雾计算中基于无配对 CP-ABE 可验证的访问控制方案

董江涛[1]，闫沛文[2]，杜瑞忠[2]

（1. 中国电子科技集团公司第五十四研究所，河北 石家庄 050081；
2. 河北大学网络空间安全与计算机学院，河北 保定 071002）

摘 要：雾计算将计算能力和数据分析应用扩展至网络边缘，解决了云计算的时延问题，也为数据的安全性带来新的挑战。基于密文策略的属性加密（CP-ABE）是保证数据机密性与细粒度访问控制的技术，其中双线性配对的计算开销过大制约了其应用与发展。针对此，提出了一种雾计算中基于无配对 CP-ABE 可验证的访问控制方案，为了使 CP-ABE 更加高效，使用椭圆曲线加密中的简单标量乘法代替双线性配对，从而减少总体计算开销；将解密操作外包给雾节点来降低用户计算复杂度，根据区块链防篡改可溯源的特性实现了对访问事务的正确性验证并记录访问授权过程。安全性与性能分析表明，所提方案在椭圆曲线的决策 DBDH 假设下是安全的，且计算效率更高。

关键词：访问控制；雾计算；基于密文策略属性加密；椭圆曲线加密

中图分类号：TP309

文献标识码：A

DOI: 10.11959/j.issn.1000−436x.2021162

Verifiable access control scheme based on unpaired CP-ABE in fog computing

DONG Jiangtao[1], YAN Peiwen[2], DU Ruizhong[2]

1. The 54th Research Institute of CETC, Shijiazhuang 050081, China
2. School of Cyber Security and Computer, Hebei University, Baoding 071002, China

Abstract: Fog computing extends computing power and data analysis applications to the edge of the network, solves the latency problem of cloud computing, and also brings new challenges to data security. Attribute encryption based on ciphertext strategy (CP-ABE) is a technology to ensure data confidentiality and fine-grained access control. The excessive computational overhead of bilinear pairing restricts its application and development. In response to this, a verifiable access control scheme was proposed based on unpaired CP-ABE in fog computing. In order to make CP-ABE more efficient, simple scalar multiplication in elliptic curve encryption was used to replace bilinear pairing, thereby reducing the overall computational overhead. Decryption operations were outsourced to fog nodes to reduce user computational complexity, and based on the tamper-proof and traceable characteristics of the blockchain, the correctness of the access transaction was verified and the access authorization process was recorded. Security and performance analysis shows that the scheme is safe under the elliptic curve decision-making DBDH (Diffie-Hellman) assumption, and the calculation efficiency is higher.

Keywords: access control, fog computing, CP-ABE, elliptic curve cryptography

收稿日期：2021−05−10；**修回日期**：2021−08−06

基金项目：国家自然科学基金资助项目（No.61572170）；河北省自然科学基金资助项目（No.F2018201153）；河北省自然科学基金重点资助项目（No.F2019201290）

Foundation Items: The National Natural Science Foundation of China (No.61572170), The Natural Science Foundation of Hebei Province (No.F2018201153), Key Project of Natural Science Foundation of Hebei Province (No.F2019201290)

1 引言

云计算能够将巨大的数据计算处理程序分解，然后通过服务器组成的系统进行处理并将结果返回给用户，在较短时间内完成对庞大数据的处理，但其提供的资源集中在用户的核心网络中，在用户与云交互信息时会产生时延较高的问题。随着用户对低时延的需求越来越高，雾计算作为云计算的延伸被提出[1]。与云计算相比，雾计算更侧重于以分布式方式部署应用程序与服务，将云服务与靠近网络边缘的分布式资源相结合，使存储和数据处理等贴近网络边缘设备，可以提供外包计算、资源分配和缓存等多种服务[2-4]，有效缓解云计算中存在的时延问题，并广泛应用在智能医疗、智慧城市等多个领域中[5-6]。

雾计算更贴近终端用户，数据中包含大量用户隐私信息，一旦发生数据泄露事件，将严重影响用户隐私。同时，为了保证数据只允许具有访问控制权限的用户查看，建立一种访问控制方案是必要的，如果采取传统的基于角色的访问控制方案[7]，用户需要通过角色去访问数据，授权形式单一。为每一个访问用户分配角色与之对应的权限，对于用户数量庞大的雾计算来说，分配角色与其对应权限的工作量是巨大的，并且仅通过角色去授予访问控制权限是粗粒度的，因此设计一种雾计算环境下高效、细粒度的访问控制方案是必要的[8]。基于密文策略的属性加密（CP-ABE, ciphertext policy attribute based encryption）技术可以实现对数据的细粒度访问控制，其密文与密钥都与用户的属性相关，数据拥有者可以根据用户的特征信息制定出由属性集合构成的加密策略，只有当用户的属性集合满足加密策略中的属性要求时才能对数据进行解密。灵活的访问权限授予能够适用于雾计算环境，并保护数据的安全，但效率不高的问题仍然制约了其发展与应用。在基于配对的密码协议中，与标量乘法相比，双线性配对被认为是开销最大、耗时最长的运算，实验表明，在同一椭圆曲线下，双线性配对操作的计算开销比标量乘法的计算开销大 2~3 倍。因此减少双线性配对的计算次数能够有效地提升 CP-ABE 的效率。为了减小用户的解密计算开销，研究者提出了外包解密的概念。解密算法中先使用转换密钥将密文转换为部分密文，且无法通过外包解密算法获取明文数据，再交由用户进行解密，这样用户仅需要较小的计算开销便可以获取明文数据。然而将解密外包后不能确保转换后密文的有效性，当外包解密机构不可信时，可能会篡改转换后密文，因此，对于外包解密的访问控制方案，验证转换后的密文是有必要的。而现有方案往往需要建立可信第三方，这需要高额的信任建立成本，且存在单点故障问题。为了解决上述问题，本文提出了一种雾计算中基于无配对 CP-ABE 可验证的访问控制方案。

结合雾计算的特点，本文建立了云−雾−用户分层的访问控制架构，用户通过雾节点与云进行交互。基于密文策略的属性加密技术来实现数据的机密性和细粒度访问控制，用线性秘密共享方案（LSSS, linear secret sharing scheme）构建访问结构来增强访问策略的表达能力，使只有满足访问结构的用户才能对数据进行访问。利用椭圆曲线加密中的简单标量乘法代替双线性配对计算，提升方案整体效率，并设计安全高效的外包解密方案，将解密中的部分计算外包给雾节点，减少数据用户的计算负担。因为区块链技术是理想的防止数据篡改的方法，在雾节点部署区块链，一方面利用区块链技术为雾计算提供可信和安全的环境，另一方面通过雾节点为区块链提供计算资源和存储能力。通过区块链记录整个访问授权的流程，以达到可溯源的目的，并对整个访问事务进行正确性验证以防被恶意篡改。对方案进行仿真实验对比，实验结果表明用户解密时间消耗较小且稳定，整体计算效率更高。

本文主要的研究工作如下。

1) 建立云−雾−用户分层的访问控制架构，并将区块链节点部署在雾层的雾节点设备中，一方面利用区块链技术为雾计算提供可信和安全的环境，另一方面通过雾节点为区块链提供计算资源和存储能力。利用区块链来记录整个访问控制的流程，使其不会被恶意篡改，保护数据安全。

