软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2023,34(9):4256−4274 [doi: 10.13328/j.cnki.jos.006638]

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

面向便携式诊所的安全数据共享方案*

朱雪岭[1], 侯慧莹[2], 付绍静[1], 赵运磊[2], 刘 波[1]

[1](国防科技大学 计算机学院, 湖南 长沙 410073)

[2](复旦大学 计算机学院, 上海 200433)

通信作者: 刘波, E-mail: Kyle.liu@nudt.edu.cn

摘 要: 随着物联网 (Internet of Things, IoT)、云计算等技术的飞速发展, 便携式诊所 (portable health clinic, PHC) 得以实现, 并广泛应用于远程医疗. 我国依托 5G 通信的大幅优势, 积极推进智慧医疗的建设, 搭建了多功能、高质量的远程医疗信息服务平台. 以 PHC 为代表的远程医疗得以实现, 离不开远程数据共享系统的技术支撑. 目前 IoT 和云服务器 (cloud server, CS) 相结合 (通常称为云边协同) 的远程数据共享系统以其灵活性、高效性广受关注, 然而其隐私和安全问题却鲜有研究. 考虑到医疗数据的敏感性, 致力于研究 PHC 数据共享系统的安全隐私问题, 实现 PHC 系统中物联网感知数据的安全上传、个性密文的归一化、云服务器上动态多用户的细粒度访问控制、高效的解密操作, 并给出形式化的安全性证明. 在具体创新上, 第一, 分别对经典的代理重加密和属性基加密算法进行改进, 提出 IPRE-TO-FAME 组合加密机制, 以保障云边协同的 PHC 系统数据共享的安全性. 第二, 为了应对物联网终端数量众多、分散性强带来的密钥更新难题, 借鉴代理重加密 (proxy re-encryption, PRE) 的思想, 实现基于单方变换的密钥更新, 即无需变换 IoT 终端密钥条件下的密钥更新. 同时, 应用场景中重加密方可视为完全可信, 而常规 PRE 机制重加密方通常为不可信的第三方服务器, 为此, 改进经典 PRE 算法, 提出一种高效的 IPRE (improved PRE) 算法, 以适应提出的场景; 第三, 改进经典的 FAME (fast attribute-based message encryption) 机制, 实现动态多用户的细粒度访问控制, 便于用户可以随时随地使用便携式智能设备访问数据. 安全性证明、理论分析和实验结果证明, 提出的方案具有较好的安全性和较强的实用性, 是一类解决 PHC 安全数据共享问题的有效方案.

关键词: 便携式诊所; 属性基加密; 本地重加密; 物联网 (IoT); 解密外包

中图法分类号: TP309

中文引用格式: 朱雪岭, 侯慧莹, 付绍静, 赵运磊, 刘波. 面向便携式诊所的安全数据共享方案. 软件学报, 2023, 34(9): 4256–4274. http://www.jos.org.cn/1000-9825/6638.htm

英文引用格式: Zhu XL, Hou HY, Fu SJ, Zhao YL, Liu B. Secure Data Sharing Solution for Portable Health Clinics. Ruan Jian Xue Bao/Journal of Software, 2023, 34(9): 4256–4274 (in Chinese). http://www.jos.org.cn/1000-9825/6638.htm

Secure Data Sharing Solution for Portable Health Clinics

ZHU Xue-Ling[1], HOU Hui-Ying[2], FU Shao-Jing[1], ZHAO Yun-Lei[2], LIU Bo[1]

[1](College of Computer Science and Technology, National University of Defense Technology, Changsha 410073, China)

[2](School of Computer Science, Fudan University, Shanghai 200433, China)

Abstract: With the rapid development of technologies such as the Internet of Things (IoT) and cloud computing, portable health clinics (PHCs) have been realized and widely used in telemedicine. Relying on the significant advantages of 5G communications, China has actively promoted the construction of smart healthcare and built a multi-function and high-quality telemedicine information service platform.The realization of telemedicine represented by PHCs is inseparable from the technical support of remote data-sharing systems. At

* 基金项目: 国家自然科学基金 (62072466)
收稿时间: 2021-08-27; 修改时间: 2021-10-24, 2021-11-11, 2021-12-13; 采用时间: 2021-12-23; jos 在线出版时间: 2022-03-24
CNKI 网络首发时间: 2023-02-24

present, the remote data-sharing system combining IoT and the cloud server (CS) has attracted wide attention due to its flexibility and efficiency, but its privacy and security issues are rarely studied. Considering the sensitivity of medical data, this paper endeavors to study the security and privacy issues in the PHC data-sharing system. As a result, in the PHC system, this study achieves the secure uploading of IoT awareness data, normalization of personalized ciphertexts, dynamic multi-user fine-grained access control, and efficient decryption operations, and it also presents formal security verification. The specific innovations of this study are as follows: (1) The classical proxy re-encryption (PRE) and attribute-based encryption algorithms are improved, and an IPRE-TO-FAME combined encryption mechanism is proposed to ensure the data-sharing security of the PHC system with cloud-edge collaboration. (2) To address the challenge of key updates caused by many highly distributed IoT terminals, this paper uses the idea of PRE to realize the key updates on the basis of the unilateral transformation without changing the keys to IoT terminals. Meanwhile, the re-encryption entities can be regarded as fully trusted in the application scenarios of this study, which is different from the situation of the conventional PRE mechanism, where the re-encryption entities are usually untrusted third-party servers. Therefore, the conventional PRE algorithm is improved, and an efficient improved PRE (IPRE) algorithm is put forward to adapt to the scenarios proposed in this study. (3) The classical fast attribute-based message encryption (FAME) mechanism is improved to enable dynamic multi-user fine-grained access control. In this way, users can easily use portable intelligent devices to access data anytime and anywhere. The security proofs, theoretical analysis, and experimental results reveal that the proposed solution is highly secure and practical, which is an effective way to ensure secure PHC data sharing.

Key words: portable health clinic; attribute-based encryption; proxy re-encryption; Internet of Things (IoT); decryption outsourcing

近年来, 随着 5G 通信、物联网 (Internet of Things, IoT)、云计算等技术的飞速发展及大规模商用, 医疗健康领域也进行着重大革新. 便携式诊所 (portable health clinic, PHC) 的概念于 2010 年提出[1]并在孟加拉国、印度、巴基斯坦、柬埔寨等多个国家投入应用[2], 有效缓解了一些交通不便而又疾病肆虐地区的医疗压力. 我国基于“互联网+医疗健康”模式, 依托 5G 通信的大幅优势, 积极推进智慧医疗的建设, 搭建了多功能、高质量的远程医疗信息服务平台, 不仅直接为偏远山区、高原等地区和国内高端医院之间架起了桥梁, 而且为实现全国人口健康信息化建设奠定了基础. 特别地, 当用于一些偏远地区的公共卫生事件 (如新冠肺炎) 防治工作时, 如果使用 PHC 进行样本数据的采集与初步分析, 然后上传至服务器进行共享, 可能会有助于缓解医疗资源不足带来的压力.

PHC 系统利用物联网感知设备 (IoT sensor, IoTS) 测量病人的生命体征, 并将数据上传至本地诊所. 为了便于数据挖掘、共享与应用, 本地诊所将来自 IoTS 的数据进行归一化处理, 然后经专用安全信道 (如 SSL) 传输至远程的在线服务器, 病人、医生、药学专家、数据分析师等可以通过便携式终端 (如智能手机、平板) 查询病史档案、开展数据分析等工作.

虽然 PHC 系统让更多人享受到了便捷的医疗服务, 然而, 由于系统中的病史数据涉及大量个人隐私, 比如疾病, 过敏数据, 家庭监控数据, 免疫接种, 药物, 遗传信息, 家庭历史, 社会模式或生活方式等信息, 可能对病人的私生活和健康造成潜在威胁[3]. 此外, 大规模的医疗数据库还可以作为战略资源. 因此如何实现系统的隐私和安全是一个亟待解决的问题.

PHC 数据共享系统的网络架构可视为典型的“云-边-端”结构, “端”即 PHC 盒中的物联网感知设备, “边”即部署在本地诊所的可移动边缘计算中心, “云”即远程在线服务器. PHC 系统使用密码工具确保数据安全, 需要将“端”采集的数据进行个性化加密, 再经“边”重新加密成归一化的共享密文, 然后上传至“云”进行存储和管理, 用户通过与“云”的交互进行数据的访问. 图 1 给出了 PHC 系统的结构示意图. 根据图示, PHC 安全数据共享要解决的根本问题可以归纳为: 是否存在一个方案, 使得众多资源受限设备 A 的个性化密文, 可以在 B 实现归一化处理, 并通过半诚实的云平台安全的共享至多个被授权的用户 C?

为解决上述问题, 如果使用传统的对称加密机制, 一种最直观的解决方案是 B 对接收自 A 的密文先解密, 再用统一密钥重新加密, 这种“加密-解密-再加密”的方式不仅产生大量的计算冗余, 而且具有较大的数据泄露风险, 不仅如此, 对称加密机制进行密钥更新时需要 A 和 B 同时更新密钥, 这对于分布式、可抛弃、数量众多的 IoT 终端来说尤为困难. Kallahalla 等人[4]提出了 Plutus 加密文件系统, 主要思想是将加密文件分组后放入“保险箱”, 为每个保险箱分配一个密钥. 共享文件时将保险箱密钥分发给用户. Plutus 使用经典的对称密码的思想, 密钥管理的复杂度与保险箱的数量成正比, 不适用于文件数量众多的情况, 并且不支持细粒度的访问控制. Vimercati 等人[5]提出

基于密钥导出的方法, 即为每个用户分发一个密钥 *sk*, 同时向服务器分发一个公共令牌. 用户可以使用公共令牌和密钥 *sk* 导出所需文件的解密密钥. 只拥有令牌的服务器则无法导出解密密钥. 该方案使得文件加密密钥数量和用户密钥数量都达到最小, 但是, 文件创建及用户授权/撤回仍与用户数量线性相关, 影响了该机制的延展性.

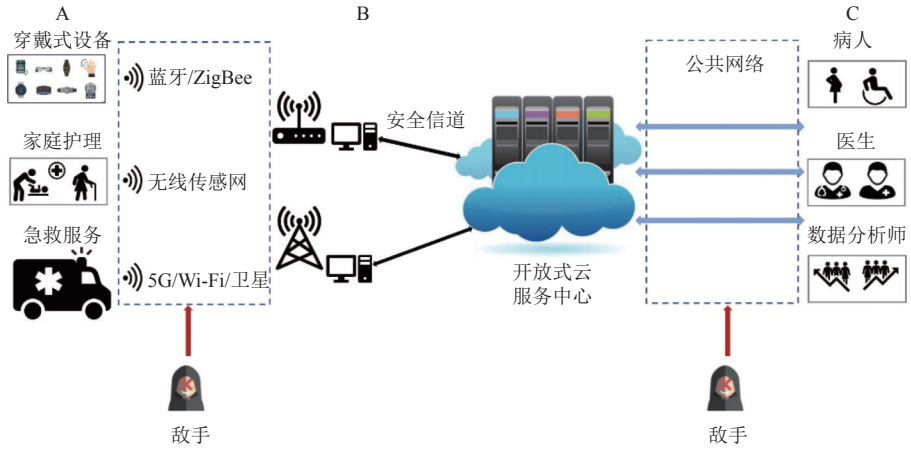


图 1 PHC 系统结构示意图

在用于数据安全共享时, 相比对称密码而言, 公钥密码体制具有更好的便捷性. Goh 等人在文献 [6] 中, 提出了 SiRiUS 系统, 将每个文件绑定一个访问控制列表 (access control list, ACL), 再使用授权用户的公钥加密文件密钥. SiRiUS 的加密复杂度与授权用户的数量成正比. Ateniese 等人[7]中提出了基于 PRE (proxy re-encryption) 的安全分布式存储机制, 数据持有者用对称密钥加密每个文件, 对称密钥再使用主公钥加密, 只有拥有主私钥的数据持有者可解密获取对称密钥. 然后, 数据持有者使用他的主私钥和用户公钥生成代理重加密密钥, 半诚实的服务器使用重加密密钥将主公钥加密的对称密钥转化为指定的授权用户可读的密文. 该机制的主要问题是无法对抗恶意服务器和任何恶意用户的同谋攻击, 并且 PRE 同样需要维护一个 ACL. 属性基加密 (attribute-based encryption, ABE) 可以实现动态多用户的访问控制, 但通用的 ABE 算法通常计算和存储负载较高, 虽然近年来提出了一些高效的 ABE 算法, 但对于资源受限的 IoTS 还是难以适用.

