可验证与恒定开销;Ghopur等^[17]提出了去中心化多权威CP-ABSE,支持多关键词检索与属性撤销;Chen等^[18]将多权威可搜索加密与属性撤销应用于车联网;Chen等^[19]进一步设计了部分隐藏策略的多关键词可搜索加密,具备验证与撤销功能;李宇溪等^[28]提出了基于区块链索引的可验证布尔搜索加密方案。

ME由Ateniese等^[29]首次提出,要求通信双方分别设定消息解密访问控制策略,并基于身份场景

设计了高效的基于身份匹配加密 (identity-based matchmaking encryption, IB-ME) 方案。随后, Wu 等^[30]将其扩展至可撤销多接收方场景。文献[31]给出了基于标准格的非黑盒 IB-ME 构造以抵抗量子攻击。文献[32]则在 SXDH (symmetric external Diffie-Hellman) 假设下提出了当前效率最高的 IB-ME 方案, 并进一步设计了支持等值策略的内积匹配加密方案。上述方案主要基于身份场景, Ateniese 等^[29]还提出了 ME 的变体, 编排式 ME (arranged matchmaking encryption, A-ME), 将属性与策略密钥合并。受此启发, Cui 等^[16]提出了基于属性 ME, 实现收发双方双向细粒度访问控制, 密钥生成中心 (key generation center, KGC) 分别为收发双方生成包含属性集和策略的密钥。文献[15]则采用中央权力机构 (central authority, CA) 为收发双方颁发属性密钥, 通过预定义访问策略、全局访问策略及代理重加密实现双向匹配。

用户撤销机制是实现 VSN 动态访问控制的关键技术。Pirretti 等^[33]首次提出了通过周期性更新属性版本号与私钥实现间接撤销。文献[23]进一步协同更新属性版本密钥、公私钥及密文构建间接撤销机制。Bethencourt 等^[25]与 Jiang 等^[34]分别将时间因子嵌入密文与私钥以限制有效期, 但仍面临重加密与密钥更新的高开销。可穿刺加密 (puncturable encryption, PE)^[35]为用户撤销提供了新思路, Phuong 等^[36]首次将 PE 与 ABE 结合, 其密钥具有可穿刺特性, 可针对特定消息、接收方或时间段撤销解密权限。基于此, Yang 等^[27]提出了基于格的可穿刺 CP-ABE 方案, 通过标签穿刺实现解密权限的撤销。Ghopur 等^[37]则通过在密文中嵌入穿刺结构, 不需要重加密与密钥更新即可完成用户撤销, 显著降低了撤销开销。

2 本文工作

结合上述研究背景, 本文着重解决以下两个核心问题: 如何设计一种支持双向匹配的 ABSE 方案; 如何进一步实现灵活且高效的撤销机制。

2.1 支持双向匹配的 ABSE 方案

本文借鉴 ME 中双向匹配的概念: 通信双方同时具备属性与策略, 要求发送方属性满足接收方策略, 且接收方属性满足发送方策略。本文在设计时考虑: KP-ABKS 使用属性加密关键词, CP-ABE

使用策略加密明文。因此, 本文结合 KP-ABKS 与 CP-ABE 机制: 前者中接收方持有访问策略和发送方关联属性, 用于实现接收方对发送方的筛选; 后者中接收方关联属性和发送方制定策略, 用于实现发送方对接收方的访问控制。本文通过将二者融合, 构建支持双向匹配的 ABSE 方案。

在设计上述融合架构时, 由于方案同时整合了 KP-ABKS 与 CP-ABE 两种底层原语, 需考虑双向匹配机制中极易产生的“混合与匹配”安全威胁: 恶意的开放信道敌手可能将异源的组件篡改或替换。

为了解决该问题, 本文在密钥生成时, 通过用户级盲化因子绑定密钥组件, 将该因子作为锚点深度耦合至搜索与解密组件的多项式中。在加密时, 采用混合加密架构, 通过哈希函数派生密钥将复合密文紧密绑定, 从根源上杜绝了发送方或云服务器对密文组件的恶意篡改与替换。在令牌生成时, 引入动态签名, 该签名将令牌组件、转换密钥和时间戳进行强绑定。这使任何试图跨身份盗用令牌、替换外包密钥的拼接行为, 都会在协议层或最终解密层导致算法异常终止。

图 1 展示了本文设计的双向匹配及防御混合与匹配攻击的关键加密逻辑。发送方和接收方同时具备属性集与策略, 通过结合底层加密逻辑与上层协议, 使其组件不可随意篡改或替换。车载云服务器接收到搜索请求后, 只有当发送方属性满足接收方策略且接收方属性满足发送方策略时, 接收方才能获取密文。随后, 接收方需使用自身的盲化因子解密并获取明文。

2.2 灵活且高效的撤销机制

在实现双向匹配的基础上, 本文进一步研究用户撤销问题。现有撤销机制主要分为直接撤销与间接撤销。直接撤销要求 DO 预先确定撤销用户列表。间接撤销不需要事先确定撤销用户列表, 但需重加密密文及更新未撤销用户密钥。为克服上述缺陷, 本文借鉴文献[37]的思路, 将 PE 与 CP-ABE 结合, 用户私钥绑定特定标签。当撤销用户权限时, DO 制定限制策略并发送至服务器进行穿刺。若用户密钥中的标签与限制策略匹配, 则该密钥无法解密密文。该方式不需要预先确定撤销用户列表, 也不需要重加密密文或更新密钥, 且限制策略采用树形结构支持 AND 门和 OR 门, 因此能够实现灵活且高效的撤销。

$F(\text{Atts}, T_{v_j}) = 1$ 。

(2) 秘密分配算法 $\text{Share}(T, s)$ 。给定秘密值 s ，为树 T 的每个叶子节点生成秘密份额。算法自顶向下为每个节点 v 构造一个 $k_v - 1$ 次多项式 q_v 。

若 v 是根节点，令 $q_v(0) = s$ ，并随机选择 $k_v - 1$ 个系数定义多项式 q_v 。

若 v 是非根节点，令 $q_v(0) = q_{\text{parent}(v)}(\text{ind}(v))$ ：若 v 是内部节点，再随机选择 v 个系数定义多项式 q_v ；若 v 是叶子节点，其多项式次数为 0，故 $q_v(0)$ 为该叶子节点的秘密份额。

算法最后输出所有叶子节点的份额 $\{q_v(0) \mid v \in \text{lvs}(T)\} \leftarrow \text{Share}(T, s)$ 。

(3) 秘密重构算法 $\text{Combine}(T, \{E_v\}_{v \in \text{Atts}})$ 。其中 $v \in \text{lvs}(T)$ 且 $F(\text{Atts}, T) = 1$ ，已知这些叶子节点的值 $E_v = e(g, h)^{q_v(0)}$ ，其中 $g, h \in G$ ， e 为双线性映射。算法自底向上计算每个节点 v 的值 E_v 。

若 $F(\text{Atts}, T_v) = 0$ ，则 $E_v = \perp$ 。

若 $F(\text{Atts}, T_v) = 1$ ，若 v 是叶子节点，则 $E_v = e(g, h)^{q_v(0)}$ ；若 v 是内部节点，对于其子节点 $v_1, \dots, v_{\text{num}_v}$ ，存在一组索引 $S \in \{v_1, \dots, v_{\text{num}_v}\}$ 使 $|S| = k_v$ ， $j \in S$ ， $F(\text{Atts}, T_{v_j}) = 1$ 。计算 $E_v = \prod_{j \in S} E_{v_j}^{\Delta_{j,S(0)}}$ $E(g, h)^{q_v(0)}$ ， $\Delta_{j,S(0)} = \prod_{l \in S, l \neq j} \frac{0 - l}{j - l}$ 。递归进行直至根节点，最终得到 $E_{\text{root}} = e(g, h)^{q_{\text{root}}(0)} = e(g, h)^s$ ，即重构的秘密。

尽管线性秘密共享方案 (linear secret sharing scheme, LSSS) 是当前主流的访问结构，但其在处理复杂的布尔逻辑时，计算和存储开销是相对固定的。本文方案访问结构最终采用访问树结构，主要基于以下考虑。

首先，本文方案属于轻量化设计，适用于中小型场景 (5~10 个核心属性)，访问树的秘密分配仅需几次简单的多项式求值，其效率实际上与线性秘密共享方案相当。其次，访问树天然支持阈值门，在处理 VSN 中常见的 k -out-of- n 阈值策略时更有优势。再者，访问树和拉格朗日插值法，在逻辑表达上与本文提出的可穿刺撤销机制契合度更高，能更自然地支持穿刺组件的构造。最后，本文方案通过离线/在线加密、外包解密机制，有效弥补了访问树的性能短板。

3.3 德摩根定律

由于访问策略是正向单调的，当策略表示否时，需通过德摩根定律将其转换为正向的访问策略。德摩根定律的表述如下。

(1) $\neg(A \vee B)$ 等价于 $(\neg A) \wedge (\neg B)$ 。

(2) $\neg(A \wedge B)$ 等价于 $(\neg A) \vee (\neg B)$ 。

3.4 计算复杂性假设

设 G_0 和 G_T 是阶为素数 p 的循环群， g 是 G_0 的生成元， $e: G_0 \times G_0 \rightarrow G_T$ 是双线性映射。

(1) 判定线性 (decisional linear, DL) 假设

随机选取 $f, h, Q \in G_0$ ， $r_1, r_2 \in \mathbb{Z}_p$ ，对于任意概率多项式时间 (probabilistic polynomial-time, PPT) 敌手 $\mathcal{A}$ ，定义其区分以下两个分布的优势为：

$$\text{Adv}_{\mathcal{A}}^{\text{DL}} = |\Pr[\mathcal{A}(gf, hf^{r_1}, g^{r_2}, hf^{r_1+r_2}) = 1] - \Pr[\mathcal{A}(gf, hf^{r_1}, g^{r_2}, Q) = 1]| \quad (1)$$

(2) 判定双线性 Diffie-Hellman (decisional bilinear Diffie-Hellman, DBDH) 假设

随机选取 $a, b, c, z \in \mathbb{Z}_p^*$ ，对于任意 PPT 敌手 $\mathcal{A}$ ，定义其区分以下两个分布的优势为：

$$\text{Adv}_{\mathcal{A}}^{\text{DBDH}} = |\Pr[\mathcal{A}(g^a, g^b, g^c, e(g, g)^{abc}) = 1] - \Pr[\mathcal{A}(g^a, g^b, g^c, e(g, g)^z) = 1]| \quad (2)$$

4 系统模型、形式化定义及安全模型

4.1 系统模型

如图2所示，Pun-ABSE-BM 系统包含4个实体：VCS、KGC、DO和数据用户 (data user, DU)。

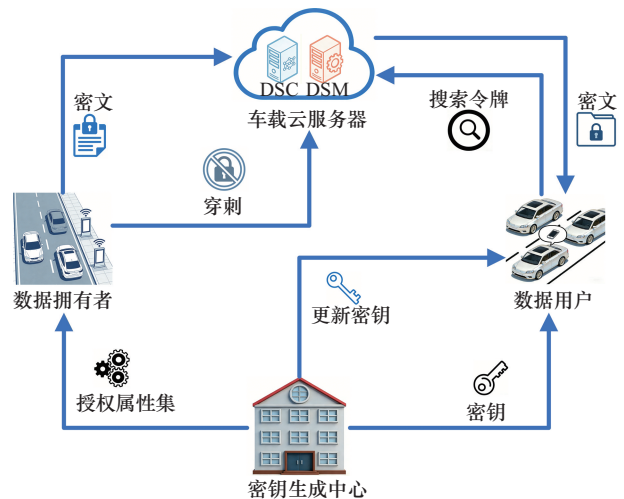


图2 Pun-ABSE-BM 系统模型

(1) KGC生成系统中的公钥与用户私钥,为DO授权属性集。具体地,为DU生成包含访问策略和属性的私钥,允许合法DU申请更新密钥中的访问策略。给DO授权属性集并为其签名。

(2) VCS由数据服务中心(data service center, DSC)和数据服务管理器(data service manager, DSM)组成,负责密文的存储与管理。DSC存储DO上传的密文,并响应DU的搜索请求。对于DU的搜索请求,首先执行关键词搜索,然后对匹配的密文进行外包解密,最后将结果返回给DU。DSM负责处理来自DO的限制策略,对指定密文执行穿刺操作,并对穿刺结果进行签名,发送给DO。

(3) DO制定数据访问和限制策略。用访问策略加密敏感数据,为数据选择一个关键词并使用属性集加密该关键词。当需要撤销用户权限时,制定限制策略并发送给DSM。

(4) DU可发起关键词搜索。若用户密钥中的访问策略与关键词的属性相匹配,且用户属性满足密文访问策略,且其标签未触发限制策略,则用户可获取密文并解密。DU可向KGC申请更新其私钥中的访问策略。

4.2 形式化定义

各算法的形式化定义如下。

$\text{Setup}(1^\lambda, d) \rightarrow (\text{PK}, \text{MSK})$: 以安全参数 λ 和用户最大标签数量 d 为输入。KGC生成公私钥对,同时DSC和DSM也生成自己的公私钥对,输出公私钥对 $(\text{PK}_K, \text{MSK}_K)$ 、 $(\text{PK}_M, \text{MSK}_M)$ 、 $(\text{PK}_S, \text{MSK}_S)$ 。

$\text{KeyGen}(\mathcal{T}_u, \text{Tag}_u, \text{Atts}_u) \rightarrow \text{SK}_u$: 该算法由KGC执行。输入访问策略 $\mathcal{T}_u$ 、属性集 Atts_u 和用户标签集 $\text{Tag}_u = \{t_1, t_2, \dots, t_d\}$ 。为用户生成标识符 UID 并在本地维护一个用户列表,分别为用户生成用于关键词搜索的密钥和用于外包解密的转换密钥,将它们组合为 SK_u ,输出用户私钥 SK_u 。