2) 利用椭圆曲线加密中的简单标量乘法代替双线性配对计算，提高了 CP-ABE 算法效率，并结合外包解密思想，将解密操作外包给雾节点运行，充分利用了雾节点的计算能力，使用户解密操作的计算开销较小且恒定。本文方案的效率更高，且更适用于设备计算能力受限的雾计算环境。

3) 对本文方案进行安全性分析，建立敌手游戏模型，结果表明该方案满足明文攻击安全。通过性能分析与仿真实验分析，验证了本文方案在满足安全性前提下效率更高。

2　相关工作

2007 年，Bethencourt 等[9]首次提出 CP-ABE 方案，基于双线性对使数据加密与访问控制结合，并支持正负属性与门限访问结构，之后被广泛应用于云存储环境中数据的细粒度访问控制[10-11]。以上方案都采用单一授权中心为系统生成密钥并且管理属性，存在单点故障问题。因此 Lewko 等[12]提出了多权威机构的 CP-ABE 方案，且不需要属性权威机构进行协作。Horvath[13]为了增加系统灵活性，在多权威机构的 CP-ABE 方案的基础上，实现了基于身份的撤销。Hur[14]改进了撤销方案，支持对每个用户的属性集进行直接撤销，并在文献[15]的方案中进行了使用。Wang 等[16]解决了 CP-ABE 方案中的密钥托管问题，增强了属性表达能力。Li 等[17]提出的方案对外包密文正确性进行了验证，将验证过程在解密前执行，并且为授权用户与未授权用户检查外包密文的正确性，但方案中需要指定 2 种访问控制策略。Zhang 等[18]提出一种可追溯的多权威机构 CP-ABE 方案，该方案可以根据跟踪算法识别出泄露解密密钥的恶意用户，并降低跟踪的存储成本。现有 CP-ABE 方案中包含了线性对配对计算与求幂计算，极大地限制了在计算能力受限的边缘设备中的使用。

基于配对的密码协议的效率取决于配对计算的速度，因此，为了提高效率，学者做了大量的研究工作。对于如何优化 ECC 协议，Freeman 等[19]对椭圆曲线进行了分类，并介绍了其构造和一些优化技术。Scott[20]分析了如何改进配对操作来提升属性加密的效率。Simon 等[21]对椭圆曲线密码中的标量乘法进行了研究，为了进一步减少用户计算将复杂操作进行外包计算。Chevallier-mames 等[22]提出了一种外包计算的访问控制方案，将双线性配对计算操作进行外包，但该方案的服务器并不是完全可信的。Chen 等[23]的方案在此基础上进行改进，使系统的计算开销进一步降低，但方案中所提安全模型并不具备现实基础。为了从本质上优化 CP-ABE 算法，Odelu 等[24]提出了一种具有恒定密钥大小的 CP-ABE 方案，使用椭圆曲线密码学技术避免了双线性配对的计算开销，但该方案用门限方法构建访问结构，制约了其可拓展性。

区块链是一种具有去中心化，并保护数据完整性、有效性和真实性的技术，众多学者将 CP-ABE 与区块链技术相结合来开展一系列工作。Maesa 等[25]的方案将区块链技术与基于属性的访问控制模型相结合，使用区块链代替传统的数据库来存储策略，并以事务的形式对访问策略进行管理，能够防止任何一方否认策略的真实性。Dagher 等[26]基于区块链的框架提出了针对电子病历的访问控制方案，使病历在被有效访问的同时保护了患者的敏感信息，并使用加密的方法实现区块链中数据的隐私性。但该方案中采用了 PoW 共识机制，在保持区块链一致性时计算开销过于庞大。Dorri 等[27]提出了一种用于智能家居设备间访问控制的私有区块链方案。具体来说，访问策略存储在块头部，而访问操作、设备添加和删除记录存储在块体中。谢绒娜等[28]提出了一种基于区块链的可溯源访问控制机制，将客体存储在链下数据服务器，通过链上与链下相结合的方式，实现了对客体访问授权与访问过程的记录。应作斌等[29]针对电子健康档案中密钥管理及用户身份追溯问题，结合变色龙哈希与零知识证明技术完成区块链节点的注册与身份认证，并结合属性加密技术实现了完全细粒度的访问控制。

综上所述，传统的访问控制方案效率较低，双线性配对计算开销大。为此本文提出了雾计算中基于无配对 CP-ABE 可验证的访问控制方案，在保障安全的前提下，提升了加解密效率。

3　预备知识

3.1　区块链

区块链是分布式共享账本，具有抗伪造、可溯源、去中心化等特征，能够在不信任的实体之间建立信任关系。联盟区块链由预选的节点组成，且每个块的生成由所有的预选节点共同决定。

1) 数据结构。每个数据块都由一个块头和一个块体组成，其中块头包含前一个区块的哈希值、一组元数据、Merkle 根。通过哈希值链接前一区块；一组元数据表示难度、时间戳与随机数，包含区块生成过程的信息；Merkle 根总结区块链中所有交易的数据结构。块体则包含交易的详细信息。为了保证联盟区块链具有防篡改、完整性，主要使用的密码学算法有哈希算法与数字签名算法。

2) 共识协议。共识协议在区块链的节点之间定义一个公共规则来生成新的区块。目前，较主流的算法有 PoS、PoWer、DPoS、PBFT 等，每种算法都有各自的优缺点并适用于不同场景。联盟区块链相较于公有

区块链弱化了中心化，根据节点的准入机制，赋予节点信任值，所以更倾向于使用 PBFT、DPoS 共识机制。

3) 智能合约。智能合约是以信息化方式传播执行合约的一种计算机协议，区块链的出现为智能合约提供了安全的运行环境，被集成到以太坊中，所有节点协商的智能合约通过交易广播到区块链中达成共识，当一个或多个预定义条件被触发时，智能合约可以自动验证与执行。

3.2 椭圆曲线密码

椭圆曲线密码属于公钥密码体制，基于椭圆曲线离散对数问题的困难性，保证了其安全性。ECC 与其他的公钥密码体制相比，系统参数与密钥长度较小，安全性较高。

椭圆曲线是一个系数和变元都在有限域 Z_p（p 为有限域的阶数）的二元三次方程 $y^2 = x^3 + ax + b$，记为 $E_p(a,b)$。$E_p(a,b)$ 是由方程的全体解（x,y）与无穷远点 O 构成的集合，$x, y \in Z_p$ 是未知数，a,b 是系数，并满足 $4a^3 + 27b^2 \neq 0$。

椭圆曲线计算规则如下。

1) 椭圆曲线上点 Q，存在 $Q+0=Q-0=Q$。

2) 椭圆曲线上点 $U(x_1, y_1)$ 在椭圆曲线上，存在对称点 $V(x_1, -y_1)$，使 $U+V=0$，则对称点 $V(x_1, -y_1)$ 为椭圆曲线上点 $U(x_1, y_1)$ 的负元。

3) 当椭圆曲线上点 $U(x_1, y_1)$ 与 $V(x_2, y_2)$ 存在 $U \neq -V$，则 $R = U + V = (x_3, y_3)$。

$$x_3 = \Delta^2 - x_1 - x_2, y_3 = \Delta(x_1, x_3) - y_1$$

$$\Delta = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} (\mathrm{mod}\ p), x_1 \neq x_2 \\ \dfrac{3x_1^2 + u}{2y_1} (\mathrm{mod}\ p), x_1 = x_2 \end{cases}$$