为了弥补单一密码机制的不足, 一些方案对多种密码机制进行组合. Yu 等人[8]提出将 KP-ABE[9]和代理重加密[10]和懒惰重加密[4]相结合应用于云环境数据访问的方法. 该方法使用 KP-ABE 实现细粒度的访问控制, 使用重加密将高复杂度的运算外包至云服务器, 解决了动态访问控制和效率提升问题. 然而, 该方案的实施基于单一的云环境, 并不适用于物联网和云计算相结合的场景.

综合以上情况, PHC 系统的安全性主要面临以下挑战.

(1) 单一的密码体制难以同时兼顾效率和功能, 目前常见的安全数据共享方案大多适用于云平台, 分布式的物联网系统难以同时实现高效的数据加密及细粒度的访问控制.

(2) 低功耗的 IoTS 设备难以支持计算复杂度较高的加密算法, 如以映射运算为基础的数据加密算法.

(3) 分布式 IoTS 设备存在易遗失特性, 一方面存在较大的密钥泄露风险, 另一方面给密钥更新带来较大困难.

(4) 系统用户数量众多动态性强, 并且大量用户使用轻量级设备访问数据, 如何既能实现细粒度的访问控制, 又能让资源受限的用户快速实施访问面临较大的挑战.

为了应对上述问题, 本文致力于 PHC 系统安全数据共享的研究, 主要创新点如下.

(1) 提出了一种适用于云边协同网络的安全数据共享方案, 实现了 PHC 系统中 IoTS 数据的安全上传、个性密文的归一化、动态多用户的细粒度访问控制、高效的解密操作, 并给出了形式化的安全性证明. 理论分析和实验证明, 该方案具有较好的安全性和较强的实用性.

(2) 针对 IoTS 数量众多、分散性强, 密钥更新较为困难的特性, 本文引入代理重加密 (PRE) 机制实现了基于

单方变换, 即无需对 IoTS 终端进行任何操作的密钥更新. 同时, 本文的应用场景中, 重加密方 MEC 可以看做是完全可信的, 与传统 PRE 应用中代理方 (即重加密方) 为不完全可信的服务器不同, 本文据此改进了 Guo 等人[11]提出的经典 PRE 算法, 摒弃了代理和被代理者之间校验、审计等运算, 从而实现了更为高效的加密、重加密和解密能力.

(3) 改进经典的 FAME ABE 算法[12], 针对用户数量众多、动态性强、资源受限的特点, 通过采用基于阈值的用户权限设定、基于解密外包的服务器映射计算, 达成了细粒度的访问控制, 进一步减少了数据泄露的风险, 并支持用户随时随地利用轻量级智能设备访问数据.

1 相关工作

与本文相似的研究方向主要包括密文共享系统以及外包数据的访问控制. PRE 和 ABE 均为用于实现密文共享和访问控制的主流密码技术.

1.1 属性基加密在访问控制中的应用

在数据安全共享系统中, 访问控制是非常重要的安全环节. 许多传统的加密机制 (包括身份基加密机制) 仅能提供粗粒度的访问控制. 2005 年, Sahai 等人在模糊身份基加密机制的基础上首次提出了 ABE 的概念[13], 可以实现细粒度的访问控制和高效的信息分享. 在早期的经典 ABE 机制[9,13−15]中, 当用户的密钥与加密时指定的一些属性或属性运算相匹配时, 才可以执行解密运算. 为了丰富访问控制策略的表达能力, Goyal 等人[13]和 Bethencourt 等人[14]分别提出了基于树的密钥策略和密文策略 ABE 方案. 在密钥策略 ABE 方案 (KP-ABE) 中, 属性集与密文关联, 访问结构与密钥关联, 其中访问结构指定用户将有权使用哪些密文解密. 密文策略 ABE (CP-ABE) 以对偶方式进行, 为每个属性分配一个私钥分片, 并让发送方指定接收方的属性集应遵守的访问策略. CP-ABE 方案目前支持与门基[16]和最具表达能力的线性秘密共享 (linear secret sharing scheme, LSSS) 基的访问结构[17]. 其中, LSSS 基访问结构由 Lewko 等人提出[17], 方案给出了一种利用与或门树的算法, 常规用户可以将任何单调布尔访问公式转换为相应的 LSSS 矩阵.

文献 [18] 提出了将 CP-ABE 应用于云环境中的密文访问控制, 有效解决了动态多用户的访问控制问题. 虽然 ABE 在云存储平台的访问控制方面具有不可比拟的优势, 然而, 在常规的 CP-ABE 方案中, ABE 的计算量通常非常大, 包括许多映射运算和指数运算, 加解密时间往往与访问结构复杂度成线性正相关, 密文及私钥大小同样与属性数量成线性正相关, 因此, 当用户设备资源受限时, 大大限制了它们的应用. 解密外包是提升性能的重要方式. Green 等人[19]在 ABE 系统中引入了外包解密, 将计算代价较高的数据解密运算转移到云端, 对于使用移动设备的用户而言, 大幅降低了用户端的计算代价. Tian 等人[20]提出了一种将计算复杂度较高的映射运算外包至云服务器的方法, 无需对算法做任何改变, 且终端只需执行高效率的模幂和乘法运算, 大幅降低了用户终端的计算负载. 该算法具有较强的实用性. Liu 等人[21]提出了一种用于外包解密的 OABE 方案, 此外, 它还支持属性撤销和策略更新. 在智能医疗系统中, 将加密和解密外包给边缘节点的 OABE 方案比云上的 OABE 方案效率更高.

在云环境的实际应用中, 大部分使用 ABE 的访问控制方案为“要么全有, 要么全无”的方式, 即一旦获得访问权限, 则可以无限制的执行访问, 这种方式存在较大的安全隐患. 为了克服这一问题, Yuen 等人[22]引入了基于有限时间属性的匿名访问控制的概念. Ning 等人[23]提出了一种基于经典的 CP-ABE 方案[24]的有限次数访问控制机制. 这两种方案都适用于限制单个授权用户的下载次数, 但不能任意限制所有授权用户的下载行为.

1.2 代理重加密及其在物联网中的应用

Blaze 等人[25]于 1998 年首次提出 PRE 机制, 利用半诚实的代理将 Alice 的公钥加密的密文转化成 Bob 的私钥可以解密的密文, 转化过程无需实现解密, 代理无法获取任何明文信息. PRE 机制的密文可转移特性引起广泛关注, 被应用于多个领域, 相关的研究方向也包括多种类型, 如基本代理重加密[7,26,27]、基于身份的代理重加密 (IB-PRE)[26−28]、基于属性的代理重加密 (AB-PRE)[29]和无证书代理重加密 (CL-PRE)[30]等. 在安全性能方面, Blaze 的机制[25]可以抵抗选择明文攻击 (CPA), 但不能对抗共谋攻击. Ateniese 等人[7]使用双线性对, 提出了抗共谋攻击的 PRE 方案, 并定义了 PRE 方案的期望属性. Green 等人[26]重点研究了基于 CCA 安全的 PRE 机制, 并提出了相应

的安全模型. Canetti 等人构造出了选择密文安全 (CCA) 安全的 PRE 机制[31]. Libert 等人[32]提出了具有重放选择密文攻击 (re-playable chosen ciphertext attack, RCCA) 安全性的 PRE 方案.

上述研究都适用于传统的云数据共享系统, 但针对物联网环境的应用研究很少. 近年来, 物联网的安全隐私问题成为研究热点, 许多工作开始探索代理重加密技术在物联网中的应用. Yang 等人[33]提出了用于电子病历 (electronic health record, EHR) 系统的代理重加密方案, 用于实现 EHR 系统中定时数据共享和搜索机制. Hong 等人[34]提出了一种用于物联网场景的密钥隔离基于属性的代理重加密 (AB-PRE) 方案. 该方案利用属性基加密算法加密数据, 采用密钥隔离机制保障用户私钥和重加密密钥的前向安全, 从而实现了安全且细粒度的数据共享. 文献 [35] 中, Kim 等人提出了一种在轻量级设备上共享和管理数据的方法. 该方法使用代理重加密以减少数据管理时的加密次数, 并提供数据共享功能, 以补充轻量级设备网络功能不足的问题. Li 等人[36]给出了基于相等测试的代理重加密技术在物联网医疗系统中的安全数据共享方案, 该方案将 PRE 和带相等测试的公钥加密 (public key encryption with equality test, PKE-ET) 的概念结合起来, 得到了具有相等测试的代理重加密 (PRE-ET). 继承 PKE-ET 和 PRE 的优点, 用户可以从不同公钥加密的数据中搜索所需的医疗记录数据. 同时, 在不泄露密钥和明文的情况下, 可以安全灵活地共享搜索到的医疗记录.

2 基础知识

2.1 基本概念

2.1.1 双线性映射

设 G 和 G_T 均为素数 p 阶的乘法循环群, g 为 G 的生成元, 双线性映射 $e: G\times G\to G_T$ 具有下述属性:

(1) 双线性: $\forall u, v\in G$, 以及 $a, b\in\mathbb{Z}_p$, 均有 $e(u^a, v^b)=e(u,v)^{ab}$;

(2) 非退化: $e(g, g)\neq 1$, $\exists g\in G$ 使得 $e(g, g)$ 在 G_T 中的阶为 N;

(3) 可计算: 对于$\forall u, v\in G$, 可以在多项式时间内有效计算 $e(u, v)$.

2.1.2 访问结构及线性秘密共享

访问结构或策略指定了获取访问权限所需的属性集合.

定义 1. 访问结构. 若 $\mathcal{U}$ 表示所有属性的全集, 则访问结构 $\mathbb{A}$ 为关于 $\mathcal{U}$ 的非空子集, 即 $\mathbb{A}\subseteq 2^{\mathcal{U}}\setminus\{0\}$. 若对每个 $B,C\subseteq\mathcal{U}$, 则有 $B\subseteq C$, $B\in\mathbb{A}\Rightarrow C\in\mathbb{A}$, 则称访问结构 $\mathbb{A}$ 为单调的.

常见的访问结构表示形式有布尔公式[14,15]和线性秘密共享 (LSSS)[24]两类. LSSS 可以用单调扩张 (monotone span programs, MSP) 来表示. MSP 由 $\mathbb{Z}_p$ 上的一个 $n_1\times n_2$ 阶的矩阵 M 和映射 $\pi: \{1,\ldots,n_1\}\to\mathcal{U}$ 给出. 文献 [37] 给出了一种简单高效的方法, 可以将布尔公式 F 转化为 MSP(M, π), 使得 M 中的每一行与 F 中的一个输入相对应, 并且 M 的列数等于 F 中与门的个数. 进一步讲, M 的每个分量为 0, 1 或−1.

假设 S 为属性集合, $I=\{i|i\in\{1,\ldots,n_1\},\pi(i)\in S\}$ 为矩阵 M 中属于 S 的行号集合. 若存在 I 中行的线性组合计算得到 (1, 0, …, 0), 我们称 (M,π) 接受 S. 形式上讲, 存在系数 $\{\gamma_i\}_{i\in I}$, 满足:

$$\sum_{i\in I}\gamma_i(M)_i=(1,0,\ldots,0) \tag{1}$$

其中, $(M)_i$ 是 M 的第 i 行.

引理 1. 若 MSP(M, π) 不满足属性集合 S, 则存在一个向量 w, 其第 1 个分量不为 0, 并且$\forall i$, 满足$\pi(i)\in S$ 时, $<w,(M)_i>=0$.