$\text{Authorize}(\text{Atts}_o, \text{ID}_{\text{DO}}) \rightarrow \sigma_{\text{DO}}$: 该算法由KGC执行。为标识符 ID_{DO} 的DO授权其属性集 Atts_o ,并为其属性集签名,输出签名 σ_{DO} 。

$\text{Encrypt}(M, w, \mathcal{T}_o, \text{Atts}_o) \rightarrow \text{CT}$: 该算法由DO执行。输入明文 M 、关键词 w 、访问策略 $\mathcal{T}_o$ 和属性集 Atts_o 。用 Atts_o 对 w 进行加密得到关键词密文;用 $\mathcal{T}_o$ 对 M 进行加密,并为其生成穿刺组件PSK。得到的密文组合为CT,本地保留PSK,输出CT。

$\text{Puncture}(\mathcal{RP}, \text{PSK}, \text{CT}) \rightarrow \text{CT}'$: 该算法由DSM执行。收到限制策略 $\mathcal{RP}$ 和穿刺组件PSK后,对CT进行穿刺。对穿刺结果 PSK' 进行签名,将签名 ξ 和 PSK' 发送给DO进行验证,输出 CT' 。穿刺后的密文 CT' 无法被含有满足 $\mathcal{RP}$ 标签的用户解密。

$\text{Verify}(\text{PSK}', \xi)$: 在该算法中,DO对DSM发来的穿刺结果 PSK' 和签名 ξ 进行验证。

$\text{TokenGen}(\text{SK}_u, w) \rightarrow \text{TK}$: 该算法由DU执行。选择关键词生成令牌TK并发送给DSC。

$\text{Search}(\text{TK}, \text{CT}) \rightarrow \text{Srchrst}$: 该算法由DSC执行。根据令牌TK对密文进行搜索,若关键词匹配且关键词的属性满足用户的访问策略,则输出搜索结果 Srchrst 。

$\text{DecryptOut}(\text{TK}, \text{Srchrst}) \rightarrow \text{Rst}$: 该算法由DSC执行。若用户的标签集未命中限制策略,且用户属性满足密文的访问策略,则对关键词搜索结果 Srchrst 进行外包解密,输出结果 Rst 。

$\text{Decrypt}(\text{Rst}, \text{SK}_u) \rightarrow M$: 该算法由DU执行。DU使用私钥 SK_u 解密 Rst 得到明文 M 。

$\text{Update}(\text{Upt}, \mathcal{T}_u') \rightarrow \text{SK}_u'$: 该算法由KGC执行。用户向KGC申请更新密钥 SK_u 中的访问策略,验证用户身份后为其重新生成密钥 SK_u' 。

4.3 安全模型

在该系统中,VCS为半诚实实体,它们会忠实执行其他实体授予的权限,试图从加密数据中获取额外信息。KGC和DO被视为完全可信实体,将诚实执行各项任务。DU被视为半诚实实体,其中部分未授权用户可能试图通过合谋攻击访问数据。

通过以下3个安全游戏来形式化目标安全属性: Game-I的目标是IND-sCKA; Game-II的目标是抵抗关键词猜测攻击的关键词保密性; Game-III的目标是IND-sCPA。

(1) Game-I

初始化: 敌手 $\mathcal{A}$ 声明挑战属性集 Atts^* 。

设置阶段: 挑战者 $\mathcal{B}$ 构造公钥,并将其发送给敌手 $\mathcal{A}$ 。

查询阶段1: $\mathcal{A}$ 可以自适应地执行以下两个查询,挑战者维护一个初始为空的关键词列表 L_{kw} 。

$\mathcal{O}_{\text{KeyGen}}(\mathcal{T}^*)$: $\mathcal{A}$ 向 $\mathcal{B}$ 查询访问策略 $\mathcal{T}^*$ 的密钥 $\text{SK}_{\mathcal{A}}$,若 $F(\text{Atts}^*, \mathcal{T}^*) = 1$ 则拒绝。

$\mathcal{O}_{\text{TokenGen}}(\mathcal{T}^*, w)$: $\mathcal{B}$ 执行 $\mathcal{O}_{\text{KeyGen}}(\mathcal{T}^*)$ 获得密钥后,执行TokenGen生成令牌并返回给 $\mathcal{A}$ 。若

$F(\text{Atts}^*, \mathcal{T}^*) = 1$, $\mathcal{B}$ 将 w 添加至关键词列表 L_{kw} 中。

挑战阶段: $\mathcal{A}$ 提交两个等长的关键词 w_0 和 w_1 , 要求 $w_0, w_1 \notin L_{\text{kw}}$ 。 $\mathcal{B}$ 随机选择 $\ell \leftarrow \{0, 1\}$ 构造 w_ℓ 的密文, 输出挑战密文并交付给 $\mathcal{A}$ 。

查询阶段 2: $\mathcal{A}$ 可继续查询 $\mathcal{O}_{\text{KeyGen}}$ 和 $\mathcal{O}_{\text{TokenGen}}$, 但不能查询满足 $F(\text{Atts}^*, \mathcal{T}^*) = 1$ 的 w_0 或 w_1 的令牌。 $\mathcal{B}$ 的回答方式与查询阶段 1 相同。

猜测阶段: $\mathcal{A}$ 输出猜测 $\ell' \in \{0, 1\}$, 如果 $\ell' = \ell$, 则 $\mathcal{A}$ 赢得游戏。在此游戏中, $\mathcal{A}$ 的优势定义为:

$$\text{Adv}_{\mathcal{A}} = \left| \Pr[\ell' = \ell] - \frac{1}{2} \right| \quad (3)$$

定义 1 若所有 PPT 敌手 $\mathcal{A}$ 在安全游戏 Game-I 中仅能以可忽略的优势获胜, 则 Pun-ABSE-BM 方案具有 IND-sCKA 安全性。

(2) Game-II

设置阶段: 挑战者 $\mathcal{B}$ 构造公钥, 并将其发送给敌手 $\mathcal{A}$ 。

查询阶段: $\mathcal{A}$ 对以下预言机进行多次查询。

$\mathcal{O}_{\text{KeyGen}}(\mathcal{T})$: $\mathcal{A}$ 向 $\mathcal{B}$ 发送 $\mathcal{T}^*$, $\mathcal{B}$ 为其构造密钥并返回, 同时将 $\mathcal{T}$ 添加至初始为空的 L_{KeyGen} 列表中。

$\mathcal{O}_{\text{TokenGen}}(\mathcal{T}, w)$: $\mathcal{B}$ 执行 $\mathcal{O}_{\text{KeyGen}}(\mathcal{T})$ 获取密钥, 再执行 TokenGen 算法生成令牌 TK 并返回给 $\mathcal{A}$ 。

挑战阶段: $\mathcal{A}$ 选择属性集 Atts^* 发送给 $\mathcal{B}$ 。 $\mathcal{B}$ 从关键词空间中均匀随机选择 w^* , 并为 w^* 生成密文和令牌。要求对于 $\forall \mathcal{T} \in L_{\text{KeyGen}}$, $F(\text{Atts}^*, \mathcal{T}) = 0$ 。

猜测阶段: 发起 q 次不同的关键词查询后, $\mathcal{A}$ 输出关键词 w' , 若 $w' = w^*$ 则获胜。 $\mathcal{A}$ 获胜的概率为 $\frac{1}{|W| - q} + \varepsilon$, 其中, $|W|$ 表示关键词空间的大小, ε 表示可忽略优势。

定义 2 若所有 PPT 敌手 $\mathcal{A}$ 在安全游戏 Game-II 中仅能以可忽略的优势获胜, 则 Pun-ABSE-BM 方案具有关键词保密性。

(3) Game-III

初始化: 敌手 $\mathcal{A}$ 声明挑战策略 $\mathcal{T}^*$ 和限制策略 $\mathcal{RP}^*$ 。

设置阶段: 挑战者 $\mathcal{B}$ 构造公钥, 并将其发送给 $\mathcal{A}$ 。

查询阶段 1: $\mathcal{A}$ 查询标签集 $\text{Tag} = \{t_1, \dots, t_d\}$ 和属性集 Atts^* 的密钥, 标签集不满足限制策略 $\mathcal{RP}^*$, 且 Atts^* 不满足挑战策略 $\mathcal{T}^*$, $\mathcal{B}$ 为其构造私钥并发

送给 $\mathcal{A}$ 。

挑战阶段: 敌手 $\mathcal{A}$ 随机选择两个等长信息 M_0 和 M_1 发送给 $\mathcal{B}$ 。 $\mathcal{B}$ 选择一个随机数 $\ell \leftarrow \{0, 1\}$, 构造 M_ℓ 的密文, 交付给 $\mathcal{A}$ 挑战密文。

查询阶段 2: 该过程与查询阶段 1 相同。

猜测阶段: $\mathcal{A}$ 输出对 ℓ 的猜测 ℓ' , 若 $\ell' = \ell$, 则 $\mathcal{A}$ 赢得游戏。在此游戏中, $\mathcal{A}$ 的优势定义为:

$$\text{Adv}_{\mathcal{A}} = \left| \Pr[\ell' = \ell] - \frac{1}{2} \right| \quad (4)$$

定义 3 如果所有 PPT 敌手 $\mathcal{A}$ 在 Game-III 中的优势均可忽略不计, 则 Pun-ABSE-BM 方案在 IND-sCPA 条件下是安全的。

除上述不可区分性安全游戏外, 本文方案还需满足以下关键安全属性。

(1) 抗合谋性。防止多个用户通过组合各自属性合谋访问其单独无法解密的未授权数据。

(2) 数据机密性。KGC 虽为诚实实体, 但必须防止其获取明文; DSC 与 DU 被视为半诚实实体, 需确保上传至 DSC 的密文不会被未授权用户或好奇但诚实的 DSM 非法访问。

(3) 可验证性。由于 DSM 是半可信实体, 其执行穿刺操作时可能伪造限制策略或忽略 DO 的请求, 因此必须能够验证穿刺过程的正确性。

(4) 抗混合与匹配攻击。考虑 VSN 部署在开放的无线通信环境中, 系统面临来自公共信道的恶意敌手威胁。敌手通过监听、截获网络信道中的合法数据, 试图将来自不同会话、不同实体或不同时间戳的密文组件、密钥组件或令牌进行恶意剥离与重组。需要抵抗敌手可能存在的 3 种攻击。

① 密文替换: 将某文件合法的关键词密文(或数据密文)恶意替换为其他会话中的组件。

② 密钥替换: 拆分不同用户的私钥, 如使用 User_A 私钥中的搜索密钥(或转换密钥)替换 User_B 私钥中的搜索密钥(或转换密钥)。

③ 令牌替换: 盗用合法用户的令牌, 并将转换密钥替换为敌手自身的转换密钥, 企图规避系统的身份认证与底层约束, 从而非法窃取搜索意图或获取明文数据。

4.4 系统适用边界与 KGC 单点风险

本文在设计时采用集中式 KGC 架构与小属性域构造, 这决定了系统的适用边界与潜在风险。

集中式 KGC 架构掌握着系统主密钥并负责所

有节点的私钥生成,这在降低系统部署复杂度的同时,也存在3类固有缺陷^[17-18,38]:①性能瓶颈风险,在高并发大规模部署场景中,KGC的密钥生成能力将率先达到系统吞吐量上限,成为制约系统整体性能的核心因素;②单点故障风险,KGC的任何故障或停机都可能导致整个系统宕机;③密钥托管风险,KGC持有所有用户的私钥生成材料,一旦被恶意攻破,将引发系统性的大规模数据泄露。

同时,小属性域构造进一步限制了方案在属性种类动态变化的开放场景中的适用性。因此,本文方案更适用于中小规模的区域级车载社交群组 and 节点规模可控的VSN细分场景。

5 具体构造

本节给出了一个Pun-ABSE-BM方案构造实例,具体的11个算法描述如下。

(1) $\text{Setup}(1^\lambda, d) \rightarrow (\text{PK}, \text{MSK})$: 输入安全参数 λ 和用户最大标签数量 d 。

KGC选取一个双线性映射 $e: G_0 \times G_0 \rightarrow G_T$, G_0 和 G_T 是阶为 p 的循环群, p 为 λ 比特素数。选择哈希函数 $H_1: \{0,1\}^* \rightarrow G_0$, $H_2: \{0,1\}^* \rightarrow \mathbb{Z}_p^*$, $H_3: \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^\lambda$ 。选择一个安全的对称加密算法 $\Pi = (\text{SEnc}, \text{SDec})$, 令其密钥长度为 k , 选择一个抗碰撞的哈希函数 $H_4: G_T \times \{0,1\}^* \rightarrow \{0,1\}^k$ 。选择 $a, b, c, \alpha, \beta, \gamma \in \mathbb{Z}_p^*$, 计算 $g^a, g^b, g^c, g^\alpha, g^\beta, e(g, g)^\gamma$ 。KGC选择一个多项式 $q(x)$, 令 $q(0) = \beta$, 对于 $i \in [1, d]$, 计算 $Q(t_i) = g^{q(t_i)}$, 并设 t_0 为正常操作期间不使用的特殊标记。对于系统中的所有属性 $U = \{1, 2, \dots, n\}$, 选择随机指数 $\tau_j \in \mathbb{Z}_p^*$, $j \in U$, 计算 $h_j = g^{\tau_j}$ 。随机选择 $\text{sk}_{\text{sig}} \in \mathbb{Z}_p^*$, 计算 $\text{pk}_{\text{sig}} = g^{\text{sk}_{\text{sig}}}$ 。KGC保留私钥 $\text{MSK}_K = (a, b, c, \alpha, \gamma, \text{sk}_{\text{sig}})$, 输出公钥 $\text{PK}_K = (e, g, g^a, g^b, g^c, g^\alpha, g^\beta, e(g, g)^\gamma, \{h_j\}_{j \in [1, |U|]}, H_1, H_2, H_3, H_4, \{Q(t_i)\}_{i \in [1, d]}, t_0, \text{pk}_{\text{sig}}, \text{SEnc}, \text{SDec})$ 。