3.3 LSSS

定义在一个多方集合的线性秘密分享方案 $\prod$ 在域 Z_p 上是线性的。

1) 每一个参与共享的秘密数据可以在域 Z_p 上形成向量。

2) 存在一个大小为 $l \times n$ 的矩阵 $\boldsymbol{M}$ 被命名为 $\prod$ 的共享生成矩阵，即存在一个 l 行 n 列的矩阵 $\boldsymbol{M}$ 为秘密共享方案 $\prod$ 的共享生成矩阵。对于所有 i=1,…, l，存在函数 ρ 将矩阵 $\boldsymbol{M}$ 的第 i 行记作 $\rho(i)$。选取随机值构成向量 $\boldsymbol{v}$=（s,r_2,…,r_n），其中 $s \in Z_p$ 是将要共享的秘密值，则 $\boldsymbol{Mv}$ 是根据 $\prod$ 得到的关于 s 的 l 个共享份额的向量形式，$\boldsymbol{Mv}_i$ 属于实体参与方 $\rho(i)$。

4 系统模型

4.1 整体框架

本文提出了一种雾计算中基于无配对 CP-ABE 可验证的访问控制方案，系统模型如图 1 所示，该模型主要由属性授权机构（AA, attribute authority）、云服务提供商（CSP, cloud service provider）、雾节点（FN, fog node）、数据拥有者（DO, data owner）、数据用户（DU, data user）5 类实体组成。

AA 是完全受信任的一方，遵循协议规范来执行任务，对系统中的属性进行管理，且每个 AA 所管理的属性都不相同，主要负责用户注册，为用户绑定唯一的用户标识符 UID，同时负责密钥的生成与分发，并维护所管理属性的属性列表。

CSP 提供数据存储等方面的服务，在本文方案中假设 CSP 可信。

FN 提供计算、存储等服务，每个 FN 都与 CSP 连接，负责将用户提交的密文上传到 CSP，并负责密文的部分解密。在 FN 建立区块链，对访问事务的正确性验证并记录访问授权过程。

DO 是有文件要上传到云端存储的用户，为文件定义访问结构，对文件进行加密后上传到 FN，再由 FN 上传到 CSP。

DU 是访问存储在 CSP 中的文件的用户，并不完全受到信任。只有数据用户的属性集满足嵌入在给定密文中的访问结构时，才可以使用私钥从密文中解密文件。

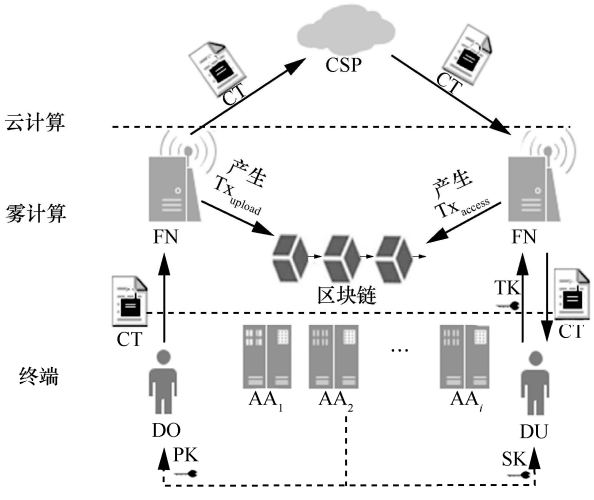

图 1 系统模型

4.2 算法定义

本文模型由系统初始化（Setup）、属性授权机

构初始化（AASetup）、私钥生成（KeyGen）、数据加密（Encypt）、数据解密（Decrypt）5 个算法构成。

1) 系统初始化 Setup(1^k)→(PP,UID)。初始化阶段输入安全参数 k 进行运算获得系统的全局参数 PP 与用户标识符 UID。

2) 属性授权机构初始化 AASetup(PP)→(PK, MSK)。多个属性授权机构独立运行对属性进行管理，且每个机构中的属性不重复。同时负责系统内密钥的生成与分发。

3) 私钥生成 KeyGen(PP,MSK,S,UID)→ ($SK_{i,UID}$, TK, USK)。由属性授权机构运行私钥生成算法通过输入全局参数 PP、主密钥 MSK，以及属性集合 S，计算得到用户的属性私钥 $SK_{i,UID}$，然后将其发送给用户。用户收到后根据属性私钥 $SK_{i,UID}$ 得到解密密钥 USK，并生成要交由雾节点外包计算的密钥 TK。在区块链中为用户与雾节点注册数字签名的公钥与私钥。

4) 数据加密 Encrypt(PP,PK,(A,ρ_i),M) → (CT)。DO 先使用对称加密算法对数据 M 进行加密，然后定义 LSSS 访问结构($\boldsymbol{A}$, ρ_i)，其中 $\boldsymbol{A}$ 表示根据访问策略生成的访问控制矩阵，ρ_i 表示矩阵 $\boldsymbol{A}$ 中属性所对应的行数，然后计算生成密文 CT 发送给雾节点。

5) 数据解密 Decrypt(PP, PK, TK, USK, CT)→(M)。该算法包括 2 个子算法：雾节点运行外包解密算法 FN.Decrypt 和用户执行本地子算法 DU.Decrypt。

FN.Decrypt(PP, PK, TK, CT)→(CT')。雾节点收到用户访问请求后，先从 CSP 获取密文 CT，然后使用转换密钥 TK 对其进行解密操作，最后将结果发送到区块链中进行正确性验证。

DU.Decrypt(PP, PK, USK, CT')→(M)。当区块链对雾节点生成的访问事务进行验证后，用户将获取转换密文 CT'，然后使用其自定义的解密密钥 USK 对密文进行解密，获取数据 M。

5　具体方案

本节描述了雾计算中基于区块链的无配对 CP-ABE 访问控制方案的具体结构，系统流程如图 2 所示。为了提高整个算法的性能，将复杂的双线性配对操作替换为椭圆曲线上的简单标量乘法，从而简化了计算。具体来说，数据拥有者首先用对称加密技术对密文进行加密，然后使用盲因子 sG 加密对称密钥 ck，其中 G 为阶是 r 的椭圆曲线循环群的生成器，s 为 Z_p 中的随机数，数据用户需要将属性密钥与密文进行计算才能够消除盲因子 sG。用户的每个属性都与身份标识 UID 绑定，如果用户在解密时互相勾结，则不能成功解密密文。在用户上传

