一种大属性域版本控制的云安全用户属性动态撤销策略

党鲜玲, 郭银章

[1] 太原科技大学 计算机科学与技术学院 太原 中国 030024

摘要 密文策略属性基加密(ciphertext-policy attribute-based encryption, CP-ABE) 作为一种一对多的数据加密技术, 因能实现密文数据安全和细粒度的权限访问控制而引起学术界的广泛关注。尽管目前在该领域已取得了一些研究成果, 然而, 大多数 CP-ABE 方案均基于小属性域, 系统属性同时被多个用户共享而难以实现动态的属性撤销, 现有的属性撤销机制在功能复杂性、计算高效性、以及抗合谋攻击安全性方面存在的问题都成为它在实际应用中的障碍。针对上述问题, 提出一种大属性域版本控制的云安全用户属性动态撤销策略。该方案在密文策略属性加密中构造属性及用户版本密钥, 通过更新属性版本密钥实现用户属性撤销, 更新用户版本密钥实现用户撤销。由此避免了基于重加密实现撤销带来的计算和通信开销。该方案基于 q-DBPBDHE 假设, 在随机预言模型下证明是静态性安全的。最后, 对方案进行了性能分析与实验验证, 实验结果表明: 在保证密文前后向安全性的前提下, 该方案可以实现动态的用户属性撤销和用户撤销且可以抵制多重合谋攻击, 较同类方案本文方案具有较优的功能特性和计算效率。此外, 所提方案基于大属性域, 在实际应用中更加灵活。

关键词 云计算; 大属性域; 版本控制; 属性撤销; 访问控制

中图法分类号 TP309 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2023.01.12

A Dynamic Revocation Strategy of Cloud Security User Attributes for Large Attribute Domain Version Control

DANG Xianling, GUO Yinzhang

[1]Department of Computer Science and Technology, Taiyuan University of Science and Technology (TUST), Taiyuan 030024, China

Abstract Ciphertext Policy Attribute-Based Encryption (CP-ABE), as a one-to-many public key encryption method, has attracted extensive attention in the academic world. Because this technique can protect data security, as well as provide fine-grained data access control. Although some research achievements have been made in this field, However, in most of existing CP-ABE schemes are based on small universe of attributes. And the attribute level dynamic revocation is an important challenge because the system attributes are shared by multiple users at the same time. There are some obstacles towards practical applications in most of the existing attribute revocation mechanisms, including the aspects of functionality, efficiency and anti-collusion attack security. In order to resolve the above mentioned issue, the paper proposes a scheme which is a dynamic revocation strategy of cloud security user attributes based on large universe version control. In the proposal, the attribute version key and user version key in the construction of the ciphertext policy attribute based encryption. Only the corresponding attribute version keys need to be update when the user attributes are revoked. similarly, only the user version key needs to be updated when the user is revoked. Therefore the expensive computation and communication overhead caused by ciphertext update based on data re-encryption can be effectively avoided. Based on the assumption of q-DBPBDHE, the scheme is proved that is statically secure in the random oracle model. Finally, the performance analysis and experimental verification are carried out, and the experimental results show that the proposed scheme can dynamically implement attribute level user revocation and user revocation, while ensuring multiple anti-collusion attacks under the premise of guaranteeing forward and backward security for ciphertext. Comparisons are provided between the proposal scheme and other lattice-based related works, analysis shows that our scheme has some advantages in terms of functional characteristics and computational efficiency. In addition, the proposed scheme supports large universe of attributes, which makes it more flexible for practical applications.

Key words cloud computing; large universe; version control; attribute revocation; access control