DSM随机选择 $\varphi \in \mathbb{Z}_p^*$, 保留 $\text{MSK}_M = \varphi$, 输出 $\text{PK}_M = g^\varphi$ 。DSC随机选择 $\psi \in \mathbb{Z}_p^*$, 计算 $\text{MSK}_S = H_1(\text{ID}_S)^\psi$, 其中 ID_S 是DSC的标识, 输出 $\text{PK}_S = g^\psi$ 。

以下算法均将系统公钥作为输入。

(2) $\text{KeyGen}(\mathcal{T}_u, \text{Tag}_u, \text{Atts}_u) \rightarrow \text{SK}_u$: 该算法由KGC执行, 输入访问策略 $\mathcal{T}_u$ 、用户标签集 $\text{Tag}_u = \{t_1, t_2, \dots, t_d\}$ 及用户属性集 Atts_u 。

随机选取 $r, z \leftarrow \mathbb{Z}_p^*$, 计算 $\text{CP}_1 = (g^r g^{ra + \beta a})^{\frac{1}{z}}$, $\text{CP}_2 = g^{\frac{1}{z}}$ 。对于 $j \in \text{Atts}_u$, 计算 $\text{CP}_{3j} = (g^{ar} h_j^r)^{\frac{1}{z}}$; 对于标签 $t_i \in \mathcal{T}_u$, 计算 $P_i = Q(t_i)^a$ 。生成转换密钥 $\text{SK}_1 = (\text{Atts}_u, \text{CP}_1, \text{CP}_2, \{\text{CP}_{3j}\}_{j \in \text{Atts}_u}, \text{Tag}_u, \{P_i\}_{t_i \in \text{Tag}_u})$ 。

执行算法 $\text{Share}(\mathcal{T}_u, ac)$, 计算每个叶子节点的秘密份额 $q_v(0)$, v 为叶子节点属性。对于每个 $v \in \mathcal{T}_u$, 选择随机数 $\varpi_v \in \mathbb{Z}_p^*$, 计算 $\text{KP}_{1,v} = g^{\frac{q_v(0)}{z}} h_v^{\varpi_v}$, $\text{KP}_{2,v} = g^{\varpi_v}$ 。搜索密钥 $\text{SK}_2 = (\mathcal{T}_u, \{\text{KP}_{1,v}, \text{KP}_{2,v}\}_{v \in \mathcal{T}_u})$ 。

为DU生成唯一标识符 $\mathcal{UID}$, 计算 $vk = g^z$, 本地保存一个用户表 $[H_2(\mathcal{UID}), (\mathcal{UID}, vk)]$ 。输出用户私钥 $\text{SK}_u = (z, \mathcal{UID}, \text{SK}_1, \text{SK}_2)$ 。

(3) $\text{Authorize}(\text{Atts}_o, \mathcal{ID}_{\text{DO}}) \rightarrow \sigma_{\text{DO}}$: 该算法由KGC执行, 输入属性集 Atts_o 和DO身份标识 $\mathcal{ID}_{\text{DO}}$, 为DO授权属性集 Atts_o , 并将其规范排序后拼接为 $\text{string}(\text{Atts}_o)$, 使用 sk_{sig} 对其签名 $\sigma_{\text{DO}} = H_2(\mathcal{ID}_{\text{DO}} \parallel \text{string}(\text{Atts}_o))^{\text{sk}_{\text{sig}}}$ 。将 $(\mathcal{ID}_{\text{DO}}, \text{Atts}_o, \sigma_{\text{DO}})$ 发送给DO。

(4) $\text{Encrypt}(M, w, \mathcal{T}_o, \text{Atts}_o) \rightarrow \text{CT}$: 该算法由DO执行, 输入消息 M 、关键词 w 、访问策略 $\mathcal{T}_o$ 及属性集 Atts_o 。

随机选择 $r_1, r_2 \in \mathbb{Z}_p^*$, 计算 $W = g^{a(r_1 + r_2)} g^{bH_2(w)r_1}$, $W' = g^{cr_1}$, $W_0 = g^{r_2}$ 。选择属性集 $\mathbb{A} \in \text{Atts}_o$, 对于属性 $j \in \mathbb{A}$, 计算 $W_j = h_j^{r_2}$ 。生成关键词密文 $\text{ckw} = (\mathbb{A}, W', W, W_0, \{W_j\}_{j \in \mathbb{A}}, \mathcal{ID}_{\text{DO}}, \sigma_{\text{DO}}, \text{Atts}_o)$ 。

首先, 随机选择 $\text{key} \leftarrow G_T$, 将 ckw 与 key 结合得到对称密钥 $\text{EK} = H_4(\text{key}, \text{ckw})$, 用于加密消息 M , 执行对称加密算法得到密文 $C_M = \text{SEnc}_{\text{EK}}(M)$ 。

接着, 加密对称密钥中的 key 。随机选取 $\rho \leftarrow \mathbb{Z}_p^*$, 计算 g^ρ 和DO与DSC的会话密钥 $K_S = e((g^\rho)^\rho, H_1(\text{ID}_S))$ 。随机选择 $s \leftarrow \mathbb{Z}_p^*$, 计算 $\tilde{C} = \text{key}_{K_S} e(g, g)^s$, $C_1 = g^{as}$, $C_2 = g^{\beta s}$, $C_3 = g^s$ 。执行 $\text{Share}(\mathcal{T}_o, s)$ 算法, 计算每个叶子节点的秘密份额 $q_v(0)$, 设叶子节点属性为 v 。对于每个 $v \in \mathcal{T}_o$, 计算 $C_{4,v} = h_v^{q_v(0)}$, $C_{5,v} = g^{q_v(0)}$ 。选择随机数 $r_0 \leftarrow \mathbb{Z}_p^*$, 计算 $K_0 = g^{a(r_0 + 1)}$, $L_0 = g^{sr_0}$, $Q_0 = Q(t_0)^{r_0}$, 初始穿刺组件 $\text{PSK} = \{ \text{psk}_0 = (t_0, K_0, L_0, Q_0) \}$, 得到消息密文 $\text{cph} = (\text{PSK}, \mathcal{T}_o, \tilde{C}, C_1, C_2, C_3, \{C_{4,v}, C_{5,v}\}_{v \in \mathcal{T}_o}, g^\rho)$ 。

将密文 $\text{CT} = (C_M, \text{ckw}, \text{cph})$ 发送给DSC, 并本

地保存初始穿刺组件 PSK。

(5) $\text{Puncture}(\mathcal{RP}, \text{PSK}, \text{CT}) \rightarrow \text{CT}'$: 该算法由 DSM 执行, 输入限制策略 $\mathcal{RP}$ 、初始穿刺组件 PSK 和密文 CT。

DO 本地保存 CT 的历史 $\mathcal{RP}$ 和初始穿刺组件, 当每次需要新增撤销用户标签时, 将其新增至原 $\mathcal{RP}$ 。因此, DO 发送的 $\mathcal{RP}$ 为总限制策略, 同时附带对应的 PSK。DSM 收到 $\mathcal{RP}$ 和 PSK 后, 首先通过德摩根定律将 $\mathcal{RP}$ 转换为正向策略 $\mathcal{AP}$, 再用 $\mathcal{AP}$ 对 PSK 进行穿刺, 具体如下。

随机选择 $\sigma, \eta \in \mathbb{Z}_p^*$, 计算 $K'_0 = K_0 g^{a(\eta - \sigma)}$, $L'_0 = L_0 g^{s\eta}$, $Q'_0 = Q_0 Q(t_0)^\eta$, $\text{psk}'_0 = (t_0, K'_0, L'_0, Q'_0)$ 。执行 $\text{Share}(\mathcal{AP}, \sigma)$ 算法, 获得每个叶子节点 $-t_j \in \mathcal{AP}$ 的份额 σ_j 。对于叶子节点 $-t_j \in \mathcal{AP}$, 选择 $\eta_j \leftarrow \mathbb{Z}_p^*$, 计算 $K_j = g^{a(\sigma_j + \eta_j)}$, $L_j = g^{s\eta_j}$, $Q_j = Q(t_j)^{\eta_j}$, $\text{psk}_j = (t_j, K_j, L_j, Q_j)$ 。更新穿刺组件 $\text{PSK}' = (\mathcal{RP}, \mathcal{AP}, \text{psk}'_0, \text{psk}_1)$, 对穿刺结果进行签名 $\zeta = g^{\frac{1}{\sigma + H_2(\text{PSK})}}$, 新密文 $\text{cph}' = (\text{PSK}', T_o, \tilde{C}, C_1, C_2, C_3, \{C_{4,v}, C_{5,v}\}_{v \in T_o}, g^{\rho})$, 输出 $\text{CT}' = (C_M, \text{ckw}, \text{cph}')$, 发送签名 (PSK', ζ) 给 DO 进行验证。

(6) $\text{Verify}(\text{PSK}', \zeta)$: 该算法由 DO 执行, 验证穿刺结果和签名, $e(\text{PK}_M g^{H_2(\text{PSK}')} \zeta) \stackrel{?}{=} e(g, g)$ 。若验证失败, 则拒绝本次穿刺。

(7) $\text{TokenGen}(\text{SK}_u, w) \rightarrow \text{TK}$: 该算法由 DU 执行, 输入私钥 SK_u 和关键词 w 。选择随机数 $\mu \leftarrow \mathbb{Z}_p$, 计算 $\text{KP}'_{1,v} = (\text{KP}_{1,v})^\mu$, $\text{KP}'_{2,v} = (\text{KP}_{2,v})^\mu$, $\text{tok}_1 = (g^a g^{bH_2(w)})^{\frac{\mu}{z}}$, $\text{tok}_2 = g^{\frac{c\mu}{z}}$ 。获取时间戳 TS, 生成动态签名 $\sigma_{\text{DU}} = H_1(\text{tok}_1 \| \text{tok}_2 \| \text{SK}_1 \| \text{TS})^\tau$, $\text{id} = H_2(\text{UID})$ 。将以上组件与 SK_1 组合为令牌 $\text{TK} = (\text{id}, \sigma_{\text{DU}}, T_u, \text{tok}_1, \text{tok}_2, \{\text{KP}'_{1,v}, \text{KP}'_{2,v}\}_{v \in T_u}, \text{SK}_1)$ 。

(8) $\text{Search}(\text{TK}, \text{CT}) \rightarrow \text{Srchrst}$: 该算法由 DSC 执行, 输入令牌 TK 和密文 CT。

在搜索执行前, 先验证密文属性是否合法, 获取密文中关于属性集的签名 $(\text{ID}_{\text{DO}}, \text{Atts}_o, \sigma_{\text{DO}})$, 规范排序后拼接 $\text{string}(\text{Atts}_o)$, 计算 $H_2(\text{ID}_{\text{DO}} \| \text{string}(\text{Atts}_o))$, 验证 $e(\sigma_{\text{DO}}, g) \stackrel{?}{=} e(H_2(\text{ID}_{\text{DO}} \| \text{string}(\text{Atts}_o)), \text{pk}_{\text{sig}})$, 随后验证属性范围 $\subseteq \text{Atts}_o \stackrel{?}{=} \text{True}$, 若验证失败, 则返回 $\perp$, 并视密文 CT 为无效密文。

随后验证令牌是否有效, 根据令牌中的 id, 从用户表中查询得到 vk , 验证 $e(\sigma_{\text{DU}}, g) \stackrel{?}{=} e(H_1(\text{tok}_1 \| \text{tok}_2 \| \text{SK}_1 \| \text{TS}), vk)$, 若验证失败, 则说明令牌

无效, 拒绝请求。

若以上验证通过, 则计算 $\text{Left} = e(W', \text{tok}_1) E_{\text{root}}$, $\text{Right} = e(W, \text{tok}_2)$, 判断 $\text{Left} \stackrel{?}{=} \text{Right}$ 。若 $\mathbb{A}$ 满足 T_u , 则有 $v = j$, $v \in T_u$, $j \in \mathbb{A}$, 设满足 T_u 的属性集合为 $S \subseteq \mathbb{A}$, 计算:

$$E_v = \frac{e(\text{KP}'_{1,v}, W_0)}{e(\text{KP}'_{2,v}, W_j)} = \frac{e(g^{\frac{\mu q_v(0)}{z}} h_v^{\mu \sigma_v}, g^{r_2})}{e(g^{\mu \sigma_v}, h_j^{r_2})} = \frac{e(g^{\frac{\mu q_v(0)}{z}} g^{r_2}) e(h_v^{\mu \sigma_v}, g^{r_2})}{e(g^{\mu \sigma_v}, h_j^{r_2})} = e(g, g)^{\frac{q_v(0) \mu r_2}{z}} \quad (5)$$

$$\text{调用 } e(g, g)^{\frac{q_{\text{root}}(0) \mu r_2}{z}} \leftarrow \text{Combine}(T_u, \{E_v | v \in S\}),$$