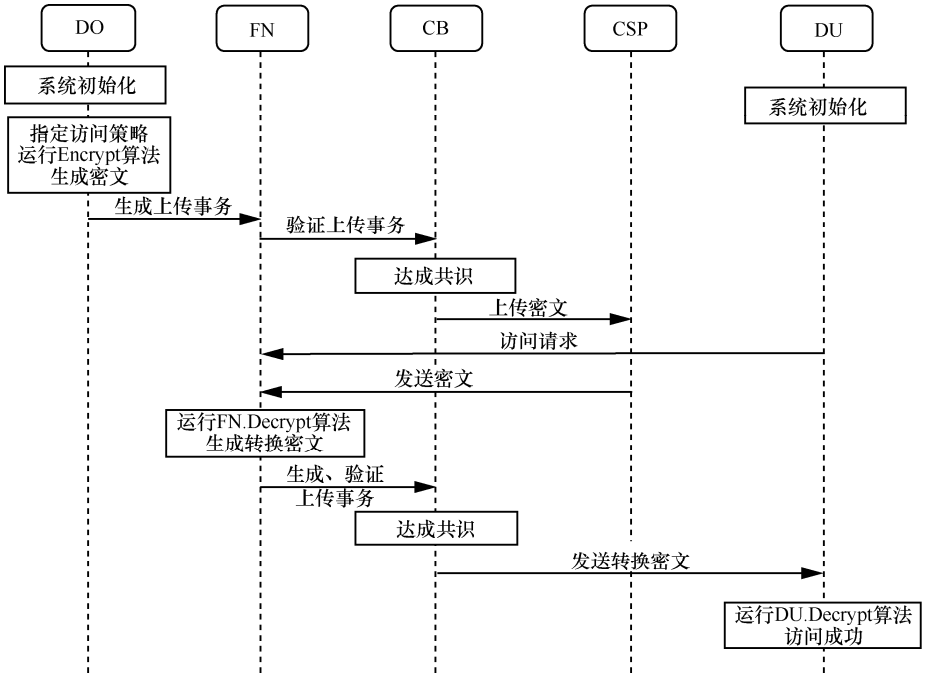

图 2　系统流程

文件与访问阶段，通过区块链对访问事务中数据进行正确性验证，并记录访问事务 Tx_{upload} 与 Tx_{access}，使数据上传与访问数据的整个过程可溯源。

阶段 1 系统初始化 Setup（1^k）→（PP，UID）

输入安全参数 k，得到全局参数 PP 与用户标识 UID。同时构建阶为 q 的有限域 GF(q)，在有限域 GF(q) 中选取椭圆曲线 E。选取哈希函数 H:{0,1}→Z_p，使用户身份标识 UID 映射到 Z 中元素，并定义属性集 $L=\{a_1,a_2,\cdots,a_m\}$，生成全局参数 PP = {GF(q), G, L, E, H}。

阶段 2 属性授权机构初始化 AASetup(PP, L)→(PK, MSK)

输入全局参数 PP，由系统中的多个属性授权机构共同运行该初始化算法，每个授权机构对属性进行管理，且每个授权中心所管理属性都不相同，对属性 a_i 选取随机数 $k_i \in Z_p$，并维护与用户标识 UID 对应的属性列表。生成系统公钥 PK 与主密钥 MSK。

$$\mathrm{PK} = \{k_i G \mid a_i \in L\}$$

$$\mathrm{MSK} = \{k_i \mid a_i \in L\}$$

阶段 3 私钥生成 KeyGen(PP, MSK, S, UID)→($SK_{i,UID}$, TK, USK)

该算法由属性授权机构运行，为用户生成私钥 $SK_{i,UID}$，表示身份标识为 UID 的用户所包含属性 a_i 的属性密钥。属性授权中心生成 $SK_{i,UID}$ 发送给对应用户，之后用户选取随机数 $z,n \in Z_p$，得到用户解密密钥 USK={d}，最后由用户生成雾节点转换密钥 TK。在用户向雾节点提交访问申请时，将 TK 交由雾节点进行外包解密计算，而用户解密密钥 USK 由用户自己保管，用来解密雾节点转换后的密文。

$$SK_{i,UID}=k_i+H(\mathrm{UID})n$$

$$TK=k_i+H(\mathrm{UID})n-d$$

阶段 4 数据加密 Encrypt (PP, PK, ($\boldsymbol{A}$, ρ_i), M) → (CT)

数据拥有者在上传文件前执行该算法对数据文件进行加密，输入全局参数 PP、系统公钥 PK，计算 $C_n=H(\mathrm{CT})$，当用户解密后用来验证解密的正确性；并根据访问控制策略定义 LSSS 访问结构($\boldsymbol{A}$, ρ_i)，来限制用户对文件的访问。其中，$\boldsymbol{A}$ 表示 $l\times m$ 的访问矩阵，$\rho(x)$表示访问矩阵 $\boldsymbol{A}$ 中每一行所对应的属性。具体步骤如下：DO 定义一个 LSSS 访问结构($\boldsymbol{A}$, ρ_i)，并随机选取密钥 ck，利用对称加密算法对文件 M 进行处理表示为 $E_{ck}(M)$，并计算 $C_n=H(E_{ck}(M))$。然后用户选取随机值 $s \in Z_p$，计算得到 $C_0 = ck + sG$。选取随机向量 $\boldsymbol{v}, \boldsymbol{u} \in Z_r^m$，其中，$\boldsymbol{v}$ 以选取的随机值 s 为第一个元素，$\boldsymbol{u}$ 以 0 为第一个元素，计算得到 $\lambda_x = \boldsymbol{A}_x\boldsymbol{v}$ 与 $\omega_x = \boldsymbol{A}_x\boldsymbol{u}$，其中 $x \in [0,l]$，$\boldsymbol{A}_x$ 表示访问矩阵的 x 行。对选取随机数 $r_x \in Z_r$，并计算得到{$C_{1,x}=\lambda_x G+r_x k_{\rho(x)}G$, $C_{2,x}=r_x G$, $C_{3,x}=\omega_x G$}，最后得到密文 CT={($\boldsymbol{A}$,ρ), C_0, C_n, $E_{ck}(M)$, {$C_{1,x}$, $C_{2,x}$, $C_{3,x} \mid \forall x \in [1,l]$}}，由用户将 CT 发送到雾节点由雾节点进行上传。

阶段 5 区块链上传验证阶段

因为用户并不直接与 CSP 交互，在上传文件时需要提交到雾节点进行处理，为了使数据文件的整个上传过程可溯源、抗伪造，用户生成上传事务 Tx_{upload}={U, checkCode, sign}，发送给 CB 进行验证，其中 U 表示上传事务的标识符，checkCode 表示密文的完整性校验码，sign 表示用户 DO 在 CB 上注册的私钥生成的数字签名。checkCode 是通过哈希散列值算法生成的，任意节点都可以通过 checkCode 验证密文的完整性。Sign 用来证明该上传请求确实是由 DO 发送的。

算法 1 生成上传事务

输入 上传事务的标识符 U，加密算法生成的密文 CT，联盟链 CB 中记录的用户 DO 的签名私钥 BSK_{DO}

输出 上传事务 Tx_{upload}

/*计算密文 CT 的信息摘要*/

1) checkCode=H(CT);

/*计算上传事务的信息摘要*/

2) MD=H(U,checkCode);

/*用 BSK_{DO} 对事务信息摘要进行签名*/

3) sign=$\mathrm{Sign}_{BSK_{DO}}$(MD);

/*生成一个上传事务*/

4) Tx_{upload}={U,checkCode,sign}

5) 返回 Tx_{upload};

在生成 Tx_{upload} 后，将其广播到 CB 中的其他节点进行验证，通过签名来验证事务的有效性，计算校验码验证密文有效性，确定其没有被恶意节点篡改。通过比较接收到的 Tx_{upload} 生成的信息摘要 MD' 与使用 DO 的公钥 BSK_{DO} 解密签名生成的信息摘要