通讯作者: 郭银章, 博士, 教授, Email: guoyinzhang@263.net。

本课题得到山西省应用基础研究项目(No. 201901D111266)资助。

收稿日期: 2021-09-28; 修改日期: 2022-01-14; 定稿日期: 2022-11-04

1 引言

随着云计算[1]规模化和集约化的发展使得云安全问题日益凸显, 访问控制[2]是云安全的核心技术。在云计算环境下, 用户对放置在云存储服务器中的数据资源失去控制, 用户与云服务商之间缺乏信任机制, 云服务商可能会擅自将用户的敏感数据泄露给竞争对手以获取不当利益。2006 年, Sahai 等人[3]首次提出基于属性的加密(attribute-based encryption, ABE)方案, 为了实现更加灵活的访问策略, 进一步提出了密文策略属性基加密[4] (cipher-text-policy attribute-based encryption, CP-ABE)方案和密钥策略属性基加密[5] (key-policy attribute-based encryption, KP-ABE)方案。在 CP-ABE 方案中, 密文和访问结构相关, 而密钥和属性相关, 只有当用户的属性满足嵌入的访问结构的属性时才能解密密文。KP-ABE 中, 这种关系恰好相反, 即密文和属性相关, 而密钥和访问结构属性相关。与 KP-ABE 比, CP-ABE 更适合于大规模云计算环境下的访问控制, 对于保护云中数据资源发挥重要作用[6]。然而, CP-ABE 在实际应用中大多基于小属性域[7], 在系统建立之处需确定属性域, 由于系统在运行过程中大量的用户属性发生变更, 严重影响了方案的扩展性; CP-ABE 在应用过程中的另一个挑战是属性撤销。由于云存储系统中拥有大量的用户, 在系统运行过程中存在一些用户的相关属性会发生改变, 或者一些用户私钥可能被泄露等问题, CP-ABE 中每一个属性可能被不同用户共享, 这意味着撤销任何属性都可能影响其他用户。因此, 如何设计一种支持用户和用户属性撤销的 CP-ABE 方案极其关键。

为了解决属性域的问题, Lewko 和 water [8]最先引入小属性域和大属性域的分类解决这个问题。在小属性域方案中, 属性域必须在初始阶段设置, 系统公共参数长度随系统的属性个数线性增加。而大属性域方案, 整个系统的属性域无需在系统建立时确定, 系统中任何一个字符串都能够作为新属性加入系统, 公钥长度也与系统的属性个数无关, 使得方案有良好的扩展性。随后, Rouselakis 和 Waters [9]在素数阶群下提出一个支持大属性域的 CP-ABE 方案。Zhang 等人[10]基于合数阶群提出了一个支持大属性域和高表达力的策略隐藏 CP-ABE 方案, 并证明了在标准模型下方案是完全安全的。Ning 等人 [7]提出了一个支持大属性域的可追责 CP-ABE 方案。然而, 上述文献[5-8]方案并不能实现用户属性和用户撤销。

针对撤销问题, 2006 年, Pirretti 等人[11]最早提出 ABE 属性撤销方案, 通过给每个属性设置一个有效期, 授权机构周期性地颁发属性的最新版本, 通过更新某个属性的版本实现对用户属性的撤销。Bethencourt 等人[12]用属性的终止时间代替有效期, 来限制密钥的使用时间。这 2 种方案均不支持属性或用户的即时撤销, 除此之外, 撤销方案均存在一个脆弱性窗口, 影响方案的影响方案的前向安全(forward secrecy)和后向安全 (backward secrecy)[13]。

Wu 等人[14]基于版本号和代理重加密技术提出一种具有用户属性和用户撤销的 CP-ABE 方案, 将版本号和系统主密钥、用户密钥和重加密密文进行关联, 当发生用户属性撤销时, 授权机构更新版本号, 可信第三方同步需更新所有与版本号相关的密钥和密文, 有效避免了撤销用户与未撤销用户的合谋攻击。Zhao 等人[15]提出一种云存储环境下无密钥托管的属性撤销的 CP-ABE 方案, 该方案通过更新属性版本密钥的方式实现用户属性撤销, 利用中央授权机构实现用户撤销。Yan 等人 [16]提出一种细粒度访问控制的属性撤销 CP-ABE 方案, 该方案通过构造属性及用户版本密钥执行用户属性和用户撤销操作, 但该方案不能抵御合谋攻击[17-18]。

Zhang 等人[19]提出了一种支持大属性域可撤销的 CP-ABE 方案, 但该方案仅可实现用户撤销。Lian 等人[20]基于广播属性加密思想提出一种支持大属性域可撤销 CP-ABE 方案。Zhao 等人 [21]提出一种云环境下支持大属性域用于电子医疗记录的隐私保护的 CP-ABE 方案, 利用代理重加密技术完成属性撤销。Hur 等人 [22]提出一种通过使用密钥加密密钥(key encrypting key, KEK)树来实现细粒度属性撤销的 CP-ABE 方案。Li 等人[23]通过优化 Hur 等人[22]的方案, 基于属性群和KEK树及半可信第三方为每个属性生成相应的属性组密钥, 通过属性组实现属性撤销操作。Liu 等人[24]基于 KEK 树提出一种支持大属性域的可撤销 CP-ABE 方案, 重加密密文和密钥维护代价高。