因为 $q_{\text{root}}(0) = ac$, 所以 $E_{\text{root}} = e(g, g)^{\frac{ac \mu r_2}{z}}$, 则有:

$$\begin{aligned} \text{Left} &= e(W', \text{tok}_1) E_{\text{root}} = \\ &= e(g^{cr_1}, g^{\frac{(a+bH_2(w))\mu}{z}}) e(g, g)^{\frac{ac \mu r_2}{z}} = \\ &= e(g, g)^{\frac{cr_1 \mu (a+bH_2(w)) + ac \mu r_2}{z}} \end{aligned} \quad (6)$$

$$\begin{aligned} \text{Right} &= e(W, \text{tok}_2) = e(g^{a(r_1+r_2)+bH_2(w)r_1}, g^{\frac{c\mu}{z}}) = \\ &= e(g, g)^{\frac{cr_1 \mu (a+bH_2(w)) + c \mu ar_2}{z}} \end{aligned} \quad (7)$$

由于 $\text{Left} = \text{Right}$, 故关键词匹配且 DO 属性满足 DU 访问策略, 将密文加入结果集合 $\text{CT} \rightarrow \text{Srchrst}$ 。

(9) $\text{DecryptOut}(\text{TK}, \text{Srchrst}) \rightarrow \text{Rst}$: 该算法由 DSC 执行, 输入令牌 TK 和关键词搜索结果 Srchrst 。获取密文中的限制策略 $\mathcal{RP}$, 若判断用户标签 Tag_u 未被撤销, 则继续执行算法。

计算一组拉格朗日系数 $\{\pi_1, \dots, \pi_d, \pi^*\}$, 令其满足 $\sum_{m=1}^d [\pi_m q(t_m)] + \pi^* q(t_0) = q(0) = \beta$, 计算:

$$\begin{aligned} Z_0 &= \frac{e(K'_0, C_2)}{e(L'_0, \prod_{m=1}^d (P_m)^{\pi_m}) e(Q'_0, C_1)^{\pi^*}} = \\ &= \frac{e(g, g)^{\alpha \beta s(r_0+1+\eta-\sigma)}}{e(g, g)^{\alpha s(r_0+\eta) \sum_{m=1}^d \pi_m q(t_m)} e(g, g)^{\alpha s(r_0+\eta) \pi^* q(t_0)}} = \\ &= \frac{e(g, g)^{\alpha \beta s(r_0+1+\eta-\sigma)}}{e(g, g)^{\alpha s(r_0+\eta) (\sum_{m=1}^d \pi_m q(t_m) + \pi^* q(t_0))}} = e(g, g)^{\alpha \beta s(1-\sigma)} \end{aligned} \quad (8)$$

对于 $-t_j \in \mathcal{AP}$, 计算拉格朗日系数 $\{\delta_1, \dots, \delta_d, \delta^*\}$,

令其满足 $\sum_{m=1}^d [\delta_m q(t_m)] + \delta^* q(t_j) = q(0) = \beta$, 计算:

$$\begin{aligned} Z_{1,j} &= \frac{e(K_j, C_2)}{e(L_j, \prod_{m=1}^d (P_m)^{\delta_m}) e(Q_j, C_1)^{\delta^*}} \\ &= \frac{e(g, g)^{a\beta s(\sigma_j + \eta_j)}}{e(g, g)^{a\sigma \eta_j \sum_{m=1}^d \delta_m q(t_m)} e(g, g)^{a\sigma \eta_j \cdot \delta^* q(t_0)}} \\ &= \frac{e(g, g)^{a\beta s(\sigma_j + \eta_j)}}{e(g, g)^{a\sigma \eta_j (\sum_{m=1}^d \delta_m q(t_m) + \delta^* q(t_0))}} = e(g, g)^{a\beta \sigma \sigma_j} \quad (9) \end{aligned}$$

通过叶子节点值 $Z_{1,j}$ 恢复限制策略 $\mathcal{AP}$ 根节点 Z_1 的值 $e(g, g)^{a\beta \sigma} \leftarrow \text{Combine}(\mathcal{AP}, \{Z_{1,j} | t_j \in \mathcal{AP}\})$, 得到 $Z_1 = e(g, g)^{a\beta \sigma}$, 随后计算 $B = Z_0 Z_1 = e(g, g)^{a\beta s}$ 。若 Atts_u 满足访问树 T_o , 则有 $v=j$, $v \in T_o$, $j \in \text{Atts}_u$, 设满足 T_o 的属性集合为 S , 计算:

$$\begin{aligned} R_v &= \frac{e(C_{5,v}, \text{CP}_{3,j})}{e(C_{4,v}, \text{CP}_2)} = \frac{e(g^{q_v(0)}, g^{\frac{ra}{z}} h_j^{\frac{r}{z}})}{e(h_v^{q_v(0)}, g^{\frac{r}{z}})} \\ &= \frac{e(g, g)^{\frac{raq_v(0)}{z}} e(g, h_j)^{\frac{q_v(0)r}{z}}}{e(h_v^{q_v(0)r})} = e(g, g)^{\frac{raq_v(0)}{z}} \quad (10) \end{aligned}$$

调用算法 $e(g, g)^{\frac{ras}{z}} \leftarrow \text{Combine}(T_o, \{R_v | v \in S\})$,

得到 $R_{\text{root}} = e(g, g)^{\frac{ras}{z}}$, 随后计算:

$$\begin{aligned} A &= \frac{e(C_3, \text{CP}_1)}{R_{\text{root}}} = \frac{e(g^s, (g^{\gamma} g^{ra + \beta\alpha})^{\frac{1}{z}})}{e(g, g)^{\frac{ras}{z}}} \\ &= \frac{e(g, g)^{\frac{s\gamma}{z}} e(g, g)^{\frac{ras}{z}} e(g, g)^{\frac{\beta\alpha s}{z}}}{e(g, g)^{\frac{ras}{z}}} = \\ &= e(g, g)^{\frac{s\gamma}{z}} e(g, g)^{\frac{\beta\alpha s}{z}} \quad (11) \end{aligned}$$

会话密钥 $K_S = e(g^{\rho}, \text{MK}_S) = e(g, H_1(\text{ID}_S))^{\rho\rho}$,

计算 $\tilde{C}' = \frac{\tilde{C}}{K_S} = \text{key} e(g, g)^{\gamma s}$, 将外包解密结果加入集合 $\text{Rst} \leftarrow (\{A, B, \tilde{C}', \text{ckw}, C_M\})$, 返回搜索结果 Rst 。

(10) $\text{Decrypt}(\text{Rst}, \text{SK}_u) \rightarrow M$: 该算法由 DU 执行, 输入搜索结果 Rst 和用户私钥 SK_u 。使用私钥 z 解密得 $\text{key} = \frac{\tilde{C}' B}{A^z} = \frac{\text{key} e(g, g)^{\gamma s} e(g, g)^{a\beta s}}{(e(g, g)^{\frac{s\gamma}{z}} e(g, g)^{\frac{\beta\alpha s}{z}})^z}$, 计

算对称密钥 $\text{EK} = H_4(\text{key}, \text{ckw})$, 解密得到明文 $M = \text{SDec}_{\text{EK}}(C_M)$ 。

(11) $\text{Update}(\text{Upt}, T_u') \rightarrow \text{SK}'_2$: 该算法由 KGC 执行, 输入更新令牌 Upt 和访问策略 T_u' 。

该算法首先由 DU 发起, 选择随机数 $\theta \in \mathbb{Z}_p^*$, 获取时间戳 TS (窗口设置为 3 min), 计算 $\text{id} = H_2(\text{UID})$, $\text{PF} = H_3(\text{UID}, \theta \| \text{TS})$, 向 KGC 发送密钥更新请求 $\text{Upt} = (\text{id}, \text{PF}, \theta, \text{TS}, T_u')$ 。

KGC 收到请求后查询身份索引验证用户身份: $\text{index}(\text{id}) \rightarrow \text{UID}$, 计算 $H_3(\text{UID}, \theta \| \text{TS}) \stackrel{?}{=} \text{PF}$, 若用户身份合法, 则继续执行算法。

执行 $\text{Share}(T_u', ac)$ 算法计算每个叶子节点的秘密份额 $q_v(0)$, v 为叶子节点属性。对于每个 $v \in T_u'$, 选择 $\sigma_v \in \mathbb{Z}_p^*$, 计算 $\text{KP}'_{1,v} = g^{q_v(0)} h_v^{\sigma_v}$, $\text{KP}'_{2,v} = g^{\sigma_v}$, 将 $\text{SK}'_2 = (T_u', \{\text{KP}'_{1,v}, \text{KP}'_{2,v}\}_{v \in T_u'})$ 返回给 DU。

图3展示了双向匹配机制中各实体间的协议交互流向及关键加密逻辑。协议流程及密码学绑定机制包含以下关键步骤。

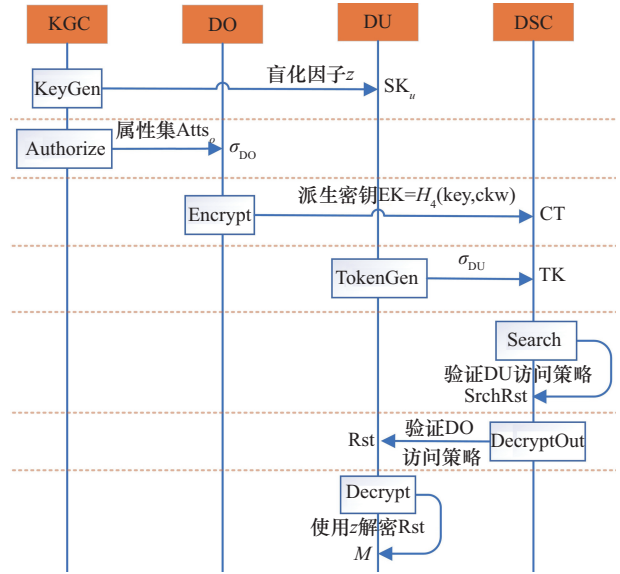


图3 双向匹配流程

在密钥生成算法中, KGC 在用户密钥中嵌入用户专属盲化因子 z , 从底层绑定用户密钥中的 SK_1 与 SK_2 。在授权算法中, KGC 对授权属性集 Atts_o 进行签名 σ_{DO} , 限制 DO 在加密时使用属性集。在加密算法中, DO 采用混合加密机制, 通过对称密钥 $\text{EK} = H_4(\text{key}, \text{ckw})$ 加密消息 M , 通过属性加

密关键词 w 得到密文 ckw , 通过策略加密密钥组件 key 得到 cph , 从而确保密文的不可分割性。在令牌生成算法中, 用户使用其专属盲化因子 z 对令牌组件和 SK_1 签名 $\sigma_{DU} = H_1(\text{tok}_1 || \text{tok}_2 || SK_1 || TS)^z$, 从协议层抵御针对令牌的重放与替换攻击。在 Search 算法中, DSC 首先对 σ_{DO} 和 σ_{DU} 进行验证, 验证通过后进行关键词匹配与 DU 访问策略匹配。在 DecryptOut 算法中, DSC 进行撤销验证与 DO 访问策略匹配, 将盲化因子 z 嵌入外包解密结果。在 Decrypt 算法中, DU 会获得满足双向访问策略的密文, 同时只有密钥和令牌未经恶意替换的用户才能用其盲化因子 z 正确解密得到 key 。最后, DU 结合密文 ckw 派生出对称密钥 EK , 仅当密文未被恶意替换时, DU 才能正确解密得到明文 M 。

考虑到 VSN 中设备资源受限, 引入离线/在线技术可有效降低计算开销。具体地, DO 在离线阶段预计算与随机数相关的中间组件, 在线阶段仅需绑定具体关键词和访问结构即可快速完成加密。DU 在离线阶段预计算令牌生成的中间组件, 在线阶段仅需绑定目标关键词即可快速生成令牌。

(12) Offline.Encrypt($Atts_o$) $\rightarrow$ ICT: 该算法由 DO 执行, 输入属性集 $Atts_o$ 。选取随机数 $r_1, r_2 \leftarrow \mathbb{Z}_p^*$, 计算 $IW_1 = g^{a(r_1 + r_2)}$, $IW_2 = g^{br_1}$, $W' = g^{cr_1}$, $W_0 = g^{r_2}$, 对于属性 $j \in Atts_o$, 计算 $IW_j = h_j^{r_2}$, 获得密文 $Ickw = (Atts_o, IW_1, IW_2, W', W_0, \{IW_j\}_{j \in Atts_o})$ 。

选取 $\rho \leftarrow \mathbb{Z}_p^*$, 计算 g^ρ 会话密钥 $K_S = e((g^w)^\rho, H_1(ID_S))$, 随机选取 $s \leftarrow \mathbb{Z}_p^*$, 计算 $I\tilde{C} = K_S e(g, g)^{rs}$, $C_1 = g^{as}$, $C_2 = g^{bs}$, $C_3 = g^s$ 。选择随机数 $r_0 \leftarrow \mathbb{Z}_p^*$, $PSK = \{psk_0 = (t_0, K_0 = g^{a(r_0+1)}, L_0 = (C_3)^{r_0}, Q_0 = Q(t_0)^{r_0})\}$, 得到密文 $Icph = (s, PSK, I\tilde{C}, C_1, C_2, C_3, g^\rho)$ 。输出中间密文 $ICT = (Ickw, Icph)$ 。

(13) Online.Encrypt(M, w, T_o, ICT) $\rightarrow$ CT: 该算法由 DO 执行, 输入明文 M 、关键词 w 、访问策略 T_o 及中间密文 ICT 。

从 ICT 中获取 W' 和 W_0 , 计算 $W = IW_1 (IW_2)^{H_2(w)} = g^{a(r_1 + r_2)} g^{bH_2(w)r_1}$ 。选择属性集 $\mathbb{A} \in Atts_o$, 对于 $j \in \mathbb{A}$, 从 ICT 中选择 $W_j = IW_j$, 得到关键词密文 $ckw = (\mathbb{A}, W', W, W_0, \{W_j\}_{j \in \mathbb{A}}, \mathcal{ID}_{DO}, \sigma_{DO}, Atts_o)$ 。