MD，来验证 Tx_{upload} 的有效性。如果相等，则认为 Tx_{upload} 是有效的。同时为了保证上传密文的完整性，进一步检查信息摘要 checkCode'与 Txupload 中 checkCode 是否相等，一旦通过验证并由共识机制达成共识后，就会被打包成一个区块，再由雾节点将密文成功上传。

算法 2　验证上传事务

输入　Txupload={U, checkCode, sign}，联盟链 CB 中记录的用户 DO 的签名公钥 BPK_{DO}

输出　上传事务 Txstorage 与密文 CT 的验证结果

/*计算事务的信息摘要*/

1)　MD'=H(U, checkCode)；

/*使用用户公钥验证签名*/

2)　MD=Compute$_{BPK_{DO}}$(sign)；

3)　if MD'=MD then{

/*获取密文 CT，计算信息摘要*/

4)　　checkCode'=H(CT)；

5)　　if checkCode'=checkCode then

6)　　return true；}

7)　return false；

阶段 6 雾节点解密算法 FN.Decrypt

当数据用户向雾节点申请访问数据时，将转换密钥 TK 发送给雾节点，之后雾节点从云服务商获取密文 CT，运行 FN.Decrypt 算法，对密文 CT 进行转换。FN.Decrypt（PP，PK，TK，CT）→（CT'）。当申请访问的用户属性能够满足访问结构时，则存在一个常数集 $\{c_x \in Z_r\}_{x\in S}$，使 $\sum_{x\in S} c_x \boldsymbol{A}_x = (1,0,\cdots,0)$，$\sum_{x\in S} c_x \lambda_x = s$ 且 $\sum_{x\in S} c_x \omega_x = 0$。可以先通过以下计算得到参数 $\boldsymbol{A}_x$。

$$\boldsymbol{A}_x = C_{1,x} - \mathrm{TK}C_{2,x} + H(\mathrm{UID})(C_{2,x}nG + C_{3,x}) =$$
$$\lambda_x G + r_x k_{\rho(x)} G - (k_{\rho(x)} + H(\mathrm{UID})n - d)r_x G +$$
$$H(\mathrm{UID})(r_x nG + \omega_x G) =$$
$$\lambda_x G + r_x k_{\rho(x)} G - k_{\rho(x)} r_x G - H(\mathrm{UID})nr_x G +$$
$$dr_x G + H(\mathrm{UID})r_x nG + H(\mathrm{UID})\omega_x G =$$
$$\lambda_x G + dr_x G + H(\mathrm{UID})\omega_x G$$
$$D = \sum_{x\in S} c_x \boldsymbol{A}_x = \sum_{x\in S} c_x(\lambda_x G + H(\mathrm{UID})\omega_x G + dr_x G) =$$
$$sG + dG\sum_{x\in S} c_x r_x G$$
$$D' = \sum_{x\in S} c_x C_{2,x} = \sum_{x\in S} c_x r_x G$$

然后雾节点生成转换密文 CT'={C_0,E_{ck}(M), D,D'}，发送到 CB 生成访问事务并进行验证。

阶段 7　区块链访问验证阶段

在区块链收到转换密文后，生成访问事务并进行验证。雾节点从属性授权机构获取到用户当前属性，当用户的属性满足访问结构时，便会存在一个常数集 $\{c_x \in Z_r\}_{x\in S}$，使 $\sum_{x\in S} c_x \omega_x = 0$，所以用户权限验证算法 $E(x)= C_x C_{3,x}$ 中，当存在 $\sum_{x\in S} E(x) = 0$ 时，表示用户拥有该文件的访问权限。使用数据用户转换密钥 TK、转换密文 C''与密文 CT，以及用户的签名公钥计算得到信息摘要，在验证过程中信息摘要能够匹配时，证明雾节点并没有对访问事务中数据进行篡改。因为这种模式，转换密钥以及密文的任何变化都会改变区块链中相应的哈希值，并且该系统中的所有实体都会感知到这些变化，因此可以达到防篡改的目的。

算法 3　生成访问事务

输入　访问事务标识符 Ac，联盟链 CB 中记录的雾节点的签名私钥 BSK_{Fn}，转换密钥 TK 与密文 CT

输出　访问事务 Tx_{access}

/*设置常数集合 C_x*/

1)　$\{c_x \in Z_r\}_{x\in S}$；

/*执行用户权限验证算法*/

2)　$E(x)= c_x C_{3,x}$；

3)　if $\sum_{x\in S} E(x) = 0$ then{

/*计算访问事务的信息摘要*/

4)　　checkCode=H(BPK_{DU}, CT, CT', TK）

5)　　MD=H(Ac, checkCode, time)；

/*雾节点签署交易*/

6)　　sign=Sign$_{BSK_{Fn}}$(MD)；

7)　　Tx_{access}={Ac, checkCode, time, sign}；

8)　　return Tx_{access}；}

9)　return false；

验证数据用户访问权限后，为数据用户生成一个有效的访问事务 Tx_{access}={Ac, checkCode, time, sign}，其中 Ac 表示识别访问事务，checkCode 表示访问事务的信息摘要，time 表示 Tx_{access} 生成时的时间，sign 表示证明了该访问事务由哪个雾节点进行解密计算。在访问事务有效生成后被打包成块存储在区块链内，整个数据访问的参与者与

数据将被记录在内且不可伪造，以达到可溯源的目的，之后雾节点将生成的外包解密密文发送给用户。

算法 4 验证访问事务

输入 Tx_{access}={Ac, checkCode, time, sign}，联盟链 CB 中记录的用户 DU 的签名公钥 BPK_{DU}

输出 访问事务 Tx_{access} 验证结果

/*计算事务的信息摘要*/

1) MD'=H(Ac, checkCode, time)；

/*使用用户公钥验证签名*/

2) MD=$Compute_{BPK_{Fn}}$(sign)；

3) if MD'=MD then{

/*计算信息摘要*/

4) checkCode'=H(BPK_{DU},CT,CT',TK)；

5) if checkCode'=checkCode then

6) return true；}

7) return false；

阶段 8 用户解密阶段

DU.Decrypt(PP, PK, USK, CT')→(M)。用户收到雾节点发送的转换密文 CT' 后，运行算法 DU.Decrypt 得到对称密钥 ck。

$$C_0 - D + \mathrm{USK}D' = \mathrm{ck} + sG - \left(sG + d\sum_{x\in S} c_x r_x G\right) + d\sum_{x\in S} c_x r_x G = \mathrm{ck}$$

得到对称密钥 ck 后，通过对称解密算法将得到文件 M。验证 C_n=H($E_{\mathrm{ck}}(M)$)是否成立，成立则表示用户解密得到的数据没有被篡改。

6 安全性分析

本节主要分析所提方案在椭圆曲线的决策 DBDH 假设下的安全性。

定理 1 如果概率多项式时间算法 1 以不可忽略的优势 $\varepsilon > 0$ 破解本文提出的方案，那么算法 2 能够区分一个 DBDH 元组和一个随机元组，并且优势为 $\varepsilon/2$。

设 G 为以大素数 r 为阶、P 为生成元的群，首先由挑战者 C 首先随机选择 $a,b,c \in Z_p^*$，$\beta \in \{0, 1\}$ 与 $R \in P$，当 β=1 时 $Z = abcG$，否则 $Z = R$。挑战者 C 发送一个元组（G, aG, bG, cGZ）给 B，在下面的安全模型中由 B 将扮演挑战者。