2.2 复杂度假设

2.2.1 DBDH 假设

DBDH (decisional bilinear Diffie-Hellman) 问题: 定义五元组 $<g,g^a,g^b,g^c,e(g,g)^{abc}>$, 其中, $a,b,c\leftarrow\mathbb{Z}_p$ 为随机均匀选取. 随机选取 $t\leftarrow\mathbb{Z}_p$, 使用算法 B 判定 $e(g,g)^{abc}=e(g,g)^t$ 是否成立, 若成立则输出 1, 否则输出 0. 定义算法 B 的优势为:

$$Adv_B^{\mathrm{DBDH}}(\lambda)=|\Pr[B(g,g^a,g^b,g^c,e(g,g)^{abc})=1]-\Pr[B(g,g^a,g^b,g^c,e(g,g)^t)=1]|.$$

定义 DBDH 假设为: 若对于任意的概率多项式时间算法 B, 其优势 $Adv_B^{\mathrm{DBDH}}(\lambda)$ 关于 λ 是可忽略的, 则称 DBDH 假设成立.

2.2.2 q-BDHI 判定假设

q-BDHI (q-Bilinear Diffie-Hellman inversion) 判定问题: 给定数组 $\vec{y}=(g,g^x,\ldots,g^{x^q})$ 为输入, 是否可以将 $e(g,g)^{1/x}$ 与群 G_T 中的随机数区别开来.

形式上讲, 若 $|\Pr[A(\vec{y},e(g,g)^{1/x})=0]-\Pr[A(\vec{y},R)=0]|\geqslant\varepsilon$, 则算法 A 解决上述 q-BDHI 判决问题时存在不可忽略优势 ϵ, 其中, $x\in\mathbb{Z}_p^*$, $R\in G_T$. 若所有 PPT 算法解决 q-BDHI 判定问题时仅存在可忽略的优势, 则 q-BDHI 判定假设成立.

2.3 端加密和重加密

本文提出了端加密和重加密的概念, 这里给出形式化的定义.

2.3.1 端加密

端加密即 IoTS 设备对采集的数据进行加密, 这里给出形式化的定义, 共包含下列 4 个算法.

1) $Setup_1(1^\lambda)$. 算法以安全参数 λ 作为输入, 输出系统公开参数 *param*.

2) *KeyGen*(*param*). 算法以公共参数 *param* 为输入, 输出公私密钥对 (sk_i, pk_i), $i=1,\ldots,n$. 其中 n 为 IoTS 设备的数量.

3) *Encrypt*(pk_i, m). 算法以公钥 pk_i 和消息 m 为输入, 输出初始密文 CT_i.

4) *Decrypt*(CT_i, sk_i). 算法以私钥 sk_i 以及密文 CT_i 为输入, 输出明文消息 m. 值得注意的是, 密文 CT_i 在 PHC 系统中无需解密, 我们给出 *Decrypt*(CT_i, sk_i) 的定义是为了保持端加密算法定义的完整性.

2.3.2 重加密

重加密即 MEC 对来自 IoTS 的初始密文进行二次加密, 以得到归一化的密文. 这里我们给出重加密的形式化定义, 共包含下列 5 个算法.

1) $Setup_1(1^\lambda)$. 同端加密中的 $Setup_1(1^\lambda)$.

2) *KeyGen*(*param*). 算法以公共参数 *param* 为输入, 输出独特公私密钥对 (sk_p, pk_p).

3) *ReKeyGen*(sk_i, pk_p). 算法以端加密中生成的私钥 sk_i 以及独特公钥 pk_p 为输入, 输出重加密密钥 $rk_{i\to p}$.

4) *ReEnc*($rk_{i\to p}$, CT_i). 算法以初始密文 CT_i 和重加密密钥 $rk_{i\to p}$ 为输入, 输出归一化密文 *CT*.

5) *Decrypt*(*CT*, sk_p). 算法以独特私钥 sk_p 以及密文 *CT* 为输入, 输出明文消息 m.

6) *KeyUpdate*(*param*, sk_i). 算法以公共参数 *param* 以及私钥 sk_i 为输入, 输出更新的独特私钥 sk'_p 以及更新的重加密密钥 $rk'_{i\to p}$.

3 系统架构及安全模型

3.1 系统框架

PHC 系统包括体征感知器群组, 本地诊所, 在线服务器, 数据用户, 管理中心 5 个模块. 与此相对应, 其网络架构包含 5 部分: 资源受限的 IoTS 群组、移动边缘计算中心 (mobile edge compute, MEC)、半可信的云服务器 (semi-trusted cloud, STC)、用户智能设备 (user smart device, USD) 和认证授权中心 (authority and authentication, AA). 各部分具体功能如下.

(1) IoTS 群组. IoTS 主要用于病人体征数据的采集及加密上传.

(2) 移动边缘计算中心 (MEC). MEC 为部署在边缘节点的小型服务器, 具有较强的计算和存储资源, 主要用于将个性化的 IoTS 密文归一化为统一密钥加密的共享密文, 并上传至云服务器.

(3) 半可信的云服务器 (STC). STC 采用“诚实但好奇”的模型, 具有丰富的存储和计算资源, 实现数据的存储和管理.

(4) 用户智能设备 (USD). 用户通过便携式智能终端, 如智能手机、平板, 实现数据的访问. 每个用户可用一组

属性集合 (例如: 城市: 上海, 国家: 中国, 身份: 科室主任, 专业: 呼吸内科) 表示.

(5) 认证授权中心 (AA), 完全可信的机构, 实现密钥管理和用户认证授权等.

PHC 系统的安全数据共享架构如图 2 所示, 数据从采集到共享至用户需经“IoTS→MEC→STC→USD”几个环节, 下面我们详细描述系统工作流程.

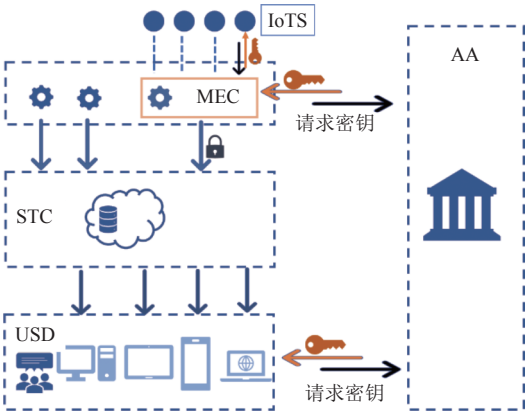


图 2 PHC 系统安全数据共享网络架构

步骤 1: 假设 IoTS 设备的数量为 n, 首先, AA 生成 n 个不同的公私密钥对 (sk_i, pk_i), i=1,…, n, 将每个公钥 pk_i 分发至 IoTS 设备. 随后, AA 生成 1 个独特的公私密钥对 (sk_p, pk_p), 用每个 sk_i 和 pk_p 生成重加密密钥 $rk_{i\to p}$, 将重加密密钥 $rk_{i\to p}$ 分发至 MEC. 此外, 对每个 USD, AA 根据其属性集合 S' 为其生成相关联的属性密钥 sk', 并发送给 USD.

步骤 2: 每个 IoTS 设备使用公钥 pk_i 加密采集的数据, 得到个性化密文 CT_i, 并上传至 MEC.

步骤 3: MEC 使用重加密密钥 $rk_{i\to p}$ 对接收的个性化密文 CT_i 重加密, 得到可用统一私钥 sk_p 进行解密的归一化密文 CT, 然后上传至云服务器.

步骤 4: STC 为用户设置访问次数上限 σ, 同时设置访问结构 (M, π), 将 σ 和 (M, π) 发送至 AA. AA 据此利用本文提出的 TO-FAME 算法将私钥 sk_p 加密为密文 ct, 然后将 ct 发送至云服务器.

步骤 5: 用户需要访问数据时, 首先由 STC 验证访问次数是否超过阈值, 若超过, 则拒绝访问. 反之, STC 将归一化密文 CT 和加密密钥 ct 发送至用户, 用户利用持有的属性密钥 sk' 解密 ct, 若属性密钥满足 (M, π), 则获取正确的 sk_p, 从而可以解密 CT.

3.2 设计目标

根据 PHC 系统的工作流程, 我们提出了其安全数据共享系统的设计目标, 主要包括如下 5 个方面.

● 数据的机密性. 数据机密性包含端加密的机密性和重加密的机密性 2 个方面, 我们分别定义为端加密的选择明文 (CPA) 安全和重加密的选择明文 (CPA) 安全.

1) 端加密的 CPA 安全. IoTS 终端采集的数据经无线通信信道上传, 容易被监听, 因此需要设计 CPA 安全的端加密机制, 以确保监听类型的敌手无法获得关于明文的消息.

2) 重加密的 CPA 安全. MEC 对端密文进行重加密, 生成归一化的共享密文, 然后外包至 STC 进行存储管理. 因为 STC 定义为“诚实但好奇”的模型, 对重加密密文的威胁同样可看作监听模式. 因此, 重加密机制同样需要满足 CPA 安全性.

● 细粒度的访问控制. 由于 PHC 系统用户数量众多, 并且涉及多类人群, 需要设定细粒度的访问控制权限, 以防止非法用户获得数据访问权限.

● 可控次数访问. 目前常用的访问控制方式通常为“要么全有, 要么全无”的方式, 即一旦用户获取授权, 则可以无限期地访问数据, 这对于动态性较强的用户群体来说, 无疑具有较高的数据泄露风险. 因此, 系统需要提前设定访问次数的阈值上限, 达成“限定 σ -次访问”的安全目标.

● 授权解密的高效性. 由于 USD 通常为便携式的移动设备, 其计算和存储资源有限, 必须尽可能减小 USD 的计算和存储开销, 在不减弱 USD 其他性能的基础上进行数据的访问.

● 基于单方变换的密钥更新. IoTS 数量众多, 且通常分布式的部署在偏远地区, 使用常规的公钥加密机制, 密钥更新时需要双方同时变换密钥, 困难性很大. 因此, 亟需一种基于单方变换的密钥更新方式, 在不改变 IoTS 密钥的情况下, 达成密钥的有效更新.

3.3 算法定义

根据 PHC 系统物联网和云平台相结合的网络架构特点, 其加密流程包含 3 部分: 一是对 IoTS 数据进行端加密; 二是 MEC 对端加密的密文进行重加密; 三是对解密密钥进行 CP-ABE 加密. 端加密算法和重加密算法均包含在 IPRE 算法中, 对密钥的加密由 TO-FAME 算法实现. 下面我们给出 IPRE 算法和 TO-FAME 算法的形式化定义.

3.3.1 IPRE 算法定义

IPRE 算法的主要功能是加密 IoTS 采集的数据并实现密文的归一化, 该算法主要包括 4 个功能模块: 一是系统建立及密钥管理, 包括密钥生成和密钥更新, 由 AA 执行; 二是端加密, 由 IoTS 执行; 三是重加密, 由 MEC 执行; 四是解密, 由合法的 USD 执行. IPRE 包含 7 个具体算法, 分别为 $Setup_1$, $KeyGen_1$, *ReKeyGen*, $Encrypt_1$, *Re-encrypt*, $Decrypt_1$, *KeyUpdate*. 下面给出形式化的定义.

1) $Setup_1(1^{\lambda})$. 该算法由 AA 执行, 以安全参数 λ 作为输入, 输出系统公开参数 *param*.

2) $KeyGen_1(param)$. 该算法由 AA 执行, 以系统公开参数 *param* 为输入, 输出公私密钥对 $(sk_i, pk_i), i = 1, \ldots, n$, 以及独特密钥对 (sk_p, pk_p). 其中, n 为 IoTS 的数量, 每个 pk_i 被分发至第 i 个 IoTS 设备, pk_p 被分发至 MEC.

3) $RekeyGen(sk_i, pk_p)$. 该算法由 AA 执行, 以 IoTS 的私钥 sk_i 和独特公钥 pk_p 为输入, 输出重加密密钥 $rk_{i\to p}$, 并将其分发至 MEC.

4) $Encrypt_1(pk_i, m)$. 该算法由 IoTS 执行, 以 IoTS 的公钥 pk_i 及明文 m 为输入, 输出初始密文 CT_i.

5) $ReEnc(rk_{i\to p}, CT_i)$. 该算法由 MEC 执行, 以重加密密钥 $rk_{i\to p}$ 和初始密文 CT_i 为输入, 输出重加密的密文 *CT*.

6) $Decrypt_1(CT, sk_p)$. 该算法由合法的 USD 执行, 以私钥 sk_p 以及密文 *CT* 为输入, 输出明文消息 *m*.

7) $KeyUpdate(param, sk_i)$. 该算法由 AA 执行, 以公共参数 *param* 以及私钥 sk_i 为输入, 输出更新的独特私钥 sk'_p 以及更新的重加密密钥 $rk'_{i\to p}$.