随机选择 $key \leftarrow G_T$, 将 ckw 与 key 结合得到对称密钥 $EK = H_4(key, ckw)$, 用于加密消息 M , 执

行对称加密算法得到密文 $C_M = \text{SEnc}_{EK}(M)$ 。

从 ICT 中获取 PSK , C_1 , C_2 , C_3 , 加密 key 值 $\tilde{C} = \text{key} I\tilde{C}$ 。执行 $\text{Share}(T_o, s)$ 算法计算每个叶子节点的秘密份额 $q_v(0)$, v 为叶子节点属性。对于每个 $v \in T_o$, 计算 $C_{4,v} = h_v^{q_v(0)}$, $C_{5,v} = g^{q_v(0)}$, 得到密文 $cph = (PSK, T_o, \tilde{C}, C_1, C_2, C_3, \{C_{4,v}, C_{5,v}\}_{v \in T_o}, g^\rho)$ 。

将密文 $CT = (C_M, ckw, cph)$ 发送给 DSC, 并本地保存初始穿刺组件 PSK 。

(14) Offline.TokenGen(SK_u) $\rightarrow$ ITK: 该算法由 DU 执行, 输入私钥 SK_u 。选择随机数 $\mu \leftarrow \mathbb{Z}_p$, 计算 $KP'_{1,v} = (KP_{1,v})^\mu$, $KP'_{2,v} = (KP_{2,v})^\mu$, $Itok_1 = g^{\frac{b\mu}{z}}$, $Itok_2 = g^{\frac{a\mu}{z}}$, $tok_2 = g^{\frac{c\mu}{z}}$, 生成中间令牌 $ITK = (Itok_1, Itok_2, tok_2, \{KP'_{1,v}, KP'_{2,v}\}_{v \in T_u})$ 。

(15) Online.TokenGen(w, ITK) $\rightarrow$ TK: 该算法由 DU 执行, 输入关键词 w 和中间令牌 ITK 。从 ITK 中获取 tok_2 , $\{KP'_{1,v}, KP'_{2,v}\}_{v \in T_u}$, 计算 $tok_1 = Itok_1 (Itok_2)^{H_2(w)} = g^{\frac{(a + bH_2(w))\mu}{z}}$, 获取时间戳 TS , 签名 $\sigma_{DU} = H_1(tok_1 || tok_2 || SK_1 || TS)^z$, $id = H_2(\mathcal{UID})$ 。将以上组件与转换密钥 SK_1 组合为 $TK = (id, \sigma_{DU}, T_u, tok_1, tok_2, \{KP'_{1,v}, KP'_{2,v}\}_{v \in T_u}, SK_1)$ 。

6 安全性证明

定理 1 基于 DL 假设和单向哈希函数 H_2 , Pun-ABSE-BM 方案在随机预言模型中满足 IND-sCKA。

定理 2 设 H_2 为单向哈希函数, Pun-ABSE-BM 方案满足关键词保密性。

定理 3 基于 DBDH 假设, 且 H_1 建模为随机预言机时, Pun-ABSE-BM 方案满足 IND-sCPA。

限于篇幅, 详细的安全性证明过程见附录 1。

除上述不可区分性安全游戏外, 本文方案还满足以下关键安全属性。

(1) 抗合谋性。未授权用户若想联合恢复根值 $(e(g, g)^{ars})^{\frac{1}{z}}$, 需将密文中的 $\{C_{4,v}, C_{5,v}\}$ 与另一未授权用户密钥中的 $\{CP_{3,j}\}$ 进行配对。然而, 密钥分量 $CP_{3,j} = (g^{ar} h_j^r)^{\frac{1}{z}}$ 被用户唯一且保密的 $\frac{r}{z}$ 混淆, 使攻击者无法通过合谋手段计算根值 $(e(g, g)^{ars})^{\frac{1}{z}}$ 。

(2) 数据保密性。若用户属性无法满足密文中

的访问策略, 则其在 DSC 外包解密过程中无法恢复根值 $(e(g, g)^{ars})^{\frac{1}{z}}$, 因而无法解密。即使授权用户在 DSC 外包解密前获得密文, 也无法直接解密。密文 $\tilde{C} = \text{key}K_S e(g, g)^{rs}$ 中的会话密钥 K_S 仅由 DO 与 DSC 协同生成, 用户无法单独恢复。DSC 虽可使用转换密钥进行外包解密, 但由于用户密钥中的随机盲化因子 z 对 DSC 保密, 故其无法获得最终明文。KGC 即使从 DSC 获得密文, 也无法解密, 因其不知道会话密钥 K_S 。被撤销用户权限失效后, DSC 虽能恢复原始访问策略对应的根值 $(e(g, g)^{ars})^{\frac{1}{z}}$, 但无法计算限制策略的根值 $e(g, g)^{qbs}$, 导致外包解密流程中断。因此, 被撤销用户无法获得明文。

(3) 可验证性。每次对密文执行穿刺后, DSM 使用哈希函数对穿刺结果进行签名, 并将签名结果发送给 DO。DO 验证签名, 以确认密文已按预设策略正确穿刺, 从而确保撤销操作的合法性与准确性。

(4) 抗混合与匹配攻击。本文方案通过为合法用户分配专属的随机盲化因子 z , 并深度耦合至该用户的搜索组件与外包解密组件的底层多项式中。此外, 通过引入混合加密、哈希绑定及动态签名, 增加协议层中的防御。

①抗密文替换: 本文方案在 Encrypt 阶段采用了混合加密架构并引入了关联数据哈希绑定。加密消息 M 的对称密钥 EK 由 key 与关键词密文 ckw 派生而来。若敌手 $\mathcal{A}$ 对密文组合进行混合与匹配, 合法 DU 派生出正确的对称密钥, 导致对称解密算法报错并拒绝输出。

②抗密钥替换: 本文方案在 KeyGen 阶段, 通过在底层结构中嵌入用户专属盲化因子 z 实现了组件间的紧密绑定。若 User_A 试图将其搜索密钥 $\text{SK}_{2,A}$ 替换为 User_B 的搜索密钥 $\text{SK}_{2,B}$ (其中含 z_B), 则在执行 TokenGen 算法时, User_A 生成的 tok_2 包含 z_A 。因 $z_A \neq z_B$, 将直接导致 $\text{Left} \neq \text{Right}$, 搜索匹配失败。若 User_A 试图将其转换密钥 $\text{SK}_{1,A}$ 替换为 User_B 的转换密钥 $\text{SK}_{1,B}$ (其中含 z_B), 则 VCS 执行外包解密后返回的中间密文将被 z_B 盲化。当 User_A 在本地尝试解密时, 因 $z_B \neq z_B$ 无法正确恢复出 key, 导致其无法获取明文。

③抗令牌替换: 本文方案在 TokenGen 算法中

引入签名机制。用户需利用其专属盲化因子 z 计算动态认证标签 $\sigma_{\text{DU}} = H_1(\text{tok}_1 \| \text{tok}_2 \| \text{SK}_1 \| \text{TS})^z$ 。该标签强制将令牌、转换密钥以及当前时间戳进行了密码学意义上的强绑定。VCS 在执行搜索前, 对动态认证标签进行验证, 若敌手篡改或替换了令牌中的任意组件, 则签名验证将直接失败, VCS 会立即阻断并丢弃该伪造请求。

7 性能分析

7.1 理论分析

表 1 将本文方案与功能相近的方案进行了对比。文献[11,17-19]支持关键词搜索; 文献[15-16]支持双向匹配; 文献[17-19]支持用户撤销, 但均采用直接撤销, 需要预先设定用户撤销列表; 本文方案不需要预先设定用户撤销列表。在访问结构方面, 本文方案采用树形访问结构, 文献[11]仅支持 AND 门, 其余方案均采用 LSSS。此外, 本文方案同时具备关键词搜索、双向匹配、用户撤销、撤销验证及外包解密功能。

表 1 各方案功能对比

方案	关键词搜索	双向匹配	用户撤销	撤销验证	撤销类型	外包解密	访问结构
文献[11]	✓	×	×	×	—	×	AND 门
文献[15]	×	✓	×	×	—	×	线性秘密共享方案
文献[16]	×	✓	×	×	—	×	线性秘密共享方案
文献[17]	✓	×	✓	×	直接	×	线性秘密共享方案
文献[18]	✓	×	✓	×	直接	×	线性秘密共享方案
文献[19]	✓	×	✓	×	直接	×	线性秘密共享方案
本文方案	✓	✓	✓	✓	间接	✓	访问树

表 2 从密钥生成、加密、令牌生成、搜索和解密 5 个维度比较了各方案的计算开销, 相关符号说明如表 3 所示。文献[11]的整体计算开销最低, 该方案实现了计算开销与属性数和关键词数的解耦, 但牺牲了访问策略灵活性和功能复杂性。相比之下, 本文方案采用灵活的树形访问结构, 同时具备双向匹配与用户撤销等功能。

表2 各方案计算开销对比

方案	KeyGen	Encrypt	TokenGen	Search	Decrypt
文献[11]	$(n_u + 2)E$	$6E + 3P$	$5E$	$6P$	$E + P$
文献[15]	$3n_u E$	$(2n_o + n_u + 3)E + E_T$	—	—	$nE_T + 2nP$
文献[17]	$n_u(l_2 + 1)E$	$l_1(l_2 + 3)E + (2l_1 + 1)E_T + (l_1 + 1)P$	$2n_u E$	$n(l_2 + 2)P$	—
文献[18]	$(n_u + 5)E$	$(3v_u + 1)E + P$	$2E$	$3P$	$(3n + 5)P + 3nE$
文献[19]	$(n_u + 4)E$	$(3l_1 + 3)E + E_T$	$(n_u + 2)E$	$(2l_1 + 1)P + l_1 E_T$	$(2l_1 + 1)P + 2E_T$
本文方案	$(3n + 2n_u + 4 + d)E$	$(n_o + 2n + 12)E + E_T + P$	$(2n_u + 4)E$	$(2n + 4)P + nE_T$	E_T

表3 符号说明

符号	说明
E, E_T	G_0, G_T 上的幂运算
P	双线性配对运算
n_o	DO属性数量
n_u	DU属性数量
n	满足策略的属性数量
d	用户密钥中标签数量
t	限制策略中标签数量
l_1	访问矩阵行数
l_2	访问矩阵列数

7.2 实验分析

实验硬件环境为搭载 Intel Core i7-14650HX 2.20 GHz 处理器及 16 GB 内存的笔记本电脑，操作系统为 Windows 11。密码算法基于 Java 标准密码库 JPBC 实现，采用 Type A 椭圆曲线 $y^2 = x^3 + x$ 作为基础曲线，参数 p 和 q 的比特长度分别为 160 和 512。在相同实验设置下，将本文方案与文献[11, 17-19]进行对比，评估属性数量变化对各方案执行效率的影响。为保证实验对比的一致性与公平性，对控制变量进行如下设定：首先，鉴于标签穿刺机制为本文方案特有，设定用户标签数 d 恒为 10 以屏蔽该专属变量对整体性能的干扰；其次，考虑到文献[11]仅支持 AND 门，为统一测试基准，所有方案的输入策略逻辑均采用 AND 逻辑（即 $A_1 \wedge A_2 \wedge \cdots \wedge A_n$ ）。

密钥生成算法效率对比如图 4 所示。本文方案为用户生成同时包含属性和访问结构的私钥以实现双向匹配，因此开销较高。实验数据表明，当属性数量分别为 10、20 和 30 时，单次密钥生成时间分别为 326.75 ms、576.08 ms 和 813.34 ms，理论上对应的同等算力、单核 KGC 并发吞吐量分别为 3.06 次/秒、1.74 次/秒和 1.23 次/秒。这一实验结果表明，若将该集中式 KGC 部署于数以万计车辆高频并发注册

的城市级 VSN 中，集中式 KGC 将成为系统吞吐量瓶颈。因此，本文方案更适用于“低频加入、高频撤销”的区域级 VSN。

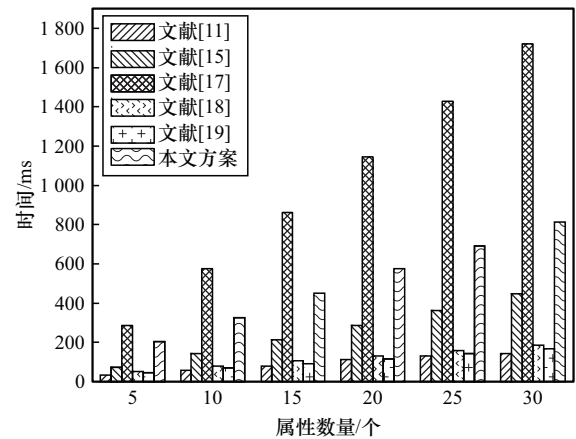


图4 密钥生成算法效率对比

加密算法效率对比如图 5 所示。在加密阶段，本文方案与文献[15]、文献[18]和文献[19]开销相近，文献[15]虽也支持双向匹配，但其不具备关键词搜索和用户撤销功能；文献[11]虽效率最高，但其仅支持 AND 门访问结构；本文方案则采用树形访问结构，支持 AND 门、OR 门及门限逻辑结构。