A 和 B 执行如下操作。

初始化 敌手 A 先定义一个访问结构($\boldsymbol{A}^*$, ρ_i)，将其交给挑战者 B。

系统设置 为系统中的每个属性 a_i 创造公钥来对抗 A，挑战者 B 对系统进行初始化，选择随机数 $k_i \in Z_r$，生成 $\mathrm{PK} = \{k_i aG\}$。对每一个属性 a_i，B 选择随机数 $n, v \in Z_r$，将 nG、vG 当作系统的公钥发布，并将公钥发送给敌手 A，其中公共参数时随机分布的。

阶段 1 A 向 B 提交用户标识所对应的属性(a_i,UID)，用来请求用户属性所对应的私钥。对用户的身份标识 UID，用 T_{UID} 来标识敌手 A 要查询的属性 a_i 在访问矩阵 $\boldsymbol{A}^*$ 中的行子集，所请求的行子集 T_{UID} 中不能包含(1,…,0)，保证了攻击者不能够请求一组能够解密的密钥。B 随机选择 $t \in Z_r$，并计算 $tk_i a$ 作为私钥发送给敌手 A。

挑战阶段 A 向 B 发送等长的两段信息 m_0 和 m_1，B 选择一个值 $S' \in Z_r$，选取参数 $\beta \in \{0,1\}$，通过执行加密算法算法对 m_β 进行计算。B 构造密文 CT_b，将产生 C_0=m_β+sG，选取随机向量 $\boldsymbol{v}, \boldsymbol{u} \in Z_r^m$，其中 $\boldsymbol{v}$ 以选取的随机值 s 为第一个元素，$\boldsymbol{u}$ 以 0 为第一个元素，得到 $\lambda_x = \boldsymbol{A}_x \boldsymbol{v}$ 与 $\omega_x = \boldsymbol{A}_x \boldsymbol{v}$，其中 $x \in [0,l]$，$\boldsymbol{A}_x$ 表示访问矩阵的 x 行。最后由挑战者 B 生成 $C_{1,x} = \lambda_x G + r_x k_{\rho(x)} Z$，$C_{2,x} = r_x G$，$C_{3,x} = \omega_x cG$，将密文 $\mathrm{CT} = \{(\boldsymbol{A}, \rho), C_0, \{C_{1,x}, C_{2,x}, \ C_{3,x} \mid \forall x \in [1,l]\}\}$ 发送给敌手 A。

阶段 2 同阶段 1，敌手 A 可以继续向挑战者 B 发送(a_i, UID)密钥查询，并不违反规定。

猜测 游戏中如果 β'=β，挑战者 B 输出 0，表明 $Z = abcG$；否则，B 输出 1，表明 $Z = R$。

当 $Z = abcG$ 时，A 获得有效的密文，则 A 的优势为

$$|\Pr[\mathrm{B}(G, aG, bG, cG, Z = abcG) = 0] = \frac{1}{2} + \varepsilon$$

当 $Z = R$ 时，A 无法获得任何数据信息，则 A 的优势为

$$|\Pr[\mathrm{B}(G, aG, bG, cG, Z = R) = 0] = \frac{1}{2}$$

因此，B 破解该敌手游戏的优势为

$$\frac{1}{2}|\Pr[\mathrm{B}(G, aG, bG, cG, Z = abcG) = 0] + \Pr[\mathrm{B}(G, aG, bG, cG, Z = R) = 0]| - \frac{1}{2} = \frac{\varepsilon}{2}$$

在以上挑战过程中，假设 DBDH 成立，如果没有敌手在多项式时间内完成，那么本文方案是满足

椭圆曲线的决策 DBDH 假设下的安全性的。

本文方案具有区块链安全性、前向安全性和抗共谋安全性。

1) 在本文方案中，为了保证外包计算的正确性，需要对外包解密密文进行正确性验证，在大部分已有的可验证方案中都需要建立一个完全可信的中间实体，这会造成很高的信任构建成本，同时会存在单点故障问题。利用区块链抗篡改与可溯源特性来解决该问题，雾节点生成外包解密校验码发送到区块链，可以在不涉及任何中间实体的情况下对外包解密结果进行验证。同时对用户权限进行了验证，使属性不满足访问结构的用户不能从雾节点获取到转换密文，也就无法进行解密，转换密文与用户权限的验证都是通过区块链进行验证的，可以较小的人工干预实现可信计算，因此本文方案是可信的且具有验证性。为了保证数据用户能够获取完整数据，在数据上传验证过程中加入了密文的完整性校验码，能够随时对密文进行校验，确保了数据的完整性。并且在本文方案中可以跟踪与验证区块链中访问控制信息，任何数据的上传与用户的访问都将被记录为一个不可篡改的访问事务，可以确认哪个数据拥有者将何文件提交到 CSP，哪个数据用户与雾节点进行交互对数据进行了访问，任何数据的上传与用户的访问都将被记录为一个不可篡改的访问事务，因此整个访问方案满足可溯源性。

2) 在本文方案中，由多属性权威机构运行密钥生成算法生成属性密钥并发送给用户，但用户并无法从该属性密钥中获取与该属性相关的其他任何信息。同时，在属性授权机构中建立用户属性列表，当用户的属性被整体撤销时，需要在属性列表中删除与用户标识 UID 相关的所有属性信息，该用户在申请访问时，雾节点并不能从属性授权机构中获取到用户的相关属性信息，所以会拒绝该用户的访问。当用户所拥有的某个属性被撤销时，只需要在属性列表中修改撤销属性信息即可，这样雾节点获取的用户属性集合将不包含已撤销属性，当文件的访问结构中包含已撤销属性的数据时，将无法通过用户权限验证阶段，因此本文方案满足前向安全性。

3) 本文方案使用对称加密密钥 ck 对文件进行加密，然后设置访问结构 $\boldsymbol{A}^*$对 ck 加密，只有 DU 的属性满足访问结构时才能解密 ck+sG。假设多个未满足访问结构的用户相互勾结交换密钥，组成了满足访问结构中的属性集。根据密钥生成 KeyGen 算法，用户私钥中属性将与用户标识 UID 进行绑定，则产生不同的私钥 $\mathrm{SK}_{i,\mathrm{UID}}=k_i+H(\mathrm{UID})n$，假设互相勾结的用户身份分别为 r_1, r_2，因为不同用户之间的用户标识 UID 不同，通过解密计算得到 $\lambda_x G + H(\mathrm{UID}_{r_1})\omega_x G + dr_x G$ 与 $\lambda_x G + H(\mathrm{UID}_{r_2})\omega_x G + dr_x G$。其中 $H(\mathrm{UID}_{r_1})\omega_x G \neq H(\mathrm{UID}_{r_2})\omega_x G$ 便无法推出 sG，所以仍然无法对密文进行解密。因此，本文方案具有抗共谋攻击性。

7　性能分析

本节对本文方案在理论上和实验上进行分析，证明其可行性。本文方案与文献[17]、文献[18]、文献[24]方案的功能比较如表 1 所示。

表 1　　**功能比较**

方案	访问结构	多授权机构	双线性配对	外包计算
文献[17]方案	LSSS	No	Yes	Yes
文献[18]方案	LSSS	Yes	Yes	No
文献[24]方案	与门	No	No	No
本文方案	LSSS	Yes	No	Yes