3.3.2 TO-FAME 算法定义

TO-FAME 算法的主要功能是对 IPRE 算法中的私钥 sk_p 进行属性基加密, 从而实现细粒度的访问控制. 该算法主要包含 4 个功能模块: 系统建立和密钥生成, 加密, 校验, 解密. 其中, AA, STC 和 USD 分别执行相应的系统建立, 并分别生成相关的密钥, 加密由 AA 执行, 校验由 STC 执行, 解密由合法的 USD 执行. 此外, 为了提高计算效率, 我们可以将该算法中复杂度较高的映射运算进行计算外包, 但由于计算外包为可选择服务, 因此并未将外包服务器列为系统角色. TO-FAME 共包含 9 个具体算法, 分别为 $Setup_2$, $Setup_c$, $Setup_u$, $KeyGen_2$, *VerkeyGen*, $Encrypt_2$, *Verify*, $Decrypt_2$, $Decrypt_{out}$. 各算法定义如下.

1) $Setup_2(1^{\lambda})$. 该算法由 AA 执行, 以安全参数 λ 作为输入, 输出公共参数 *pp*, 公共密钥 *pk* 和主密钥 *msk*.

2) $Setup_c()$. 该算法由 STC 执行, 初始化一个计数变量 *ctr*=0, 以及一个存储密钥的空集 *ST*.

3) $Setup_u(pp)$. 该算法由 USD 执行, 以公共参数 *pp* 作为输入, 输出用户的公私密钥 (sk_u, pk_u).

4) $KeyGen_2(msk, S)$. 该算法由 AA 执行, 输入系统主密钥 *msk* 和用户属性集合 *S*, 输出属性密钥 *sk*.

5) $VerkeyGen(sk_u, ctr, pk)$. 该算法由 USD 执行, 以用户私钥 sk_u, 计数器 *ctr* 以及公共密钥 *pk* 为输入, 输出校验密钥 $vkey$.

6) $Encrypt_2(pk, (M,\pi), m)$. 该算法由 AA 执行, 以密钥 *pk*, 访问结构 (M, π) 以及明文 *m* 为输入, 输出密文 *ct*.

7) $Verify(Z_u, vkey, pk)$. 该算法由 STC 执行, 输入用户公钥 Z_u, 校验密钥 *vkey*, 以及公共密钥 *pk*, 输出校验结果.

8) $Decrypt_2(pk, ct, sk)$. 该算法由 USD 执行, 输入公共密钥 *pk*, 密文 *ct*, 属性密钥 *sk*, 输出 $\perp$ 或明文消息 *m*.

9) $Decrypt_{out}(pk, ct, sk)$. 该算法由 USD 和外包服务器共同执行, 输入公共密钥 *pk*, 密文 *ct*, 属性密钥 *sk*, 输出 $\perp$

或明文消息 m.

3.4 安全模型

3.4.1 数据的机密性

定义 2. 数据的机密性. 数据的机密性包括端加密的机密性和重加密的机密性两部分, 分别定义为端加密的选择明文 (CPA) 安全和重加密的选择明文 (CPA) 安全.

为了描述 CPA 安全的概念, 我们引入了关于挑战者 C 和敌手 A 之间的游戏, 并假设了一个不允许敌手适应性地攻陷用户的静态模型[27,28]. 敌手 A 可以执行下列询问:

● $Q_{hkg}(i)$: 未被攻陷密钥的生成询问. 敌手 A 向挑战者询问密钥. 挑战者 C 运行 $KeyGen_1$ 算法, 计算得到 (pk_i, sk_i), 将 pk_i 返回给敌手 A.

● $Q_{ckg}(i)$: 被攻陷密钥的生成询问. 敌手 A 向挑战者询问密钥. 挑战者 C 运行 $KeyGen_1$ 算法, 计算得到 (pk_i, sk_i), 将其返回给敌手 A.

● $Q_{rkg}(pk_i, pk_j)$: 重加密密钥的生成询问. 敌手 A 向挑战者询问重加密密钥. 挑战者 C 输入 (pk_i, pk_j), 运行 $RekeyGen$ 算法生成重加密密钥 $rk_{i\to j}$, 然后将 $rk_{i\to j}$ 返回敌手 A. 其中 pk_i, pk_j 通过上述询问获取.

定义 3. 端加密的 CPA 安全[11]. 对任意的端加密算法$\prod$, 我们利用挑战者 C 和敌手 A 之间的游戏, 形式化的描述端加密 CPA 安全的概念. 游戏包括以下几个阶段.

1) 初始化阶段. 挑战者 C 运行 $Setup_1(1^\lambda)$ 算法得到系统参数 $param$, 并发送给敌手 A.

2) 询问阶段 1. 敌手 A 可以执行下列询问.

● 询问 $Q_{hkg}(i)$, 挑战者 C 返回公钥 pk_i.

● 询问 $Q_{ckg}(i)$, 挑战者 C 返回公私密钥对 (pk_i, sk_i).

● 询问 $Q_{rkg}(pk_i, pk_j)$, 若公钥 pk_j 已被攻陷, 返回 $\perp$. 否则, 挑战者 C 返回一个重加密密钥 $rk_{i\to j}$.

3) 挑战阶段. 在该阶段, 敌手 A 选择长度相同的两个消息 m_0 和 m_1, 通过询问获取参数 pk^*, 将 m_0, m_1 和 pk^*发送给挑战者 C. C 随机选择 $d\leftarrow\{0, 1\}$, 加密 m_d 生成挑战密文 CT^*. 然后, 挑战者 C 将挑战密文 CT^*返回敌手 A.

4) 询问阶段 2. 敌手 A 执行和阶段 1 同样的询问.

5) 猜测阶段. 敌手 A 输出猜测的比特 d'. 若 $d'=d$, 则称 A 获胜.

设置敌手 A 在上述游戏中获胜的优势是 $\Pr[d'=d]-1/2$. 如果对于任意多项式时间 (PPT) 的敌手 A, 对于足够大的 n, 有$|\Pr[d'=d]-1/2|\leqslant 1/poly(n)$, 则称方案$\prod$为 CPA 安全的. 其中 $poly$ 表示多项式函数.

定义 4. 重加密的 CPA 安全[11]. 对任意的重加密算法$\prod$, 我们利用挑战者 C 和敌手 A 之间的游戏, 形式化地描述重加密 CPA 安全的概念. 游戏包括以下几个阶段.

1) 初始化阶段. 挑战者 C 运行 $Setup_1(1^\lambda)$ 算法, 并将系统的公开参数 $param$ 发送给敌手 A.

2) 询问阶段 1. 敌手 A 可以执行下列询问.

● 和端加密相同的询问 $Q_{hkg}(i)$ 和 $Q_{ckg}(i)$.

● 询问 $Q_{rkg}(pk_i, pk_j)$, 挑战者 C 返回一个重加密密钥 $rk_{i\to j}$.

3) 挑战阶段. 在该阶段, 敌手 A 选择长度相同的两个消息 m_0 和 m_1, 通过询问获取参数 pk^*, 将 m_0, m_1 和 pk^*发送给挑战者 C. C 随机选择 $d\leftarrow\{0, 1\}$, 加密 m_d 生成挑战密文 CT^*. 然后, 挑战者 C 将挑战密文 CT^*返回敌手 A.

4) 询问阶段 2. 敌手 A 执行和阶段 1 同样的询问.

5) 猜测阶段. 敌手 A 输出猜测的比特 d'. 若 $d'=d$, 则称 A 获胜.

设置敌手 A 在上述游戏中获胜的优势是 $\Pr[d'=d]-1/2$. 如果对于任意多项式时间的敌手 A, 对于足够大的 n, 有$|\Pr[d'=d]-1/2|\leqslant 1/poly(n)$, 则称重加密算法$\prod$为 CPA 安全的. 其中 $poly$ 表示多项式函数.

3.4.2 细粒度的访问控制

定义 5. 细粒度的访问控制的适应性安全. 对任意的 CP-ABE 方案$\prod$, 设 λ 为安全参数, 消息 m_0 与 m_1 长度相等, $b\in\{0, 1\}$. 为了形式化的定义 ABE 方案$\prod$的适应性安全, 我们描述了关于敌手 A 和挑战者 C 之间的游戏, 其

中挑战者 C 可以获取安全参数 λ 和 b 的内容, 而敌手 A 只能获得 λ 的内容. 游戏包括以下几个阶段.

1) 系统建立阶段. 挑战者 C 运行$\prod$中的 $Setup_2(1^{\lambda})$ 算法, 该算法以安全参数 λ 作为输入, 输出公钥 pk 和主密钥 msk. 挑战者 C 将公钥 pk 发送至敌手 A.

2) 询问阶段 1. 敌手 A 可以执行以下几种询问.

● 询问 $KeyGen_1$. 敌手 A 选取属性集合 S, 并发送至挑战者 C. 挑战者 C 运行 $KeyGen_1(msk, S)$, 生成属性密钥 sk^*, 并将 sk^* 返回至 A.

● 询问 $VerkeyGen$. 敌手 A 向挑战者 C 询问校验密钥, 挑战者 C 运行 $VerkeyGen$ 算法得到校验密钥 $Vkey^*$, 并返回至敌手 A.

3) 挑战阶段. 敌手 A 向挑战者 C 提交两个等长的消息 m_0, m_1, 以及 $(M,\pi)^*$. 挑战者 C 随机选取消息 m_b, $b\in\{0, 1\}$, 运行 $Encrypt_2(pk, (M,\pi)^*, m_b)$ 算法得到密文 ct^*, 并返回至敌手 A.

4) 询问阶段 2. 敌手 A 执行和阶段 1 同样的询问.

5) 猜测阶段. 敌手 A 输出猜测的比特 b'. 若 $b'=b$, 则称 A 获胜.

上述游戏要求对敌手 A 用于询问的每个属性集 S, 都有 S 不满足 $(M, \pi)^*$. 设敌手 A 在上述游戏中获胜的优势是 $\Pr[d'=d]-1/2$. 如果对于任意多项式时间的敌手 A, 对于足够大的 n, 有$|\Pr[d'=d]-1/2|\leqslant 1/poly(n)$, 则称方案$\prod$为适应性安全的. 其中 $poly$ 表示多项式函数.

3.4.3 限定 σ -次访问

定义 6. 限定 σ -次访问. 限定 σ -次访问确保只有 σ 次解密权限的数据用户无法执行第 $\sigma+1$ 次解密. 我们借助下述挑战者 C 和敌手 A 之间的安全游戏进行形式化的描述.

1) 初始化阶段. 挑战者 C 运行 $Setup_2$ 算法得到密钥 (pk, msk), 并将 pk 发送至敌手 A.

2) 询问阶段. 敌手 A 可以发起与第 3.4.2 节相同的询问.

3) 挑战阶段. 假设这是第 j 次加密请求. 敌手 A 向挑战者 C 提交访问结构 $(M,\pi)^*$. 挑战者 C 执行 $Encrypt_2(pk, (M,\pi)^*, m_b)$ 加密算法得到密文 ct^*, 并返回至敌手 A. 敌手 A 运行 $Decrypt_2(pk, ct^*, sk^*)$ 解密算法.

4) 输出阶段. 敌手 A 输出解密后的消息 m.

上述游戏中, 若 $j=\sigma+1$, 则 A 赢得游戏. 设敌手 A 在上述游戏中获胜的优势是 $\Pr[j=\sigma+1]-1/2$, 如果对于任意多项式时间的敌手 A, 对于足够大的 n, 有 $|\Pr[j=\sigma+1]-1/2|\leqslant 1/poly(n)$, 则称该方案为限定 σ -次数访问的. 其中 $poly$ 表示多项式函数.

4 方案实施

本节介绍整体方案的实施细节, 因为 IPRE 算法由端加密和重加密两个算法组合而成, 前面已给出形式化的定义, 这里重点介绍端加密、重加密和 TO-FAME 这 3 个算法. 下面分别给出每个算法的具体描述.

4.1 端加密算法

端加密算法主要包含以下 4 个算法.