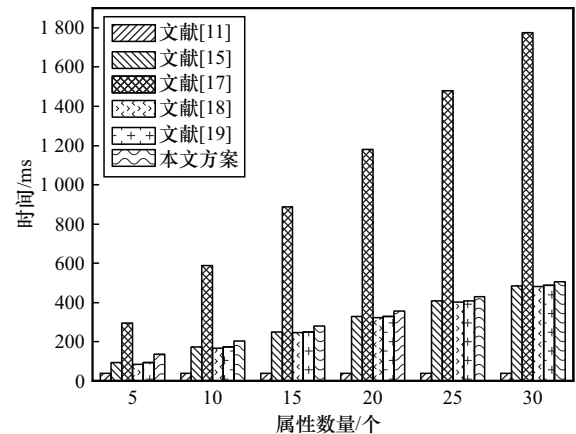


图5 加密算法效率对比

令牌生成算法效率对比如图6所示。在令牌生成阶段,本文方案的计算开销高于对比方案。为缓解DU生成令牌的计算压力,本文采用离线/在线令牌生成算法,将大部分计算转移至离线阶段完成。

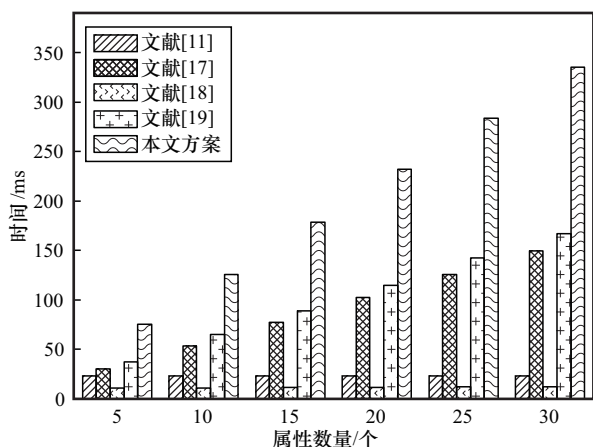


图6 令牌生成算法效率对比

搜索算法效率对比如图7所示。在搜索阶段,文献[17]由于其去中心化的多属性权威设计耗时最长;本文方案与文献[19]的开销相近;文献[18]在搜索阶段仅执行关键词匹配,不验证访问策略,因此表现最优。

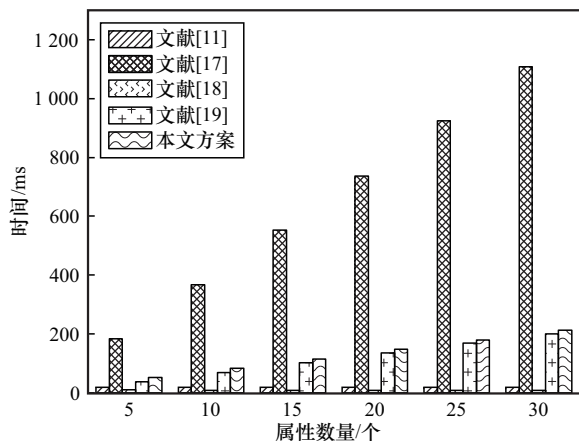


图7 搜索算法效率对比

解密算法效率对比如图8所示。由于本文方案采用外包解密技术,用户仅需在本地执行少量常数级计算即可恢复明文,因此解密效率最高。文献[15]、文献[18]和文献[19]在解密过程中需验证策略与属性是否匹配,因此耗时较长。

图9展示了离线/在线加密算法的执行效率。实验结果表明,经过离线加密,DO在线加密的速度得到了显著提升。

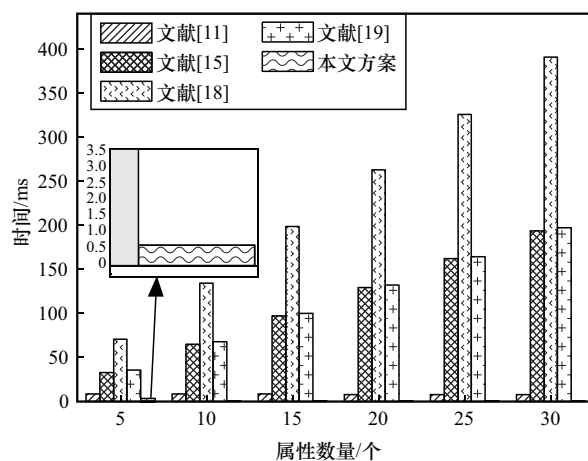


图8 解密算法效率对比

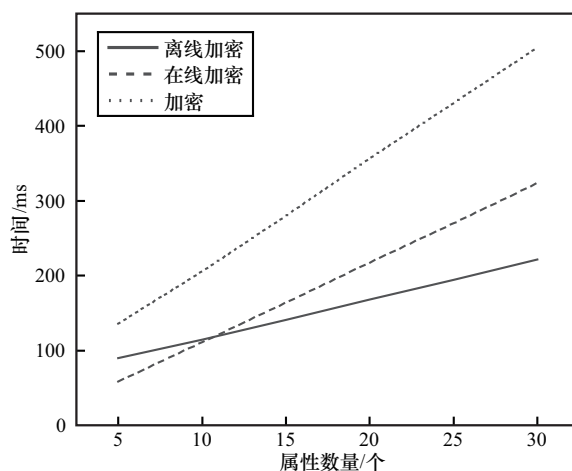


图9 离线/在线加密算法效率

图10展示了离线/在线令牌生成算法的执行效率。结果表明,离线计算承担了绝大部分计算开销,使在线阶段DU仅需少量常数级计算即可完成令牌生成,显著降低了实时计算压力。

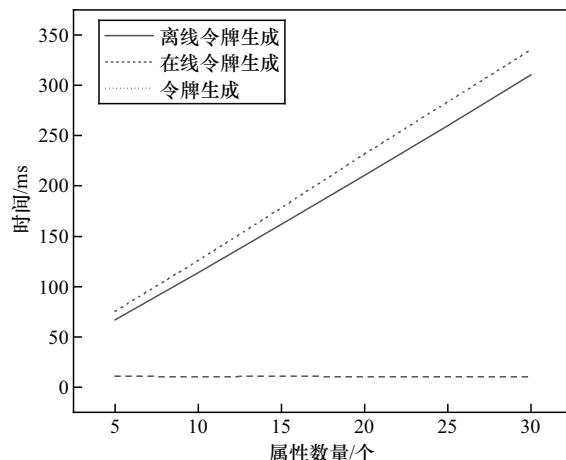


图10 离线/在线令牌生成算法的执行效率

图 11 分析了外包解密算法在不同属性数量与限制策略标签数量影响下的性能表现。结果显示, 属性数量与标签数量的增加均会导致耗时线性增长, 其中标签数量对解密时间的影响更为显著。

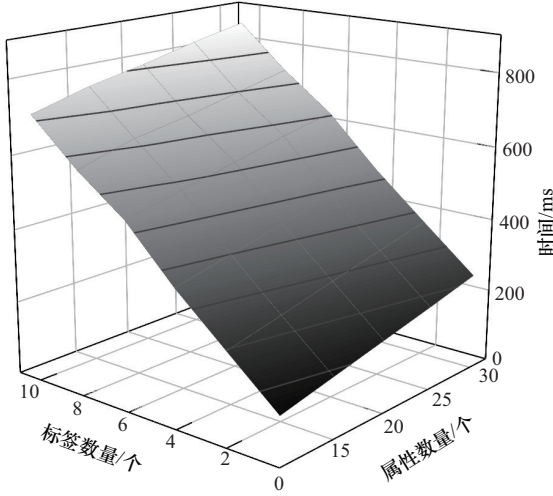


图 11 外包解密算法效率

8 结束语

本文提出了适用于车载社交网络支持双向匹配的可穿刺基于属性可搜索加密 (Pun-ABSE-BM), 给出了 Pun-ABSE-BM 系统模型、形式化定义以及安全模型, 并构造了一个具体方案。该方案支持关键词搜索与双向匹配, 通过穿刺加密实现不需要密钥更新与密文重加密的灵活高效用户撤销。此外, 该方案通过离线/在线加密算法、离线/在线令牌生成算法以及外包解密算法, 大大降低资源受限设备的本地计算压力。该方案满足 IND-sCKA、IND-sCPA 及关键词保密性, 在兼具功能性和安全性的同时具有良好的执行效率, 适用于中小型 VSN 细分场景。针对本文方案集中式 KGC 架构在大规模城市级 VSN 部署中存在的性能瓶颈与单点故障风险, 未来工作将重点研究适配车载场景的多权威分布式授权机制。

附录 1 IND-sCKA、关键词保密性及 IND-sCPA 详细安全性证明

定理 1 基于 DL 假设和单向哈希函数 H_2 , Pun-ABSE-BM 方案在随机预言模型中满足 IND-sCKA。

证明 若存在一个多项式时间复杂度的敌手 $\mathcal{A}$, 其在 Game-I 博弈中以优势 ε 获胜, 则存在一个挑战者算法, 能以优势 $\frac{\varepsilon}{2}$ 解决 DL 问题。

以优势 $\frac{\varepsilon}{2}$ 解决 DL 问题。

初始化: 敌手 $\mathcal{A}$ 声明挑战属性集 Atts^* , 挑战者 $\mathcal{B}$ 获得 DL 挑战实例 $(g, f, h, f^{r_1}, g^{r_2}, Q)$, 其中 $r_1, r_2 \in \mathbb{Z}_p$, $g, f, h, Q \in G_0$ 。

设置阶段: 挑战者 $\mathcal{B}$ 隐式设置 $g^a = h$, $g^c = f$, 且 $\mathcal{B}$ 不知道 a 和 c 。 $\mathcal{B}$ 选择随机值 $d \leftarrow \mathbb{Z}_p$, 通过计算 $g^b = f^d = g^{cd}$, 隐式设置 $b = cd$ 。设 H_2 为单向哈希函数, $\mathcal{B}$ 将 $\text{PK} = (g, p, h, f^d, f)$ 发送给 $\mathcal{A}$, 保留私钥 $\text{MSK} = (d)$ 。对于所有属性 $j \in U$, 如果 $j \in \text{Atts}^*$, $\mathcal{B}$ 选择 $\beta_j \leftarrow \mathbb{Z}_p$, 设置 $\alpha_j = 0$, 计算 $h_j = g^{\beta_j}$; 如果 $j \notin \text{Atts}^*$, $\mathcal{B}$ 选择 $\alpha_j, \beta_j \leftarrow \mathbb{Z}_p$, 计算 $h_j = f^{\alpha_j} g^{\beta_j}$ 。

查询阶段 1: $\mathcal{A}$ 可以多项式次数自适应查询以下预言机, $\mathcal{B}$ 维护一个初始为空的列表 L_{kw} 。

$\mathcal{O}_{\text{KeyGen}}(\mathcal{T}^*)$: $\mathcal{A}$ 向 $\mathcal{B}$ 发送一个访问树 $\mathcal{T}^*$ 。如果 $F(\text{Atts}^*, \mathcal{T}^*) = 1$, 则 $\mathcal{B}$ 终止; 否则, $\mathcal{B}$ 选择随机值 $z \leftarrow \mathbb{Z}_p$, 本地维护请求记录 $[\mathcal{T}^*, g^z]$, 并按以下方式生成密钥。

定义以下步骤以确定 $\mathcal{T}^*$ 中每个节点的多项式。

(1) $\text{PolySat}(\mathcal{T}_x, \text{Atts}, \lambda_x)$: 当 $F(\text{Atts}, \mathcal{T}_x) = 1$ 时, 输入根节点秘密值 λ_x , 该过程确定以 x 为根的 $\mathcal{T}_x$ 中每个节点的多项式。其工作原理如下: 假设节点 x 的阈值为 k_x , 设 $q_x(0) = \lambda_x$, 并随机选取 $k_x - 1$ 个系数确定多项式 q_x 。对于 x 的每个子节点 x' , 递归调用 $\text{PolySat}(\mathcal{T}_{x'}, \text{Atts}, \lambda_{x'})$, 其中 $\lambda_{x'} = q_x(\text{index}(x'))$ 。若 x 是叶子节点, 无任何子节点, 则门限 $k_x = 1$, 多项式为常数多项式 $q_x(0) = \lambda_x$, 不需要选取系数和递归。

(2) $\text{PolyUnsat}(\mathcal{T}_x, \text{Atts}, g^{\lambda_x})$: 当 $F(\text{Atts}, \mathcal{T}_x) = 0$ 时, 输入已知群元素 g^{λ_x} (仅知道 g^{λ_x} , 不知道 λ_x 本身), 对根节点 x 构造多项式 q_x , 满足 $q_x^{q_x(0)} = g^{\lambda_x}$ 。设节点 x 的子节点中, 子树满足属性集 Atts 的子节点数为 h_x (因 $F(\text{Atts}, \mathcal{T}_x) = 0$, 故 $h_x < k_x$); 对 h_x 个满足属性集的子节点 x' , 随机选取 $\lambda_{x'} \leftarrow \mathbb{Z}_p$, 并设置 $q_x(\text{index}(x')) = \lambda_{x'}$ 。然后设置剩余 $k_x - h_x$ 个节点以确定多项式 q_x , 并确保 $q_x^{q_x(0)} = g^{\lambda_x}$ 。对于 x 的每个子节点 x' : 若 $F(\text{Atts}, \mathcal{T}_{x'}) = 1$, $\mathcal{B}$ 已知 $q_x(\text{index}(x'))$ 的值, 执行 $\text{PolySat}(\mathcal{T}_{x'}, \text{Atts}, q_x(\text{index}(x')))$; 否则, 调用 $\text{PolyUnsat}(\mathcal{T}_{x'}, \text{Atts}, g^{\lambda_{x'}})$, 其中 $\mathcal{B}$ 已知 $g^{\lambda_{x'}}$ 。