7.1　理论分析

通过功能比较可以发现，在访问结构的比较中，文献[24]方案使用了与门的访问结构，其结构单一不够灵活，并且文献[17, 24]方案中采用单一授权机构进行授权，存在单点故障问题。而本文方案与文献[24]方案均避免了双线性配对带来的计算开销，能够提升整体效率。采用外包计算的本文方案与文献[17]方案能够降低用户在解密时的计算开销。

本文方案与文献[17]、文献[18]、文献[24]方案中用户加密、用户解密、外包解密中的计算开销比较如表 2 所示，其中，L 表示访问矩阵中包含属性个数，L_s 表示满足访问矩阵的最少属性数量，N 表示系统内属性数量，ω 表示与门结构中属性数，P 表示双线性配对操作，E 表示群内的乘法，E_g、E_T 分别表示群 G 内的指数操作与群 G_T 内的指数操作。

从表 2 可以看出，文献[18, 24]方案并没有使用外包计算技术，用户的计算开销随着访问控制策略的复杂化而呈线性增加趋势。但在使用了外包计算的本文方案与文献[17]方案中，用户解密时的计算

开销是稳定的。文献[17]方案中授权用户与未授权用户检查外包密文的正确性，虽然实现外包计算的正确性验证，但需要指定 2 种访问控制策略，用户加密时的开销要大于本文方案。文献[18]方案为了使整个 CP-ABE 方案做到可溯源，设计了可溯源算法，能够识别系统中泄露信息的恶意用户，但会导致用户的计算开销过大，用户加密与解密开销要远大于其他方案。本文方案与文献[24]方案中均使用椭圆曲线加密中标量乘法取代了双线性配对操作，在加解密时的计算开销均小于其他方案。综上所述，本文方案的计算开销要小于其他方案。

表 2　　计算开销比较

方案	用户加密	用户解密	外包解密
文献[17]方案	$(4L+2)E_g+4E_T$	E_T	$(L+2)P+2LE_g$
文献[18]方案	$6LE+(2L+1)E_T+(2L+1)P$	$3L_sP+3L_sE_g$	—
文献[24]方案	$(N-\omega+2)E$	$(N-\omega+3)E$	—
本文方案	$(4L+1)E$	$2E$	$(7L+1)E$

注：—表示方案中不涉及该项功能。

7.2　实验分析

本文方案的效率主要受区块链与 CP-ABE 加解密算法的影响，而区块链上的性能主要取决于共识算法如 PBFT，因此本文更侧重于运用椭圆曲线加密中的简单标量乘法代替双线性配对计算后 CP-ABE 的效率。通过仿真实验，将本文方案与文献[17-18, 24]方案在用户加密、用户解密与外包解密等方面进行比较分析。本文基于 charm-crypto 框架使用 Python2.7 进行仿真实验，并选择基于 512 B 有限域上阶为 160 的椭圆曲线；使用 Intel Xeon E5-2690 处理器，16 GB 内存，搭建环境为 Ubuntu16.04 系统的服务器作为雾节点和 AA；使用 Intel i5-4200U 1.6 GHz 处理器，4 GB 内存，搭建环境为 Ubuntu16.04 64 位系统的笔记本电脑充当用户。实验中所有结果均为 10 次结果的平均值。

数据拥有者加密计算时间比较如图 3 所示。从图 3 中可以看出，本文方案与文献[17-18, 24]方案的数据拥有者在加密计算时的消耗时间随属性数量的增加而越来越多。其中，文献[18]方案的增长率最高，随着属性数量的增加其消耗时间远超其他方案，而本文方案的增长率最小。使用椭圆曲线中简单标量乘法的本文方案与文献[24]方案在数据拥有者加密时的消耗时间远少于其他方案，且本文方案效率更高。

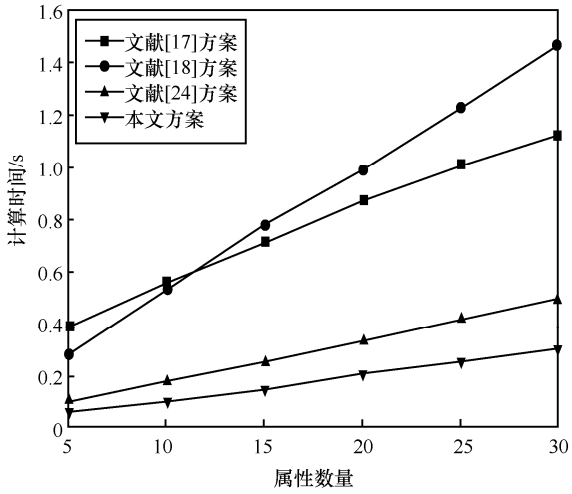

图 3　数据拥有者加密计算时间比较

数据用户解密计算时间比较如图 4 所示。从图 4 中可以看出，没有将解密计算外包的文献[18,24]方案中数据用户在执行解密计算时，其消耗时间会随属性数量的增加而越来越多；而在设计了外包计算的本文方案与文献[17]方案中，用户在解密计算时的开销不会因为访问策略的复杂化而增加用户计算负担，并且本文方案解密消耗时间更少。

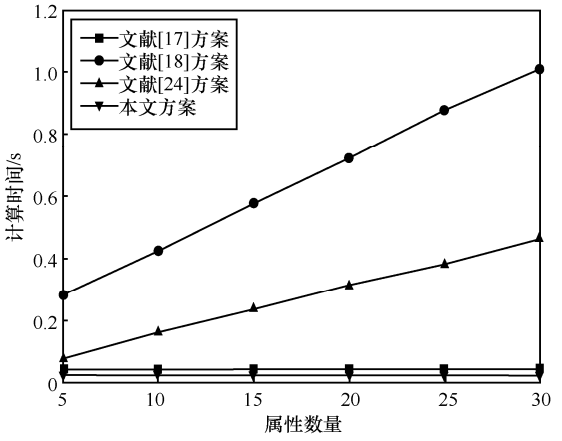

图 4　数据用户解密计算时间比较

外包解密时间比较如图 5 所示。从图 5 可以明显看出，避免了双线性对配对操作的本文方案在外包解密操作时所消耗时间要远少于文献[17]方案，并且随着访问结构中属性数量的增加，这种差距将越来越明显，所以本文方案在外包解密算法中效率更高。