● $Setup_1(1^{\lambda})$. 设定 λ 为安全参数, G 和 G_T 均为素数 p 阶群, 其中 $p\geqslant 2^{\lambda}$, $e: G\times G\to G_T$ 为双线性映射. 随机选取 $g_1, g_2, h_1, h_2\leftarrow G$ 并计算 $L=e(h_1, h_2)$. 设置系统参数为:

$$param=(g_1, g_2, h_1, h_2, L).$$

● $KeyGen_1(param)$. 均匀随机地选取 $x_i, y_i\leftarrow\mathbb{Z}_p$ 并计算:

$$(X_i, Y_i)=(h_1^{x_i}, g_1^{y_i}).$$

将 $sk_i=(x_i, y_i)$ 设置为私钥, $pk_i=(X_i, Y_i)$ 设置为公钥.

● $Encrypt_1(pk_i, m)$. 给定输入 X_i 及 m, 均匀并随机地选取 $r\leftarrow\mathbb{Z}_p$, 计算:

$$c_0=L^r\cdot m, c_1=g_1^r, c_2=e(h_1, g_2)^r, c_3=X_i^r.$$

设置 $CT_i=(c_0, c_1, c_2, c_3)$ 作为初始密文.

● $Decrypt_1(sk_i, CT_i)$. 给定 sk_i 以及 CT_i 作为输入, 计算消息:

$$m = c_0/e(c_3,h_2)^{1/x_i}.$$

下面给出算法的正确性描述.

对密文 $CT_i=(c_0,c_1,c_2,c_3)$ 的解密是正确的, 由于:

$$e(c_3,h_2)=e(X_i^r,h_2)=e(h_1^{x_i\cdot r},h_2)=L^{x_i\cdot r},$$

进而有:

$$c_0/e(c_3,h_2)^{1/x_i}=L^r\cdot m/(L^{x_i\cdot r})^{1/x_i}=m.$$

4.2 重加密算法

重加密算法需要在端加密的基础上实现, 主要包含以下 5 个算法.

● $Setup_1(1^\lambda)$. 同端加密中的 $Setup_1(1^\lambda)$.

● $KeyGen_1(param)$. 均匀随机地选取 $x_p,y_p\leftarrow\mathbb{Z}_p$ 并计算:

$$(X_p,Y_p)=(h_1^{x_p},g_1^{y_p}),$$

将 $sk_p=(x_p,y_p)$ 设置为私钥, $pk_p=(X_p,Y_p)$ 设置为公钥.

● $ReKeyGen(sk_i, Y_p)$. 将 sk_i, Y_p 作为输入, 其中 Y_p 为公钥 pk_p 中的参数, 计算:

$$W=(h_2Y_pg_2)^{1/x_i},$$

将 $rk_{i\to p}=W$ 设置为重加密密钥.

● $Re\text{-}encrypt(rk_{i\to p},CT_i)$. 给定输入 $rk_{i\to p}$ 以及端加密的密文 CT_i, 计算:

$$c_0'=c_0,c_1'=c_1,c_2'=e(c_3,W)/c_2,$$

设置 $CT=(c_0',c_1',c_2')$ 作为重加密的密文.

● $Decrypt(CT, sk_p)$. 给定 CT 作为输入, 计算消息:

$$m=c_0'\cdot e(h_1,c_1')^{y_j}/c_2'.$$

● $KeyUpdate(param, sk_i)$. 密钥更新过程只需要更新 MEC 上的密钥, 无需对 IoTS 的密钥做任何改变. 均匀随机地选取 $x_p',y_p'\leftarrow\mathbb{Z}_p$, 计算 $X_p'=h_1^{x_p'},Y_p'=g_1^{y_p'}$, 将 $sk_p'=(x_p',y_p')$ 设置为私钥, $pk_p'=(X_p',Y_p')$ 设置为公钥. 计算:

$$W'=(h_2Y_p'g_2)^{1/x_i},$$

将 $rk_{i\to p}'=W'$ 设置为重加密密钥. 加密和解密过程不变.

下面给出算法解密的正确性描述.

对密文 $CT'=(c_0',c_1',c_2')$ 的解密是正确的, 由于:

$$c_2'=e(c_3,W)/c_2=e(X_i^r,(h_2Y_jg_2)^{1/x_i})/e(h_1,g_2)^r=L^r\cdot e(h_1,Y_j)^r,$$

进而有:

$$c_0'\cdot e(h_1,c_1')^{y_j}/c_2'=L^r\cdot m\cdot\frac{e(h_1,g_1^r)^{y_j}}{L^r\cdot e(h_1,g_1^{y_j})^r}=m.$$

4.3 TO-FAME 算法

TO-FAME 算法是 FAME[12]的改进, 在原来的基础上, 增加了 $Setup_u$, $Setup_c$, $VkeyGen$, $Verify$, $Decrypt_{\text{out}}$ 共 5 个算法, 用以实现用户访问次数的检验和解密外包功能. 为了介绍该算法, 我们首先给出两个形式化的定义. 参照 FAME[12]中的描述, 该机制使用了抗碰撞攻击的 Hash 函数 H, 该函数可以将任意字符串映射成群 G 中的元素. 针对该函数有两种类型的输入: (x, ℓ, t) 形式或者 (j, ℓ, t) 形式, 其中, x 为字符串, j 为正整数, $\ell\in\{1, 2, 3\}$且 $t\in\{1, 2\}$. 为了简化表示, 我们将这两类输入分别表示为 $x\ell t$ 和 $0j\ell t$, 在第 2 种形式的开始附加 0, 以表示与第 1 种的区别. 我们假定所有输入经过适当的编码, 因此没有两个数组产生碰撞.

4.3.1 算法描述

TO-FAME 算法包含以下几个具体算法.

● $Setup_2(1^\lambda)$. 设定 λ 为安全参数, 生成公共参数 pp=(p, G, H, G_T, e, g, h), 选取抗碰撞攻击的 Hash 函数

$\mathcal{H}\{0,1\}\to\mathbb{Z}_p$, 计算 $E=e(g,h)$. 随机选取 $a_1,a_2\leftarrow_R\mathbb{Z}_p^*$, 以及 $d_1,d_2,d_3\leftarrow_R\mathbb{Z}_p$. 输出:

$$(h,E,H_1:=h^{a_1},H_2:=h^{a_2},T_1:=e(g,h)^{d_1a_1+d_3},T_2:=e(g,h)^{d_2a_2+d_3}).$$

作为公共密钥 pk. 选取 $b_1,b_2\leftarrow_R\mathbb{Z}_p^*$, 输出:

$$(g,h,a_1,a_2,b_1,b_2,g^{d_1},g^{d_2},g^{d_3}),$$

做为主密钥 msk.

● $Setup_c(1^\lambda)$. 该算法主要功能是实现 STC 的系统建立. 初始化一个解密运算计数变量 ctr=0, 以及一个空集 ST, 用以存放将要接收的密钥. STC 为一个包含每个用户 ctr 和 ST 的列表 L.

● $Setup_u(pp)$. 算法选取随机数 $z_u\leftarrow\mathbb{Z}_p$, 计算用户公钥 $Z_u=g^{z_u}$, 输出用户公私密钥对 $(pk_u=Z_u,sk_u=z_u)$.

● $KeyGen_2(msk,S)$. 选取 $r_1,r_2\leftarrow_R\mathbb{Z}_p$, 利用 msk 中的 h, b_1, r_1, 计算 $sk_0:=(h^{b_1r_1},h^{b_2r_2},h^{r_1+r_2})$. 对所有的 $y\in S$, 以及 t=1, 2, 计算:

$$sk_{y,t}:=\mathcal{H}(y1t)^{\frac{b_1r_1}{a_t}}\cdot\mathcal{H}(y2t)^{\frac{b_2r_2}{a_t}}\cdot\mathcal{H}(y3t)^{\frac{r_1+r_2}{a_t}}\cdot g^{\frac{\sigma_y}{a_t}},$$

其中, $\sigma_y\leftarrow_R\mathbb{Z}_p$.

设 $sk_y:=(sk_{y,1},sk_{y,2},g^{-\sigma_y})$, t=1, 2 时, 计算:

$$sk_t':=g^{d_t}\cdot\mathcal{H}(011t)^{\frac{b_1r_1}{a_t}}\cdot\mathcal{H}(012t)^{\frac{b_2r_2}{a_t}}\cdot\mathcal{H}(013t)^{\frac{r_1+r_2}{a_t}}\cdot g^{\frac{\sigma'}{a_t}},$$

其中, $\sigma'\leftarrow_R\mathbb{Z}_p$. 设 $sk'=(sk_1',sk_2',g^{d_3}\cdot g^{-\sigma'})$, 输出 $(sk_0,\{sk_y\}_{y\in S'},sk')$ 作为属性密钥 sk.

● $VerkeyGen(sk_u,ctr,pk)$. 用户计算 $K_c=E^{1/(z_u+H(ctr))},K_p=h^{1/(z_u+H(ctr))}$, 向 STC 返回检验密钥 $Vkey=(K_c,K_p,ctr)$.

● $Encrypt_2(pk,(M,\pi),msg)$. 选取 $s_1,s_2\leftarrow_R\mathbb{Z}_p$, 利用 pk 计算:

$$ct_0:=(H_1^{s_1},H_2^{s_2},h^{s_1+s_2}).$$

假设矩阵 M 有 n_1 行 n_2 列, 则对 i=1, …, n_1 以及 l=1, 2, 3, 计算:

$$ct_{i,l}:=\mathcal{H}(\pi(i)l1)^{s_1}\cdot\mathcal{H}(\pi(i)l2)^{s_2}\cdot\prod\nolimits_{j=1}^{n_2}[\mathcal{H}(0jl1)^{s_1}\cdot\mathcal{H}(0jl2)^{s_2}]^{(M)_{i,j}},$$

其中, $(M)_{i,j}$ 表示 M 的第 (i,j) 个元素. 设置 $ct_i:=(ct_{i,1},ct_{i,2},ct_{i,3})$, 同时, 计算:

$$ct':=T_1^{s_1}\cdot T_2^{s_2}\cdot msg.$$

输出 $(ct_0,ct_1,\ldots,ct_{n_1},ct')$ 作为密文 ct.

● $Verify(Z_u,vkey,pk)$. STC 从维护的列表 L 中取出与用户相关的数组 (ctr,ST), 并执行下述操作:

1) 校验下列条件是否满足:

① $e\left(g^{H(ctr)}\cdot Z_u,K_p\right)=E$, 并且 $K_c=e(g,K_p)$;

② $ctr+1\leqslant\sigma$, 其中 σ 为密钥更新时的最大数值;

③ $K_c\notin ST$.

如果不能全满足, 则输出 $\perp$. 否则, 转至 2).

2) 更新 $ctr\leftarrow ctr+1$ 并将 K_c 存储至 ST 备用.

● $Decrypt_2(sk,pk,ct)$. 输入用户属性密钥 sk, 公钥 pk 以及密文 ct, 若 sk 中的属性集 S 不满足密文中的 MSP (M,π), 则算法输出 $\perp$; 否则, 存在满足公式 (1) 的常数 $\{\gamma_i\}_{i\in I}$. 计算:

$$num:=ct'\cdot e\left(\prod_{i\in I}ct_{i,1}^{\gamma i},sk_{0,1}\right)\cdot e\left(\prod_{i\in I}ct_{i,2}^{\gamma i},sk_{0,2}\right)\cdot e\left(\prod_{i\in I}ct_{i,3}^{\gamma i},sk_{0,3}\right),$$

$$den:=e\left(sk_1'\cdot\prod_{i\in I}sk_{\pi(i),1}^{\gamma_i},ct_{0,1}\right)\cdot e\left(sk_2'\cdot\prod_{i\in I}sk_{\pi(i),2}^{\gamma_i},ct_{0,2}\right)\cdot e\left(sk_3'\cdot\prod_{i\in I}sk_{\pi(i),3}^{\gamma_i},ct_{0,3}\right).$$

输出 $m=num/den$, 此处 $sk_{0,1}$, $sk_{0,2}$, $sk_{0,3}$ 分别表示 sk_0 的第 1、2 和 3 个元素; 对 ct_0 也是如此.