由于 $F(\text{Atts}^*, \mathcal{T}^*) = 0$, 因此 Atts^* 不满足 $\mathcal{T}^*$, $\mathcal{B}$ 执行 $\text{PolyUnsat}(\mathcal{T}^*, \text{Atts}^*, g^a)$, 隐式设置根节点秘密为 a , 即 $q_{\text{root}}(0) = a$ 。算法执行后, 对于每个叶子节点属性 $v \in \mathcal{T}^*$: 若 $v \in \text{Atts}^*$, 则 $\mathcal{B}$ 知道 $q_v(0)$; 若 $v \notin \text{Atts}^*$, 则 $\mathcal{B}$ 只知道 $g^{q_v(0)}$ 而不知道 $q_v(0)$ 。

因此, 对于每个叶子节点属性 v : 若有 $v = j \in \text{Atts}^*$,

则选择 $\varpi \leftarrow \mathbb{Z}_p^*$, 计算 $\text{KP}_{1,v} = f^{\frac{q_v(0)}{z}} h_j^{\varpi} = g^{\frac{cq_v(0)}{z}} h_j^{\varpi}$, $\text{KP}_{2,v} = g^{\varpi}$; 若有 $v = j \notin \text{Atts}^*$, 则 $h_j = f^{\alpha_j} g^{\beta_j} = g^{c\alpha_j + \beta_j}$ 。选择 $\varpi' \leftarrow \mathbb{Z}_p^*$, $\mathcal{B}$ 只知道 $g^{q_v(0)}$ 但不知道 $q_v(0)$, 计算:

$$\begin{aligned}
\text{KP}_{2,v} &= g^{\frac{-q_v(0)}{a_j}} g^{\varpi'} = g^{\varpi' - \frac{q_v(0)}{a_j}} \\
\text{KP}_{1,v} &= (g^{q_v(0)})^{\frac{-\beta_j}{z\alpha_j}} (h_j)^{\varpi'} = g^{\frac{-\beta_j q_v(0)}{z\alpha_j}} (f^{a_j} g^{\beta_j})^{\varpi'} = \\
&= f^{\frac{q_v(0)}{z}} f^{\frac{-q_v(0)}{z}} g^{\frac{-\beta_j q_v(0)}{z\alpha_j}} (f^{a_j} g^{\beta_j})^{\varpi'} = \\
&= f^{\frac{q_v(0)}{z}} (f^{a_j} g^{\beta_j})^{\frac{-q_v(0)}{z\alpha_j}} (f^{a_j} g^{\beta_j})^{\varpi'} = \\
&= g^{\frac{cq_v(0)}{z}} (h_j)^{\varpi' - \frac{q_v(0)}{z\alpha_j}}
\end{aligned}$$

隐式设置 $\varpi = \varpi' - \frac{q_v(0)}{z\alpha_j}$, 则 $\{\text{KP}_{1,v}, \text{KP}_{2,v}\}_{v \in T_u}$ 和 z 给 $\mathcal{A}$ 。

钥, 返回 $\text{SK}_{\mathcal{A}} = (T^*, \{\text{KP}_{1,v}, \text{KP}_{2,v}\}_{v \in T_u})$ 和 z 给 $\mathcal{A}$ 。

$\mathcal{O}_{\text{TokenGen}}(T^*, w)$: 挑战者 $\mathcal{B}$ 执行 $\mathcal{O}_{\text{KeyGen}}(T^*)$ 以获取密钥 $\text{SK}_{\mathcal{A}} = (T^*, \{\text{KP}_{1,v}, \text{KP}_{2,v}\}_{v \in T_u})$, 计算 $\text{TK} \leftarrow \text{TokenGen}(w, \text{SK}_{\mathcal{A}}, \text{PK}_{\mathcal{K}})$, 并将 TK 返回至 $\mathcal{A}$ 。若 $F(\text{Atts}^*, T^*) = 1$, 挑战者将 w 添加至列表 L_{kw} 。

挑战阶段: $\mathcal{A}$ 提交两个等长的关键词 w_0 和 w_1 , 要求 $w_0, w_1 \notin L_{\text{kw}}$ 。 $\mathcal{B}$ 选择 $\ell \leftarrow \{0, 1\}$ 来构造密文: $W' = f^{r_1}$, $W = Q(f^{r_1})^{dH_2(w_\ell)}$, $W_0 = g^{r_2}$, 对于每个 $j \in \text{Atts}^*$, 令 $W_j = (g^{r_2})^{\beta_j}$ 。此外, $\mathcal{B}$ 生成关于 Atts^* 合法的签名组件 Signatur 加入密文, 这些组件并不影响 IND-sCKA 的核心安全目标。 $\mathcal{B}$ 输出挑战密文 $\text{ckw}^* = (\text{Atts}^*, W, W', W_0, \{W_j\}_{j \in \text{Atts}^*}, \text{Signatur})$ 。

若 $Q = h^{r_1 + r_2}$, 则 ckw^* 是关键词 w_ℓ 的合法密文, $W = Q(f^{r_1})^{dH_2(w_\ell)} = h^{r_1 + r_2} g^{cdr_1 H_2(w_\ell)} = g^{a(r_1 + r_2)} g^{br_1 H_2(w_\ell)}$, $W' = f^{r_1} = g^{cr_1}$, $W_0 = g^{r_2}$, $W_j = (g^{r_2})^{\beta_j} = (h_j)^{r_2}$ 。

查询阶段 2: $\mathcal{A}$ 可以继续查询 $\mathcal{O}_{\text{KeyGen}}$ 和 $\mathcal{O}_{\text{TokenGen}}$, 但不能查询满足 $F(\text{Atts}^*, T^*) = 1$ 的 w_0 或 w_1 的令牌。 $\mathcal{B}$ 的回答方式与查询阶段 1 相同。

猜测阶段: $\mathcal{A}$ 输出猜测 $\ell' \in \{0, 1\}$, $\mathcal{B}$ 根据 $\mathcal{A}$ 的输出来判断 DL 问题。如果 $\ell' = \ell$, 则 $\mathcal{B}$ 猜测 $Q = h^{r_1 + r_2}$, 输出 1; 如果 $\ell' \neq \ell$, 则 $\mathcal{B}$ 猜测 Q 是随机群元素, 输出 0。

分析: 在挑战阶段, 若 $Q = h^{r_1 + r_2}$, 则 ckw^* 是关键词 w_ℓ 的合法密文, $\mathcal{A}$ 猜测 $\ell' = \ell$ 的概率 $\Pr[\ell' = \ell | Q = h^{r_1 + r_2}] = \frac{1}{2} + \varepsilon$ 。如果 Q 是随机群元素, 则 ckw^* 不是关键词 w_ℓ 的合法密文, $\mathcal{A}$ 猜测 $\ell' = \ell$ 的概率 $\Pr[\ell' = \ell | Q] = \frac{1}{2}$ 。因此, $\mathcal{B}$ 解决 DL 问题的优势 $\text{Adv}_{\mathcal{B}}^{\text{DL}} = \frac{1}{2} (\frac{1}{2} + \varepsilon + \frac{1}{2}) - \frac{1}{2} = \frac{\varepsilon}{2}$ 。如果 $\mathcal{A}$ 能以不可忽略的优势 ε 赢得 Game-I, 则 $\mathcal{B}$ 能以优势 $\frac{\varepsilon}{2}$ 解决 DL 问题。但根据 DL 假设, 任何 PPT 算法解决该问题的优势是可忽略的。因此, ε 可忽略, 即 Pun-ABSE-BM 方案是 IND-sCKA 安全的。证毕。

定理 2 设 H_2 为单向哈希函数, Pun-ABSE-BM 方案满

足关键词保密性。

证明 构建一个 Game-II 的挑战者 $\mathcal{B}$, 游戏过程各阶段仅生成支撑关键词保密性的核心参数, 具体如下。

设置阶段: 挑战者 $\mathcal{B}$ 选择 $a, b, c \leftarrow \mathbb{Z}_p^*$, $f \leftarrow G_0$, 令 $\text{PK} = (g, g^a, g^b, g^c, f)$, $\text{MSK} = (a, b, c)$ 。对于所有属性 $j \in U$, 挑战者 $\mathcal{B}$ 选择 $\alpha_j \leftarrow \mathbb{Z}_p^*$, 计算 $h_j = g^{\alpha_j}$ 。

挑战阶段: $\mathcal{A}$ 可对以下预言机进行多项式次数的自适应查询。

$\mathcal{O}_{\text{KeyGen}}(T)$: $\mathcal{A}$ 向 $\mathcal{B}$ 发送 T , $\mathcal{B}$ 执行 KeyGen 算法中转换密钥相关部分, 生成密钥 $\text{SK}_{\mathcal{A}}$ 返回给 $\mathcal{A}$, 同时将 T 添加至初始为空的 L_{KeyGen} 列表。

$\mathcal{O}_{\text{TokenGen}}(T, w)$: $\mathcal{B}$ 执行 $\mathcal{O}_{\text{KeyGen}}(T)$ 获取 $\text{SK}_{\mathcal{A}}$ 。 $\mathcal{B}$ 执行 TokenGen 算法生成令牌 $\text{TK}_{\mathcal{A}}$ 并返回给 $\mathcal{A}$ 。

挑战阶段: $\mathcal{A}$ 选择属性集 Atts^* , $\mathcal{B}$ 选择访问控制策略 T^* , 使 $F(\text{Atts}^*, T^*) = 1$, 生成私钥 $\text{SK}^* \leftarrow \text{KeyGen}(T^*)$ 。以 Atts^* 和 SK^* 作为输入, $\mathcal{B}$ 从关键词空间中均匀随机选择 w^* , 并通过 Encrypt 和 TokenGen 算法为 w^* 生成 ckw^* 和 TK^* 。 Atts^* 应满足关键词保密游戏中定义的要求。

猜测阶段: $\mathcal{A}$ 输出一个关键字 w' 并发送给挑战者, 挑战者 $\mathcal{B}$ 执行 Encrypt 算法生成 ckw' 。若 ckw' 与 TK^* 匹配, 则 $\mathcal{A}$ 赢得游戏。

分析: 若 $\mathcal{A}$ 在关键词保密性游戏中已发起 q 次不同的关键词查询, 则 $\mathcal{A}$ 获胜的概率至多为 $\frac{1}{|W| - q} + \varepsilon$, 其中,

$|W|$ 表示关键词空间的大小, ε 表示可忽略优势。这是因为剩余关键词空间的大小为 $|W| - q$, 而 H_2 作为单向哈希函数, 意味着从 $H_2(w^*)$ 推导出 w^* 的概率最多为可忽略的 ε 。因此, 给定 $\mathcal{A}$ 已尝试的 q 个不同关键词, $\mathcal{A}$ 在 Game-II 中获胜的概率最多为 $\frac{1}{|W| - q} + \varepsilon$ 。由此可知, Pun-ABSE-BM 方案实现了关键词保密性。证毕。

定理 3 基于 DBDH 假设, 且 H_1 建模为随机预言机时, Pun-ABSE-BM 方案满足 IND-sCPA。

证明 输入 DBDH 挑战实例 $(g, A = g^a, B = g^b, C = g^c, Z)$, 挑战者 $\mathcal{B}$ 的目标是在 Game-III 中判断 $Z = e(g, g)^{abc}$ 还是 Z 为 G_T 中随机元素。游戏过程各阶段仅生成支撑密文 IND-sCPA 安全的核心参数。

初始化: 敌手 $\mathcal{A}$ 声明访问策略 T^* 和限制策略 $\mathcal{RP}^*$, 并承诺后续查询的用户属性集 Atts^* 不满足 T^* 且标签集 Tag^* 不满足 $\mathcal{RP}^*$ 。

设置阶段: 挑战者 $\mathcal{B}$ 选择随机数 $\gamma, \alpha, \beta \leftarrow \mathbb{Z}_p^*$, 隐式设置 $g^a = g^a = A$, $g^b = g^b = B$, 通过计算 $e(g, g)^\gamma = e(g^a, g^b) = e(A, B)$ 隐式设置 $\gamma = ab$ 。

对于所有属性 $j \in U$, $\mathcal{B}$ 选择随机指数 $\tau_j \in \mathbb{Z}_p^*$, 若 $j \in T^*$ 则 $h_j = g^{\alpha \tau_j} = A^{\tau_j}$, 若 $j \notin T^*$ 则 $h_j = g^{\tau_j}$ 。

挑战者 $\mathcal{B}$ 设置一个多项式 Q_x , 其中 $q(0) = \beta$, 选择一组

随机数 $\{\vartheta_{t_0}, \vartheta_{t_1}, \dots, \vartheta_{t_d}\}$, 令 $q(t_i) = \vartheta_{t_i}$, $Q(t_i) = g^{q(t_i)}$, $i \in [0, d]$, 设 t_0 为正常操作期间不使用的特殊标记。 $\mathcal{B}$ 发送给 $\mathcal{A}$ 公钥 $PK = (g, A, B, e(g, g)^r, Q(t_1), \dots, Q(t_d), \{h_j\}_{j \in [1, |U|]}, t_0)$ 。

查询阶段 1: $\mathcal{A}$ 重复查询 $\mathcal{O}_{\text{KeyGen}}(\text{Atts}^*, \text{Tag}^*)$ 且 Atts^* 不满足 $\mathcal{T}^*$, Tag^* 不满足 $\mathcal{RP}^*$ 。