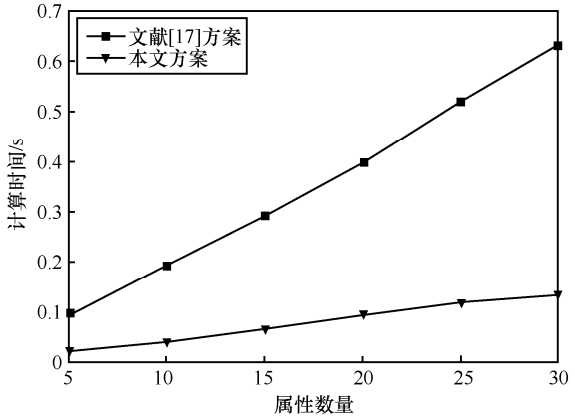

图 5　外包解密计算时间比较

8　结束语

本文提出了一种雾计算中基于无配对 CP-ABE 可验证的访问控制方案，利用椭圆曲线加密中的简单标量乘法替代双线性配对计算，并设计安全高效的外包计算，减少了用户解密时计算开销。在雾节点中部署区块链不仅可以为雾计算提供可信和安全的环境，还可以根据区块链防篡改可溯源的特性实现对访问事务的正确性验证并记录访问授权过程。对本文方案的安全性进行了分析，仿真实验结果表明，本文方案的用户解密时间消耗较小且稳定，整体计算效率更高。但本文方案仍存在一些不足之处。现有访问策略会显式地附加在密文上，或者仅实现了部分隐藏以防止公开可见，会出现接收者的持有属性被窥探的情况，导致用户的隐私泄露。因此需要一种完全隐藏属性的访问策略，能够将属性信息完全隐藏在访问策略中，使任何有价值的属性信息都不透露给未授权的接收者。

参考文献：

[1] 贾维嘉, 周小杰. 雾计算的概念、相关研究与应用[J]. 通信学报, 2018, 39(5): 153-165.
JIA W J, ZHOU X J. Concepts, issues, and applications of fog computing[J]. Journal on Communications, 2018, 39(5): 153-165.

[2] GUO R, ZHUANG C Y, SHI H X, et al. A lightweight verifiable outsourced decryption of attribute-based encryption scheme for blockchain-enabled wireless body area network in fog computing[J]. International Journal of Distributed Sensor Networks, 2020, 16(2): 155014772090679.

[3] JIANG J F, TANG L Y, GU K, et al. Secure computing resource allocation framework for open fog computing[J]. The Computer Journal, 2020, 63(4): 567-592.

[4] SHAHID M H, HAMEED A R, ISLAM S U, et al. Energy and delay efficient fog computing using caching mechanism[J]. Computer Communications, 2020, 154: 534-541.

[5] DESIKAN K E S, KOTAGI V J, MURTHY C S R. Topology control in fog computing enabled IoT networks for smart cities[J]. Computer Networks, 2020, 176: 107270.

[6] VILELA P H, RODRIGUES J J P C, RIGHI R D R, et al. Looking at fog computing for E-health through the lens of deployment challenges and applications[J]. Sensors, 2020, 20(9): 2553.

[7] FERRAIOLO D, CUGINI J, KUHN D R. Role-based access control (RBAC): features and motivations[C]//Proceedings of 11th Annual Computer Security Application Conference. Piscataway: IEEE Press, 1995: 241-248.

[8] ZHANG P Y, ZHOU M C, FORTINO G. Security and trust issues in fog computing: a survey[J]. Future Generation Computer Systems, 2018, 88: 16-27.

[9] BETHENCOURT J, SAHAI A, WATERS B. Ciphertext-policy attribute-based encryption[C]//2007 IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2007: 321-334.

[10] WANG H, ZHENG Z H, WU L. New large-universe multi-authority ciphertext-policy ABE scheme and its application in cloud storage systems[J]. Journal of High Speed Networks, 2016, 22(2): 153-167.

[11] LIANG K T, SUSILO W. Searchable attribute-based mechanism with efficient data sharing for secure cloud storage[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(9): 1981-1992.

[12] LEWKO A, WATERS B. Decentralizing attribute-based encryption[C]// Advances in Cryptology – EUROCRYPT 2011. Berlin: Springer, 2011: 568-588.

[13] HORVATH M. Attribute-based encryption optimized for cloud computing[C]//Theory and Practice of Computer Science. Berlin: Springer, 2015, DOI:10.1007/978-3-662-46078-8_47.

[14] HUR J. Improving security and efficiency in attribute-based data sharing[J]. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(10): 2271-2282.

[15] LIANG K T, SUSILO W. Searchable attribute-based mechanism with efficient data sharing for secure cloud storage[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(9): 1981-1992.

[16] WANG S L, LIANG K T, LIU J K, et al. Attribute-based data sharing scheme revisited in cloud computing[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(8): 1661-1673.

[17] LI J G, WANG Y, ZHANG Y C, et al. Full verifiability for outsourced decryption in attribute based encryption[J]. IEEE Transactions on Services Computing, 2020, 13(3): 478-487.

[18] ZHANG K, LI H, MA J F, et al. Efficient large-universe multi-authority ciphertext-policy attribute-based encryption with white-box traceability[J]. Science China Information Sciences, 2017, 61(3): 1-13.

[19] FREEMAN D, SCOTT M, TESKE E. A taxonomy of pairing-friendly elliptic curves[J]. Journal of Cryptology, 2010, 23(2): 224-280.

[20] SCOTT M. On the efficient implementation of pairing-based protocols[C]// Cryptography and Coding. Berlin: Springer, 2011: 296-308.

[21] PONTIE S, MAISTRI P, LEVEUGLE R. Dummy operations in scalar multiplication over elliptic curves: a tradeoff between security and performance[J]. Microprocessors & Microsystems, 2016, 47: 23-36.

[22] CHEVALLIER-MAMES B, CORON J S, MCCULLAGH N, et al. Secure delegation of elliptic-curve pairing[C]//Lecture Notes in Computer Science. Berlin: Springer, 2010: 24-35.

[23] CHEN X F, SUSILO W, LI J, et al. Efficient algorithms for secure outsourcing of bilinear pairings[J]. Theoretical Computer Science,

2015, 562: 112-121.

[24] ODELU V, DAS A K. Design of a new CP-ABE with constant-size secret keys for lightweight devices using elliptic curve cryptography[J]. Security and Communication Networks, 2016, 9(17): 4048-4059.

[25] MAESA D D F, MORI P, RICCI L. Blockchain based access control[C]//Distributed Applications and Interoperable Systems. Berlin: Springer, 2017: 206-220.

[26] DAGHER G G, MOHLER J, MILOJKOVIC M, et al. Ancile: privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology[J]. Sustainable Cities and Society, 2018, 39: 283-297.

[27] DORRI A, KANHERE S S, JURDAK R, et al. Blockchain for IoT security and privacy: the case study of a smart home[C]//2017 IEEE International Conference on Pervasive Computing and Communications Workshops. Piscataway: IEEE Press, 2017: 618-623.

[28] 谢绒娜, 李晖, 史国振, 等. 基于区块链的可溯源访问控制机制[J]. 通信学报, 2020, 41(12): 82-93.
XIE R N, LI H, SHI G Z, et al. Blockchain-based access control mechanism for data traceability[J]. Journal on Communications, 2020, 41(12): 82-93.

[29] 应作斌, 斯元平, 马建峰, 等. 基于区块链的分布式 EHR 细粒度可追溯方案[J]. 通信学报, 2021, 42(5): 205-215.
YING Z B, SI Y P, MA J F, et al. Blockchain-based distributed EHR fine-grained traceability scheme[J]. Journal on Communications, 2021, 42(5): 205-215.

[作者简介]

董江涛（1981− ），男，河北石家庄人，中国电子科技集团公司第五十四研究所高级工程师，主要研究方向为航天地面运控应用。

闫沛文（1994− ），男，河北张家口人，河北大学硕士生，主要研究方向为信息安全、访问控制、雾计算等。

杜瑞忠（1975− ），男，河北保定人，博士，河北大学教授、博士生导师，主要研究方向为可信计算、信息安全等。