● $Decrypt_{\text{out}}(sk,pk,ct)$ 为了便于用户使用轻量级的移动设备随时随地访问数据, 本文基于文献 [20] 中的安全外包算法, 将解密算法中的 6 个映射运算外包至云服务器. 以 num 中的映射运算 $e\left(\prod_{i\in I}ct_{i,1}^{\gamma_i},sk_{0,j}\right)$, j=1, 2, 3 为例,

令 $\prod_{i\in I} ct_{i,1}^{\gamma_i} = A$, $sk_{0,j} = B$, 用户需要计算 $e(A, B)$, 利用文献 [20] 中的算法 A, 系统生成公共参数 (G_1, G_2, G_T, e, q, P_1, P_2), 用户调用 RANDA 函数生成静态表 ST 和动态表 DT. ST 包含参数 $e(P_1, P_2)$ 和 $\{\alpha_j, \beta_{j,1}, \beta_{j,2}\}_{j=1,2,\ldots,n}$; DT 包含参数:

$$(x_1P_1, x_3P_1, x_1x_2^{-1}x_5P_1, x_7P_1, x_1^{-1}x_2P_2, x_4P_2, x_1^{-1}x_6P_2, x_8P_2, e(P_1,P_2)^{x_7x_8}, e(P_1,P_2)^{x_5+x_6-x_2}).$$

用户向外包服务器 S1 请求如下值:

1) $U_1(A + x_1P_1, B + x_1^{-1}x_2P_2) \to \alpha_1$

2) $U_1(x_3P_1, x_4P_2) \to \alpha_2$

类似地, 用户向外包服务器 S2 请求如下值:

1) $U_2(A + x_1x_2^{-1}x_5P_1, -x_1^{-1}x_2P_2) \to \alpha'_1$

2) $U_2(-x_1P_1, B + x_1^{-1}x_6P_2) \to \alpha'_2$

3) $U_2(x_3P_1, x_4P_2) \to \alpha'_3$

4) $U_2(x_7P_1, x_8P_2) \to \alpha'_4$

用户收到上述值后, 分别校验 $\alpha_2 = \alpha'_3$ 以及 $e(P_1, P_2)^{x_7x_8} = \alpha'_4$, 若方程成立, 则计算:

$$o = \alpha\alpha'_1\alpha'_2 e(P_1, P_2)^{x_5+x_6-x_2}.$$

输出 o 即为 e(A, B). 同样的方式可以计算 *den* 中的 3 个映射运算.

4.3.2 正确性证明

1) *Verify*(Z_u, *vkey*, *pk*) 函数的正确性. 下列公式可以直观的表明 *Verify*(Z_u, *vkey*, *pk*) 函数的正确性.

$$e\left(g^{H(ctr)} \cdot Z_u, K_p\right) = e\left(g^{H(ctr)+z_u}, h^{1/(H(ctr)+z_u)}\right) = E,$$

$$K_c = e\left(g, K_p\right) = e\left(g, h^{1/(H(ctr)+z_u)}\right) = E^{1/(H(ctr)+z_u)}.$$

2) 解密正确性. 我们证明当属性集合 S 满足 (M, π) 时, 解密算法可以概率 1 恢复正确的明文消息. 对于 l=1, 2, 3, 有:

$$\begin{aligned}\prod_{i\in I} ct_{i,l}^{\gamma_i} &= \prod_{i\in I} (\mathcal{H}(\pi(i)l1)^{s_1} \cdot \mathcal{H}(\pi(i)l2)^{s_2}) \cdot \prod_{j=1}^{n_2} [\mathcal{H}(0jl1)^{s_1} \cdot \mathcal{H}(0jl2)^{s_2}]^{\gamma_i(M)_{i,j}} \\ &= \left(\prod_{j=1}^{n_2} [\mathrm{H}(0jl1)^{s_1} \cdot \mathcal{H}(0jl2)^{s_2}]^{\gamma_i(M)_{i,j}}\right) \cdot \prod_{i\in I} (\mathcal{H}(\pi(i)l1)^{s_1} \cdot \mathcal{H}(\pi(i)l2)^{s_2}) \\ &= \mathcal{H}(01l1)^{s_1} \cdot \mathcal{H}(01l2)^{s_2} \cdot \prod_{i\in I} (\mathcal{H}(\pi(i)l1)^{s_1} \cdot \mathcal{H}(\pi(i)l2)^{s_2}).\end{aligned}$$

以上公式中的第 3 个等式根据公式 (1) 得到. 则 *num* 中除第 1 项外的乘积可由如下公式给出:

$$\prod_{t\in\{1,2\}} \Big[e(\mathcal{H}(011t), h)^{b_1r_1s_t} \cdot e(\mathcal{H}(012t), h)^{b_2r_2s_t} \cdot e(\mathcal{H}(013t), h)^{(r_1+r_2)s_t} \cdot \prod_{i\in I} e(\mathcal{H}(\pi(i)1t)^{\gamma_i}, h)^{b_1r_1s_t} \cdot e(\mathcal{H}(\pi(i)2t)^{\gamma_i}, h)^{b_2r_2s_t} \cdot e(\mathcal{H}(\pi(i)3t)^{\gamma_i}, h)^{(r_1+r_2)s_t}) \Big].$$

当上式除以 *den* 时, 容易看出我们只剩下下列式子的倒数 (其余项抵消).

$$\left(\prod_{t\in\{1,2\}} e(g^{d_t} \cdot g^{\frac{\sigma'}{a_t}} \prod_{i\in I} g^{\frac{\gamma_i\sigma_{\pi(i)}}{a_t}}, h^{a_ts_t})\right) \cdot e(g^{d_3} \cdot g^{-\sigma'} \prod_{i\in I} g^{-\gamma_i\sigma_{\pi(i)}}, h^{(s_1+s_2)}).$$

上式正好等于 $e(g,h)^{d_1a_1s_1+d_2a_2s_2+d_3(s_1+s_2)}$, 因此, 可以成功恢复消息明文.

5 安全性分析

5.1 数据的机密性

数据的机密性通过 IPRE 算法实现, 本节我们证明 IPRE 算法为选择明文 (CPA) 安全的.

定理 1. 数据的机密性. 若在 DBDH 假设下, 端加密和重加密均为 CPA 安全的, 则本文方案满足数据的机密性.

由于 IPRE 算法包含端加密和重加密两部分, 我们通过引理 2 和引理 3 的证明完成定理 1 的证明.

引理 2. 在 DBDH 假设下, 端加密为 CPA 安全的.

证明: 如果一个 CPA 类型的敌手 $\mathcal{A}$ 能以不可忽略的概率 ϵ 攻破端加密算法, 则可以构建算法 $\mathcal{B}$, 通过与 $\mathcal{A}$ 的交互解决 DBDH 问题. 算法 $\mathcal{B}$ 的输入为随机的挑战值 (g, g^a, g^b, g^c, T), $\mathcal{B}$ 的目标是判决 $T=e(g, g)^{abc}$ 还是随机值.

(1) 系统建立. 均匀随机选取 $\delta \leftarrow \mathbb{Z}_p^*$, 设 $g_1 = g, g_2 = g^\delta, h_1 = g^a, h_2 = g^b, L = e(g^a, g^b)$. 输出 $param=(g_1, g_2, h_1, h_2, L)$ 作为系统参数.

(2) 询问阶段 1. $\mathcal{B}$ 按下列方式回答敌手 $\mathcal{A}$ 的询问.

- $Q_{hkg}(i)$. 均匀随机选择 $x_i, y_i \leftarrow \mathbb{Z}_p^*$.

若为第 k 次密钥生成询问 (B 猜测该用户为目标用户), 令 $i^*=i$, 计算 $pk_{i^*} = (X_i = g^{x_{i^*}}, Y_i = g^{y_{i^*}})$, 设 $sk_{i^*} = \left(\frac{x_{i^*}}{a}, y_{i^*}\right)$. 返回 pk_{i^*}.

否则, 计算 $pk_i = (X_i = h_1^{x_i}, Y_i = h_2^{-1} g^{y_i})$, 设 $sk_i = (x_i, y_i - b)$. 返回 pk_i.

- $Q_{ckg}(i)$: 均匀随机选择 $x_i, y_i \leftarrow \mathbb{Z}_p^*$, 设 $sk_i = (x_i, y_i)$ 计算 $pk_i = (X_i = h_1^{x_i}, Y_i = g^{y_i})$, 返回 (sk_i, pk_i).
- $Q_{hkg}(pk_i, pk_j)$. 给定 (pk_i, pk_j), pk_i 和 pk_j 分别由询问 Q_{hkg} 和 Q_{ckg} 获得. B 区分下列情况.

若 $pk_i \neq pk_{i^*}$, 运行函数 *ReKeyGen(sk*$_i$*, pk*$_j$*)*, 返回函数输出.

若 $pk_i = pk_{i^*}$, 且 pk_j 未被攻陷, 则计算 $W = \left(h_2 Y_j g_2\right)^{\left(\frac{x_{i^*}}{a}\right)^{-1}} = (g^a)^{(y_j+\delta)/x_{i^*}}$, 返回 $rk_{i\to j} = W$.

若 $pk_i = pk_{i^*}$, 且 pk_j 已被攻陷, 则返回 $\perp$.

(3) 挑战阶段. 当 $\mathcal{A}$ 决定完成阶段 1 时, 输出 m_0, m_1 以及目标公钥 pk^*. 若 $pk_{i^*} \neq pk^*$, $\mathcal{B}$ 输出 1 个随机比特并拒绝. 否则, $\mathcal{B}$ 挑选一个随机比特 $d \in \{0, 1\}$ 并计算 $c_0 = T \cdot m_d, c_1 = g_1^r = g^c, c_2 = e(h_1, g_2)^r = e(g^a, g^c)^\delta, c_3 = X_i^r = (g^c)^{x_{i^*}}$, 隐式设置 $r=c$. 假设 $\mathcal{A}$ 执行 Q_{hkg} 最多 q_{hkg} 次. 则 $\mathcal{B}$ 在挑战阶段猜测 i^* 正确的概率至少 $1/q_{hkg}$.

(4) 询问阶段 2. $\mathcal{A}$ 执行和阶段 1 相同的询问.

(5) 输出阶段. $\mathcal{A}$ 输出猜测值 $d' \in \{0, 1\}$. 若 $d=d'$, $\mathcal{B}$ 输出 1, 否则输出 0.

$\mathcal{B}$ 解决 DBDH 问题的概率至少为 $1/q_{hkg} \cdot \epsilon$, 证毕.

引理 3. 在 DBDH 假设下, 重加密为 CPA 安全的.

证明: 如果敌手 $\mathcal{A}$ 能以不可忽略的概率 ϵ 攻破重加密算法, 则可以构建算法 $\mathcal{B}$, 通过与敌手 $\mathcal{A}$ 的交互解决 DBDH 问题. 算法 $\mathcal{B}$ 的输入为随机的挑战值 (g, g^a, g^b, g^c, T), B 的目标是判决 $T=e(g, g)^{abc}$ 还是随机值. 算法 $\mathcal{B}$ 工作流程如下.

(1) 系统建立 (λ). 均匀随机选取 $\delta \leftarrow \mathbb{Z}_p^*$, 设 $g_1 = g, g_2 = g^\delta, h_1 = g^a, h_2 = g^b, L = e(g^a, g^b)$. 输出 $param=(g_1, g_2, h_1, h_2, L)$ 作为系统参数.

(2) 问询阶段 1. $\mathcal{B}$ 按下列方式回答敌手 $\mathcal{A}$ 的询问.

- $Q_{hkg}(i)$. 均匀随机选择 $x_i, y_i \leftarrow \mathbb{Z}_p^*$

若为第 k 次密钥生成询问 ($\mathcal{B}$ 猜测该用户为目标用户), 令 $i^*=i$, 计算 $pk_{i^*} = (X_i = {h_1}^{x_{i^*}}, Y_i = h_2^{-1} g^{y_{i^*}})$, 隐式设置 $sk_{i^*} = (x_{i^*}, y_{i^*} - b)$. 返回 pk_{i^*}.

否则, 计算 $pk_i = (X_i = h_1^{x_i}, Y_i = g^{y_i})$, 设 $sk_i = (x_i, y_i)$. 返回 pk_i.

- $Q_{ckg}(i)$. 均匀随机选择 $x_i, y_i \leftarrow \mathbb{Z}_p^*$, 设 $sk_i = (x_i, y_i)$ 计算 $pk_i = (X_i = h_1^{x_i}, Y_i = g^{y_i})$, 返回 (sk_i, pk_i).
- $Q_{hkg}(pk_i, pk_j)$. 给定 (pk_i, pk_j), pk_i 和 pk_j 通过询问 Q_{hkg} 或 Q_{ckg} 获得. $\mathcal{B}$ 运行函数 *ReKeyGen*(sk_i, pk_j), 返回函数输出.