$\mathcal{O}_{\text{KeyGen}}(\text{Atts}^*, \text{Tag}^*)$: $\mathcal{B}$ 随机选择 $r, r', z \leftarrow \mathbb{Z}_p^*$ 。通过 $CP_2 = (g^r)^{\frac{1}{z}} = (\frac{g^{r'}}{B^2})^{\frac{1}{z}} = (g^{r'-2b})^{\frac{1}{z}}$ 隐式设置 $r = r' - 2b$, 计算 $CP_1 = (g^\gamma g^{r\alpha + \beta\alpha})^{\frac{1}{z}} = (g^{ab} g^{a(r'-2b+b)})^{\frac{1}{z}} = (g^{ar'})^{\frac{1}{z}} = (A^{r'})^{\frac{1}{z}}$, 针对每个属性 $j \in \text{Atts}^*$, 计算 $CP_{3j} = (g^{\omega} h_j^r)^{\frac{1}{z}} = (A^r h_j^r)^{\frac{1}{z}}$ 。对于敌手标签集中每个标签 $t_i \in \text{Tag}^*$, 计算 $P_i = Q(t_i)^\alpha = g^{a\vartheta_{t_i}} = A^{\vartheta_{t_i}}$, 返回 $SK_{\mathcal{A}} = (\text{Atts}^*, CP_1, CP_2, \{CP_{3j}\}_{j \in \text{Atts}^*}, \text{Tag}^*, \{P_i\}_{t_i \in \text{Tag}^*})$ 和 z 给 $\mathcal{A}$ 。

挑战阶段: 敌手 $\mathcal{A}$ 随机选择两个等长信息 M_0 和 M_1 发送给 $\mathcal{B}$ 。 $\mathcal{B}$ 选择一个随机数 $\ell \leftarrow \{0, 1\}$ 并构造 M_ℓ 的密文。 $\mathcal{B}$ 随机选择 $\psi, \rho, s \leftarrow \mathbb{Z}_p^*$, 通过 $C_3 = g^s = g^c$ 隐式设置 $s = c$, 随后计算 $K_s = e((g^\psi)^s, H_1(\text{ID}_s)) = e(g, H_1(\text{ID}_s))^{\psi\rho}$, 其中 H_1 为单向哈希函数。计算 $\tilde{C} = M_\ell K_s e(g, g)^{rs} = M_\ell K_s e(g, g)^{abc} = M_\ell K_s Z$, $C_1 = A^s$, $C_2 = B^s$, $C_3 = C$ 。 $\mathcal{B}$ 执行 $\text{Share}(\mathcal{T}^*, s)$ 算法计算 $\mathcal{T}^*$ 每个叶子节点属性秘密份额 $q_v(0)$, 对于每个属性 $v \in \mathcal{T}^*$, 计算 $C_{4,v} = h_v^{q_v(0)} = g^{a\tau_v q_v(0)} = A^{\tau_v q_v(0)}$, $C_{5,v} = g^{q_v(0)}$ 。初始化穿刺组件 PSK , 输出密文 cph^* 给敌手 $\mathcal{A}$ 。

$\text{PSK} = \{\text{psk}_0 = (t_0, K_0 = A^{(r_0+1)}, L_0 = g^{sr_0} = C^{r_0}, Q_0 = Q(t_0)^{r_0})\}$
 $\text{cph}^* = (\text{PSK}, \mathcal{T}^*, \tilde{C}, C_1, C_2, C_3, \{C_{4,v}, C_{5,v}\}_{v \in \mathcal{T}^*}, g^\rho)$

查询阶段 2: 该过程与查询阶段 1 的流程相同。

猜测阶段: $\mathcal{A}$ 输出猜测 ℓ' 。若 $\ell' = \ell$, 则 $\mathcal{B}$ 判定 $Z = e(g, g)^{abc}$ 输出 1, 否则 $\mathcal{B}$ 判定 Z 为随机元素, 输出 0。

分析: 若 $Z = e(g, g)^{abc}$, 则 cph^* 是有效密文, $\mathcal{A}$ 猜测 $\ell' = \ell$ 的概率为 $\frac{1}{2} + \varepsilon$; 若 Z 为随机元素, 则 cph^* 与明文无关, $\mathcal{A}$ 猜测 $\ell' = \ell$ 的概率为 $\frac{1}{2}$, 因此 $\mathcal{A}$ 解决 DBDH 问题的优势为 $|\frac{1}{2} + \varepsilon - \frac{1}{2}| = \varepsilon$ 。又因 DBDH 是困难问题, ε 可忽略, 所以 Pun-ABSE-BM 方案满足 IND-sCPA。另外, 在 Game-III 中, 穿刺操作不影响密文的不可区分性。证毕。

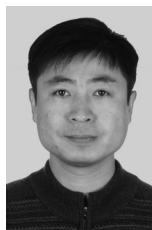
参考文献:

- [1] Ning Z L, Xia F, Ullah N, et al. Vehicular social networks: enabling smart mobility[J]. IEEE Communications Magazine, 2017, 55(5): 16-55.
- [2] Vegni A M, Loscri V. A survey on vehicular social networks[J]. IEEE Communications Surveys & Tutorials, 2015, 17(4): 2397-2419.
- [3] Kumar R, Kumar P, Tripathi R, et al. P2SF-IoV: a privacy-preservation-based secured framework for Internet of vehicles[J]. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(11): 22571-22582.

- [4] Sahai A, Waters B. Fuzzy identity-based encryption[C]//Advances in Cryptology-EUROCRYPT 2005. Berlin: Springer, 2005: 457-473.
- [5] Chen L Q, Tai Z L, Zhang H, et al. Traceable and revocable multi-authority attribute-based encryption with cloud and fog computing feasible for IoV[J]. IEEE Transactions on Dependable and Secure Computing, 2026, 23(3): 6137-6150.
- [6] Feng C S, Yu K P, Aloqaily M, et al. Attribute-based encryption with parallel outsourced decryption for edge intelligent IoV[J]. IEEE Transactions on Vehicular Technology, 2020, 69(11): 13784-13795.
- [7] Wang C H, Ming Y, Liu H, et al. Puncturable registered ABE for vehicular social networks: enhancing security and practicality[J]. IEEE Transactions on Dependable and Secure Computing, 2025, 22(6): 5998-6011.
- [8] Chen L Q, Xu S G, Zhang H, et al. Fair-and-exculpable-attribute-based searchable encryption with revocation and verifiable outsourced decryption using smart contract[J]. IEEE Internet of Things Journal, 2025, 12(4): 4302-4317.
- [9] Song D X, Wagner D, Perrig A. Practical techniques for searches on encrypted data[C]//Proceedings of the Proceeding 2000 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2000: 44-55.
- [10] Zheng Q J, Xu S H, Ateniese G. VABKS: verifiable attribute-based keyword search over outsourced encrypted data[C]//Proceedings of the IEEE INFOCOM 2014 - IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2014: 522-530.
- [11] Zhang K, Zhang Y, Li Y P, et al. A blockchain-based anonymous attribute-based searchable encryption scheme for data sharing[J]. IEEE Internet of Things Journal, 2024, 11(1): 1685-1697.
- [12] Hou R W, Zhou F C, Wang Q, et al. Revokable blockchain-enabled ranked multi-keyword attribute-based searchable encryption scheme with mobile edge computing for vehicular[J]. IEEE Transactions on Network and Service Management, 2025, 22(3): 2764-2779.
- [13] 郭瑞, 杨鑫, 贾晨阳, 等. 云辅助医疗物联网中支持策略隐藏的可搜索属性加密方案[J]. 密码学报, 2025, 12(1): 49-68.
- Guo R, Yang X, Jia C Y, et al. A searchable attribute-based encryption scheme supporting policy hiding in cloud-assisted Internet of medical things[J]. Journal of Cryptologic Research, 2025, 12(1): 49-68.
- [14] 刘媛, 王励成, 周永彬. TTTC-ABE: 可追踪可撤销的基于循环代数带误差学习问题的格基属性加密方案[J]. 电子与信息学报, 2025, 47(6): 1911-1926.
- Liu Y, Wang L C, Zhou Y B. TTTC-ABE: traitor traceable and revocable CLWE-based ABE scheme from lattices[J]. Journal of Electronics & Information Technology, 2025, 47(6): 1911-1926.
- [15] Xu S M, Ning J T, Li Y J, et al. Match in my way: fine-grained bilateral access control for secure cloud-fog computing[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 19(2): 1064-1077.
- [16] Cui J, Li B, Zhong H, et al. A practical and efficient bidirectional access control scheme for cloud-edge data sharing[J]. IEEE Transactions on Parallel and Distributed Systems, 2022, 33(2): 476-488.
- [17] Ghopur D, Ma J F, Ma X D, et al. Decentralized multiauthority attribute-based searchable encryption for E-health cloud[J]. IEEE Internet of Things Journal, 2025, 12(11): 15723-15735.
- [18] Chen L Q, Liu Y, Zhang H, et al. Multi-authority attribute-based searchable encryption with attribute revocation for Internet of vehicles[J]. IEEE Transactions on Vehicular Technology, 2026, 75(5): 8522-8533.
- [19] Chen L Q, Xu S G, Jin C H, et al. Partially hidden policy attribute-based multi-keyword searchable encryption with verification and revo-

- cation[J]. IEEE Transactions on Mobile Computing, 2025, 24(9): 9020-9035.
- [20] 李朋祥, 王玥欢, 贾春福, 等. 支持属性验证与用户撤销的IoMT数据共享方案[J]. 通信学报, 2025, 46(12): 121-137.
- Li P X, Wang Y H, Jia C F, et al. IoMT data sharing scheme supporting attribute verification and user revocation[J]. Journal on Communications, 2025, 46(12): 121-137.
- [21] Niu S F, Hu Y, Zhou S W, et al. Attribute-based searchable encryption in edge computing for lightweight devices[J]. IEEE Systems Journal, 2023, 17(3): 3503-3514.
- [22] Guo Y Y, Lu Z H, Ge H, et al. Revocable blockchain-aided attribute-based encryption with escrow-free in cloud storage[J]. IEEE Transactions on Computers, 2023, 72(7): 1901-1912.
- [23] Xiong H, Huang X, Yang M H, et al. Unbounded and efficient revocable attribute-based encryption with adaptive security for cloud-assisted Internet of things[J]. IEEE Internet of Things Journal, 2022, 9(4): 3097-3111.
- [24] 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: ACM Press, 2006: 89-98.
- [25] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption[C]//Proceedings of the 2007 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2007: 321-334.
- [26] Li J G, Zhang E F, Zhang Y C, et al. A traceable and revocable ciphertext policy attribute-based encryption with policy authentication[J]. IEEE Transactions on Dependable and Secure Computing, 2026, 23(2): 2962-2973.
- [27] Yang M X, Wang H Q, He D B. Puncturable attribute-based encryption from lattices for classified document sharing[J]. IEEE Transactions on Information Forensics and Security, 2024, 19: 4028-4042.
- [28] 李宇溪, 陈晶晶, 周福才, 等. 基于区块链索引的可验证布尔搜索加密方案[J]. 计算机研究与发展, 2025, 62(11): 2855-2869.
- Li Y X, Chen J J, Zhou F C, et al. Verifiable Boolean searchable encryption scheme based on blockchain index[J]. Journal of Computer Research and Development, 2025, 62(11): 2855-2869.
- [29] Ateniese G, Francati D, Nuñez D, et al. Match me if you can: match-making encryption and its applications[J]. Journal of Cryptology, 2021, 34(3): 16.
- [30] Wu A X, Feng D G, Zhang M, et al. Privacy-preserving bilateral multi-receiver matching with revocability for mobile social networks[J]. IEEE Transactions on Mobile Computing, 2024, 23(12): 11080-11090.
- [31] Wang H G, Chen K F, Xie Q, et al. Post-quantum secure identity-based matchmaking encryption[J]. IEEE Transactions on Dependable and Secure Computing, 2025, 22(1): 833-844.
- [32] 陈洁, 楚乔涵, 杜秋妍, 等. 标准模型下效率更高的身份基匹配加密[J]. 软件学报, 2025, 36(2): 698-714.
- Chen J, Chu Q H, Du Q Y, et al. More efficient identity-based matchmaking encryption under standard model[J]. Journal of Software, 2025, 36(2): 698-714.
- [33] Piretti M, Traynor P, McDaniel P, et al. Secure attribute-based systems[C]//Proceedings of the 13th ACM Conference on Computer and Communications Security. New York: ACM Press, 2006: 99-112.
- [34] Jiang F, Qi B R, Wu T H, et al. CPSS: CP-ABE based platoon secure sensing scheme against cyber-attacks[C]//Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC). Piscataway: IEEE Press, 2019: 3218-3223.
- [35] Green M D, Miers I. Forward secure asynchronous messaging from puncturable encryption[C]//Proceedings of the 2015 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2015: 305-320.
- [36] Phuong T V X, Ning R, Xin C S, et al. Puncturable attribute-based encryption for secure data delivery in Internet of things[C]//Proceedings of the IEEE INFOCOM 2018 - IEEE Conference on Computer Communications. Piscataway: IEEE Press, 2018: 1511-1519.
- [37] Ghopur D, Ma J F, Ma X D, et al. Puncturable key-policy attribute-based encryption scheme for efficient user revocation[J]. IEEE Transactions on Services Computing, 2023, 16(6): 3999-4011.
- [38] Lewko A, Waters B. Decentralizing attribute-based encryption[C]//Advances in Cryptology-EUROCRYPT 2011. Berlin: Springer, 2011: 568-588.

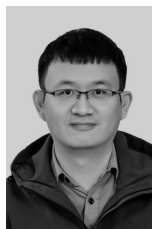
[作者简介]



陈礼青 (1982-), 男, 博士, 淮安大学副教授, 主要研究方向为密码学、云计算安全及隐私保护。



郭笑笑 (1998-), 女, 淮安大学硕士生, 主要研究方向为密码学、云计算安全及隐私保护。



刘志全 (1989-), 男, 博士, 暨南大学教授, 主要研究方向为密码学、车联网安全、数据安全。