(3) 挑战阶段. 当敌手 $\mathcal{A}$ 决定完成阶段 1 时, 输出 m_0, m_1 以及目标公钥 pk^*. 若 $pk_{i^*} \neq pk^*$, $\mathcal{B}$ 输出 1 个随机比特并拒绝. 否则, $\mathcal{B}$ 挑选一个随机比特 $d \in \{0, 1\}$, 并计算 $c_0' = T \cdot m_d, c_1' = g_1^r = g^c, c_2' = L^r \cdot e({h_1}^r, Y_{i^*})^r = e(g^a, g^c)^{y_{i^*}}$. 假设 $\mathcal{A}$ 执行 Q_{hkg} 最多 q_{hkg} 次, 则 $\mathcal{B}$ 猜测 i^* 正确的概率至少 $1/q_{hkg}$.

(3) 询问阶段 2. $\mathcal{A}$ 执行和阶段 1 相同的询问.

(4) 输出阶段. $\mathcal{A}$ 输出猜测值 $d' \in \{0, 1\}$. 若 $d=d'$, $\mathcal{B}$ 输出 1, 否则输出 0.

$\mathcal{B}$ 解决 DBDH 问题的概率至少为 $1/q_{hkg} \cdot \epsilon$, 证毕.

5.2 细粒度的访问控制

本方案使用目前较为高效的 FAME 算法实现细粒度的访问控制, 因此其安全性可由 FAME 保证.

定理 2. 基于随机应答器 (random oracle, RO) 模型, 细粒度的访问控制在线性判决假设 (decisional linear assumption, DLIN)[38]下, 关于非对称映射群为适应性安全的.

证明见文献 [12] 的附录 C.

5.3 限定 σ -次访问

定理 3. 若 q-BDHI 判决假设成立, 其中 q 为解密算法执行的次数, 则 TO-FAME 机制中, 敌手在限定 σ -次数安全游戏中的优势是可忽略的.

证明: 假设存在敌手 $\mathcal{A}$ 可以攻陷限定 σ -次访问算法, 它与仿真者 $\mathcal{B}$ 交互过程如下.

(1) 系统建立. $\mathcal{B}$ 随机选取 $a_1,a_2\leftarrow_R\mathbb{Z}_p^*$, 以及 $d_1,d_2,d_3\leftarrow_R\mathbb{Z}_p$. 输出:

$$(h,H_1:=h^{a_1},H_2:=h^{a_2},T_1:=e(g,h)^{d_1a_1+d_3},T_2:=e(g,h)^{d_2a_2+d_3}).$$

作为公共密钥 pk. 选取 $b_1,b_2\leftarrow_R\mathbb{Z}_p^*$, 输出 $g,h,a_1,a_2,b_1,b_2,g^{d_1},g^{d_2},g^{d_3}$ 作为主密钥 msk. 将 pk 发送至 $\mathcal{A}$.

(2) 询问阶段. $\mathcal{A}$ 执行第 3.4 节中的适应性询问, 由于 $\mathcal{B}$ 知道主密钥 msk, 因此 $\mathcal{B}$ 可以回答 $\mathcal{A}$ 的询问.

(3) 挑战阶段. 假设在第 j 次解密请求时, $\mathcal{A}$ 声明了一个访问结构 $(M,\pi)^*$ 并将其发送至 $\mathcal{B}$. $\mathcal{B}$ 生成密文 ct^*, $\mathcal{A}$ 执行对 ct^* 的解密.

(4) 输出阶段. $\mathcal{A}$ 使用 *Decrypt* 算法对 ct^* 进行解密变换, 并输出解密明文消息 m.

若 $j>\sigma$ 时, 敌手 $\mathcal{A}$ 仍被允许执行解密算法, 则该敌手赢得比赛. 不失一般性, 我们假设 $j=\sigma+1$. 敌手 $\mathcal{A}$ 在 $j=\sigma+1$ 时赢得比赛, 表明解密的计数器 $ctr=\sigma+1$, 且集合 ST 存储 $\{K_{c,i}\}_{i\in[\sigma+1]}$, 其中, $\forall i\in[\sigma+1]$, $K_{c,i}$ 用于第 i 次解密. 注意到 $K_{c,i}=E^{\frac{1}{z_u+H(ctr)}}$ 实际上是 VRF[39]的输出, 当 q-BDHI 假设成立时, VRF 的唯一性成立. 唯一性保证 $\forall i,i'\in[\sigma+1]$, 若 $i\neq i'$, 则 $K_{c,i}\neq K_{c,i'}$. 由于 $\mathcal{A}$ 至多被允许解密 σ 次, 根据鸽笼原理, 至少 2 个 $K_{c,i}$ 的值相等. 然而算法中约束, 当存在重复的 $K_{c,i}$ 值时, 云服务器将拒绝解密请求, 明显存在矛盾现象. 因此, $\mathcal{A}$ 只能以可忽略的概率赢得比赛.

6 评估与实验

在本节中, 我们对本文提出的边缘重加密机制和外包的 FAME 机制进行了性能评估, 并分别给出了实验结果.

6.1 功能对比

我们将本文提出的 IPRE-TO-FAME 方案与现存的数据安全共享方案的功能进行了对比, 发现本文的方案可以实现动态多用户的细粒度访问控制、轻量级设备的安全加密、密文的归一化与高效的解密功能, 相比其他方案有较明显的优势, 具体情况如表 1 所示.

表 1 功能对比

机制	DC	MUAC	DUAC	IoTE	DeOS
文献[11]	√	—	—	√	—
文献[8]	√	√	√	—	√
文献[23]	√	√	√	—	√
本文	√	√	√	√	√

表 1 中, DC: 数据的机密性; MUAC: 多用户访问控制; TLAC: 次数受限的访问控制; IoTE: 物联网数据加密; DeOS: 解密外包.

6.2 性能评估

考虑到 IoTS, USD 资源受限特性及 MEC 的任务复杂性, 我们重点评估 IPRE 的端加密, 重加密、数据解密, 以及 TO-FAME 的授权解密算法的性能. 为了方便表示, 我们将文献 [11] 中的方案记为 A-PRE, 将文献 [40] 中的

方案记为 U-PRE.

6.2.1 IPRE 性能评估

IPRE 中所有运算均为群上的运算, 主要分为映射, 模幂和乘法 3 类. 其中, 乘法和模幂运算效率较高, 映射是属于复杂度较高的运算. 我们用这 3 类运算的数量对算法的计算效率进行初步评估, 具体情况如表 2 所示.

表 2 端加密计算效率

实验	算法	映射	模幂	乘法
端加密	本文	1 (可预计算)	4	1
	A-PRE	2 (可预计算)	6	2
	U-PRE	1	7	2
重加密	本文	1	0	1
	A-PRE	1	1	1
	U-PRE	2	2	2
解密	本文	1	1	2
	A-PRE	2	2	3
	U-PRE	2	2	3

可以看出, 本文的端加密算法, 重加密算法和解密算法具有明显的计算优势. 其中端加密虽然有 1 次映射运算, 但映射的输入为公共参数, 可以通过预计算的方式作为固定值输入, 实际加密时只需要 4 次模幂运算和 1 次乘法运算.

6.2.2 授权解密

授权解密算法将运算复杂度较高的映射运算全部外包至云服务器实现, 用户在接收到云服务器的结果后, 只需要在终端上执行 6 次 $\mathbb{Z}_p$ 上的乘法, 即可完成解密操作, 获取相应明文.

6.3 实验结果

本节描述实验的基本设置环境及实验结果. 实验环境为 2.30 GHz 的 Intel 处理器、8 GB 内存的 Linux 服务器. 实验使用 Visual Studio C++作为仿真平台, 添加了基于映射的密码学库 (版本为 PBC 0.5.14), 在 GNU 多精度算法帮助下完成. 在实验中, 使用 PBC 中的参数 *a.param* 来设置基础字段大小为 320 KB, $\mathbb{Z}_p$ 中一个元素的大小是 20 B.

我们分别进行了端加密、重加密和解密的实验, 将本文的方案与 A-PRE、U-PRE 方案做了对比, 实验结果如图 3–图 5 所示. 其中, 图 3–图 5 中纵轴均为算法运行耗时, 单位为 s. 图 3 和图 4 中, 横轴为文件大小, 范围为 10–50 MB, 图 5 的横轴为密文个数, 范围为 10–50 个.

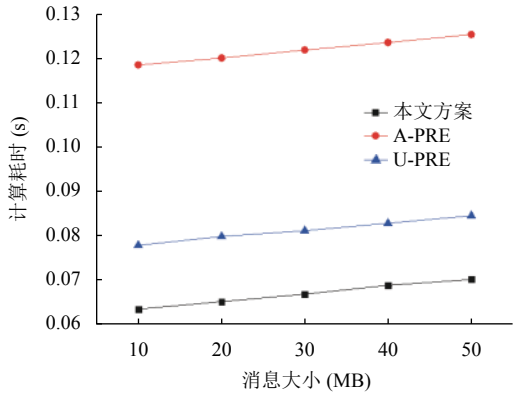


图 3 端加密算法计算效率对比

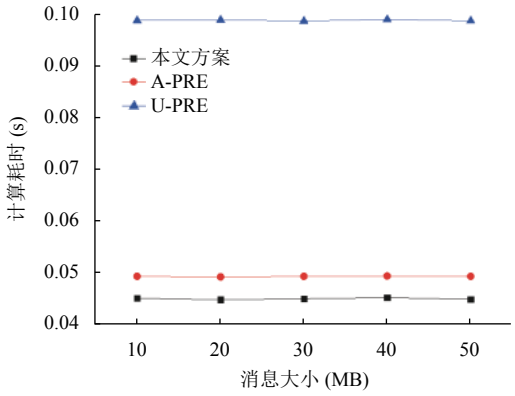


图 4 重加密算法计算效率对比

6.3.1 端加密的实验结果

端加密实验结果如图 3 所示, 可以看出随着明文数据量的增加, 3 类方案的端加密耗时均有轻微的增加, 但基

本不受影响. 我们回忆端加密算法, 以本文方案为例.

给定输入 X_i 及 m, 均匀并随机地选取 $r \leftarrow \mathbb{Z}_p$, 计算 $c_0 = L^r \cdot m, c_1 = g_1^r, c_2 = e(h_1, g_2)^r, c_3 = X_i^r$. 设置 $C_i = (c_0, c_1, c_2, c_3)$ 作为初始密文.

可以看出, 端加密算法仅在计算 c_0 时与明文 m 相关, 而且只需 1 次高效的乘法运算, 因此, 对算法的整体计算效率影响不大.

6.3.2 重加密的实验结果

重加密实验结果如图 4 所示. 可以看出, 随着明文数据量的增加, 这 3 类方案的重加密耗时完全不受影响. 这是因为重加密算法没有任何与明文有关的运算, 因此计算效率完全与明文无关.

6.3.3 数据解密的实验结果

数据解密实验结果如图 5 所示. 可以看出, 解密耗时随密文的个数线性增长. 这是因为, 每次解密都需要对每个密文的所有内容进行运算, 因此有明显的线性关系. 由于用户一般只对自己感兴趣的数据进行解密, 通常认为解密的规模是可控的, 因此解密计算效率的线性增长并不影响系统的整体性能.

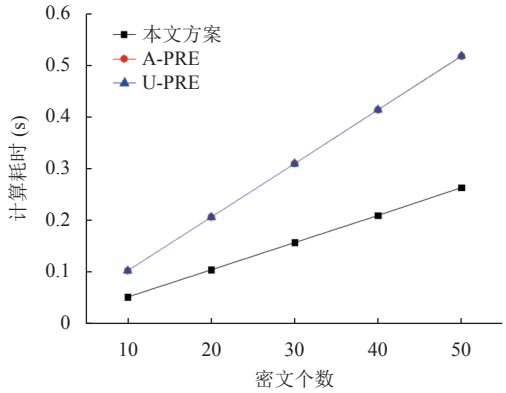


图 5 数据解密算法计算效率对比

7 总 结

本文提出了一种适用于 PHC 系统的安全数据共享方案, 解决了物联网感知数据的安全上传、加密归一化、动态多用户的细粒度的访问控制, 以及高效的解密等问题, 实现了物联网和云服务器相结合的网络架构中的安全数据共享. 下一步我们考虑将该方案应用于更多的场景, 尤其是根据目前广受关注的公共卫生事件 (如新冠肺炎防疫) 的需求, 探索将本文的方案用于样本数据的采集、上传和存储等, 以防止用户隐私 (如行为轨迹, 身份等) 的泄露.

References:

[1] Islam R, Kikuchi K, Sato Y, Izukura R, Yokota F, Nishikitani M, Tasneem R, Sultana N, Ahmed A, Nakashima N. Personal health record (PHR) system in portable health clinic. In: Pape-Haugaard LB, Lovis C, Madsen IC, Weber P, Nielsen PH, Scott P, eds. Digital Personalized Health and Medicine. Amsterdam: IOS Press, 2020. [doi: 10.3233/SHTI200435]

[2] Islam R, Nohara Y, Rahman J, Sultana N, Ahmed A, Nakashima N. Portable health clinic: An advanced tele-healthcare system for unreached communities. In: Ohno-Machado L, Séroussi B, eds. MEDINFO 2019: Health and Wellbeing e-Networks for All. Amsterdam: IOS Press, 2019. 616–619. [doi: 10.3233/SHTI190296]

[3] Ying ZB, Jiang WJ, Liu XM, Xu SM. Implementing security-enhanced PHR system in the cloud using FAME. In: Proc. of the 2019 IEEE Global Communications Conf. Waikoloa: IEEE, 2019. 1–6. [doi: 10.1109/GLOBECOM38437.2019.9014230]

[4] Kallahalla M, Riedel E, Swaminathan R, Wang Q, Fu K. Plutus: Scalable secure file sharing on untrusted storage. In: Proc. of the 2nd USENIX Conf. on File and Storage Technologies. San Francisco: USENIX Association, 2003. 29–42. [doi: 10.5555/1090694.1090698]

[5] De Capitani di Vimercati S, Foresti S, Jajodia S, Paraboschi S, Samarati P. Over-encryption: Management of access control evolution on outsourced data. In: Proc. of the 33rd Int'l Conf. on Very Large Data Bases. Vienna: ACM, 2007. 123–134. [doi: 10.5555/1325851.

1325869]

[6] Goh EJ, Shacham H, Modadugu N, Boneh D. SiRiUS: Securing remote untrusted storage. In: Proc. of the 2003 Network and Distributed System Security Symp. San Diego: The Internet Society, 2003.

[7] Ateniese G, Fu K, Green M, Hohenberger S. Improved proxy re-encryption schemes with applications to secure distributed storage. In: Proc. of the 2005 Network and Distributed System Security Symp. San Diego: The Internet Society, 2005.

[8] Yu SC, Wang C, Ren K, Lou WJ. Achieving secure, scalable, and fine-grained data access control in cloud computing. In: Proc. of the 2010 IEEE INFOCOM. San Diego: IEEE, 2010. 1–9. [doi: 10.1109/INFCOM.2010.5462174]

[9] Goyal V, Pandey O, Sahai A, Waters B. Attribute-based encryption for fine-grained access control of encrypted data. In: Proc. of the 13th ACM Conf. on Computer and Communications Security. Alexandria: ACM, 2006. 89–98. [doi: 10.1145/1180405.1180418]

[10] Xu L, Wu XX, Zhang XW. CL-PRE: A certificateless proxy re-encryption scheme for secure data sharing with public cloud. In: Proc. of the 7th ACM Symp. on Information, Computer and Communications Security. Seoul: ACM, 2012. 87–88. [doi: 10.1145/2414456.2414507]

[11] Guo H, Zhang ZF, Xu J, An NY, Lan X. Accountable proxy re-encryption for secure data sharing. IEEE Trans. on Dependable and Secure Computing, 2021, 18(1): 145–159. [doi: 10.1109/TDSC.2018.2877601]

[12] Agrawal S, Chase M. FAME: Fast attribute-based message encryption. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: ACM, 2017. 665–682. [doi: 10.1145/3133956.3134014]

[13] Sahai A, Waters B. Fuzzy identity-based encryption. In: Proc. of the 24th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Aarhus: Springer, 2005. 457–473. [doi: 10.1007/11426639_27]

[14] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption. In: Proc. of the 2007 IEEE Symp. on Security and Privacy. Berkeley: IEEE, 2007. 321–334. [doi: 10.1109/SP.2007.11]

[15] Pirretti M, Traynor P, McDaniel P, Waters B. Secure attribute-based systems. In: Proc. of the 13th ACM Conf. on Computer and Communications Security. Alexandria: ACM, 2006. 99–112. [doi: 10.1145/1180405.1180419]

[16] Nishide T, Yoneyama K, Ohta K. Attribute-based encryption with partially hidden encryptor-specified access structures. In: Proc. of the 6th Int'l Conf. on Applied Cryptography and Network Security. New York: Springer, 2008. 111–129. [doi: 10.1007/978-3-540-68914-0_7]

[17] Lewko A, Waters B. Decentralizing attribute-based encryption. In: Proc. of the 30th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Tallinn: Springer, 2011. 568–588. [doi: 10.1007/978-3-642-20465-4_31]

[18] Lv ZQ, Zhang M, Feng DG. Cryptographic access control scheme for cloud storage. Journal of Frontiers of Computer Science and Technology, 2011, 5(9): 835–844 (in Chinese with English abstract). [doi: 10.3778/j.issn.1673-9418.2011.09.007]

[19] Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts. In: Proc. of the 20th USENIX Conf. on Security. San Francisco: USENIX Association, 2011. 34. [doi: 10.5555/2028067.2028101]

[20] Tian HB, Zhang FG, Ren K. Secure bilinear pairing outsourcing made more efficient and flexible. In: Proc. of the 10th ACM Symp. on Information, Computer and Communications Security. Singapore: ACM, 2015. 417–426. [doi: 10.1145/2714576.2714615]

[21] Liu ZC, Jiang ZL, Wang X, Yiu SM. Practical attribute-based encryption: Outsourcing decryption, attribute revocation and policy updating. Journal of Network and Computer Applications, 2018, 108: 112–123. [doi: 10.1016/j.jnca.2018.01.016]

[22] Yuen TH, Liu JK, Au MH, Huang XY, Susilo W, Zhou JY. *k*-times attribute-based anonymous access control for cloud computing. IEEE Trans. on Computers, 2015, 64(9): 2595–2608. [doi: 10.1109/TC.2014.2366741]

[23] Ning JT, Cao ZF, Dong XL, Liang KT, Ma H, Wei LF. Auditable σ-time outsourced attribute-based encryption for access control in cloud computing. IEEE Trans. on Information Forensics and Security, 2018, 13(1): 94–105. [doi: 10.1109/TIFS.2017.2738601]

[24] Rouselakis Y, Waters B. Practical constructions and new proof methods for large universe attribute-based encryption. In: Proc. of the 2013 ACM SIGSAC Conf. on Computer & Communications Security. Berlin: ACM, 2013. 463–474. [doi: 10.1145/2508859.2516672]

[25] Blaze M, Bleumer G, Strauss M. Divertible protocols and atomic proxy cryptography. In: Proc. of Int'l Conf. on the Theory and Application of Cryptographic Techniques. Espoo: Springer, 1998. 127–144. [doi: 10.1007/BFb0054122]

[26] Green M, Ateniese G. Identity-based proxy re-encryption. In: Proc. of the 5th Int'l Conf. on Applied Cryptography and Network Security. Zhuhai: Springer, 2007. 288–306. [doi: 10.1007/978-3-540-72738-5_19]

[27] Ge CP, Xia JY, Wu A, Li HW, Wang Y. A source hiding identity-based proxy reencryption scheme for wireless sensor network. Security and Communication Networks, 2018, 2018: 6395362. [doi: 10.1155/2018/6395362]

[28] Wang XA, Yang XY, Li C, Liu YD, Ding Y. Improved functional proxy re-encryption schemes for secure cloud data sharing. Computer Science and Information Systems, 2018, 15(3): 585–614. [doi: 10.2298/CSIS171218024W]

[29] Liang XH, Cao ZF, Lin H, Shao J. Attribute based proxy re-encryption with delegating capabilities. In: Proc. of the 4th Int'l Symp. on

Information, Computer, and Communications Security. Sydney: ACM, 2009. 276–286. [doi: 10.1145/1533057.1533094]

[30] Sur C, Jung CD, Park Y, Rhee KH. Chosen-ciphertext secure certificateless proxy re-encryption. In: Proc. of the 11th Communications and Multimedia Security. Linz: Springer, 2010. 214–232. [doi: 10.1007/978-3-642-13241-4_20]

[31] Canetti R, Hohenberger S. Chosen-ciphertext secure proxy re-encryption. In: Proc. of the 14th ACM Conf. on Computer and Communications Security. Alexandria: ACM, 2007. 185–194. [doi: 10.1145/1315245.1315269]

[32] Libert B, Vergnaud D. Unidirectional chosen-ciphertext secure proxy re-encryption. In: Proc. of the 11th Int'l Workshop on Practice and Theory in Public-key Cryptography. Barcelona: Springer, 2008. 360–379. [doi: 10.1007/978-3-540-78440-1_21]

[33] Yang Y, Ma MD. Conjunctive keyword search with designated tester and timing enabled proxy re-encryption function for e-health clouds. IEEE Trans. on Information Forensics and Security, 2016, 11(4): 746–759. [doi: 10.1109/TIFS.2015.2509912]

[34] Hong HS, Sun ZX. Sharing your privileges securely: A key-insulated attribute based proxy re-encryption scheme for IoT. World Wide Web, 2018, 21(3): 595–607. [doi: 10.1007/s11280-017-0475-8]

[35] Kim SH, Lee IY. IoT device security based on proxy re-encryption. Journal of Ambient Intelligence and Humanized Computing, 2018, 9(4): 1267–1273. [doi: 10.1007/s12652-017-0602-5]

[36] Li WC, Jin CJ, Kumari S, Xiong H, Kumar S. Proxy re-encryption with equality test for secure data sharing in Internet of Things-based healthcare systems. Trans. on Emerging Telecommunications Technologies, e3986. [doi: 10.1002/ett.3986]

[37] Lewko A, Waters B. Unbounded HIBE and attribute-based encryption. In: Proc. of the 30th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Tallinn: Springer, 2011. 547–567. [doi: 10.1007/978-3-642-20465-4_30]

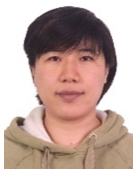
[38] Boneh D, Boyen X, Shacham H. Short group signatures. In: Proc. of the 24th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 2004. 41–55. [doi: 10.1007/978-3-540-28628-8_3]

[39] Dodis Y, Yampolskiy A. A verifiable random function with short proofs and keys. In: Proc. of the 8th Int'l Workshop on Theory and Practice in Public Key Cryptography. Les Diablerets: Springer, 2005. 416–431. [doi: 10.1007/978-3-540-30580-4_28]

[40] Hui G, Zhang ZF, Jiang Z. Proxy re-encryption with unforgeable re-encryption keys. In: Proc. of the 13th Int'l Conf. on Cryptology and Network Security. Heraklion: Springer, 2014. 20–33. [doi: 10.1007/978-3-319-12280-9_2]

附中文参考文献:

[18] 吕志泉, 张敏, 冯登国. 云存储密文访问控制方案. 计算机科学与探索, 2011, 5(9): 835–844. [doi: 10.3778/j.issn.1673-9418.2011.09.007]



朱雪岭(1982－), 女, 博士, 副研究员, 主要研究领域为密码学, 信息安全.

赵运磊(1974－), 男, 博士, 教授, 博士生导师, 主要研究领域为后量子密码, 密码协议, 计算理论.

侯慧莹(1992－), 女, 博士, CCF 学生会员, 主要研究领域为应用密码学, 信息安全, 车联网安全和属性基密码.

刘波(1973－), 男, 研究员, 博士生导师, 主要研究领域为机器学习, 自然语言处理, 网络空间安全.

付绍静(1984－), 男, 教授, CCF 高级会员, 主要研究领域为密码学, 网络空间安全.