综合分析国内外相关研究的优点和不足, 本文提出一种基于大属性域版本控制的动态撤销 CP-ABE 方案。该方案通过更新为用户及属性设置的版本密钥实现用户属性和用户撤销, 避免了基于重加密技术实现撤销带来的巨大开销。同时, 该方案中用于生成属性私钥的秘密参数由属性授权机构随机选取, 并与用户身份标识相关联, 可有效抵抗合谋攻击。所提方案在随机预言机模型下基于 q-Decisional Parallel Bilinear Diffiffiffie Hellman Exponent Assumption (q-DBPBDHE)假设对方案进行了

选择明文攻击的安全性证明。最后,对方案进行了性能分析与实验验证, 实验结果表明: 本文方案在保证密文前后向安全性的前提下, 可以有效抵抗多重合谋攻击。

2 理论基础知识

2.1 双线性映射

假设 G 和 G_T 是 2 个阶为素数 p 的循环群, g 是群 G 的生成元。双线性映射 $e:G\times G\to G_T$ 满足以下性质:

1) 双线性: 对 $\forall g_1, g_2\in G$ $a,b\in Z_P$, 有 $e\left(g_1^a,g_2^b\right)=e\left(g_1,g_2\right)^{ab}$。

2)非退化性: 存在 $g\in G$, 使得 $e\left(g,g\right)\neq 1$。

3)可计算性: 对 $\forall g_1, g_2\in G$, 存在多项式时间算法有效计算 $e\left(g_1,g_2\right)$。

2.2 线性秘密共享方案

参与者组成的集合 $\mathbb{P}$ 上的秘密共享方案 $\prod$ 如果满足以下两个条件, 则 $\prod$ 定义为在 Z_p (p 是一个素数)域上的线性秘密共享方案 (linear secret sharing scheme, LSSS)。

1)每个参与者的秘密份额构成 Z_p 域上的向量。

2)对于每个秘密共享方案 $\prod$ 存在一个 $l\times n$ 共享生成矩阵 $\boldsymbol{M}$ 和单射函数 ρ, 对于矩阵 $\boldsymbol{M}_{l\times n}$ 中的每一行 $i=1,2,\cdots,l$, 映射 $\rho:\{1,2,...,l\}\to\mathbb{P}$ 把 $\boldsymbol{M}_{l\times n}$ 的每一行映射到参与者 $\rho(i)$。随机选择参数 $r_2,...,r_n\in Z_P$, 构造向量 $\mathrm{v}^{\mathrm{T}}=\left(s,r_2,r_3,...,r_n\right)^{\mathrm{T}}$, 向量 M$v$ 为共享秘密值 s ($s\in Z_P$)关于 $\prod$ 的 l 个秘密份额, $\lambda_i=\left(\boldsymbol{M}v\right)_i$ 表示参与者 $\rho(i)$ 所获得的共享秘密份额。

上述 LSSS 方案具有线性重构特性, 假设 $\prod$ 是访问策略 $\mathbb{A}$ 的一个 LSSS 秘密共享方案, 令 $S\in\mathcal{A}$ 是一个访问授权集合, 定义 $I=\{i:\rho(i)\in S\}$ $I\subset\{1,2,\cdots,l\}$。存在常量 $\{\omega_i\in Z_P\}_{i\in I}$ 对任意共享秘密 $\{\lambda_i\}$, 使 $\sum_{i\in I}\left(\omega_i\lambda_i\right)=\sum_{i\in I}\omega_i\left(\boldsymbol{M}_i v\right)=\boldsymbol{\varepsilon}\cdot v=s$ 成立, 其中 $\boldsymbol{\varepsilon}=(1,0,...,0)$。

2.3 判定性 q-DBPBDHE 假设

假设给定一个五元组 $\left(G,G_T,e,p,g\right)$, 其中, G 和 G_T 均表示阶为素数 p 的双线性群, g 是群 G 的生成元, $e:G\times G\to G_T$ 表示双线性映射。$s,a,b_1,b_2,\cdots,b_q\in Z_p$ 表示群 G 中的随机参数, $R\in G_T$ 表示群 G_T 中的随机元素。并设置

$$D=\left(\begin{array}{c} G,p,e,g,g^s,\left\{g^{a^i}\right\}_{i\in[2q],i\neq q+1},\left\{g^{b_j a^i}\right\}_{(i,j)\in[2q,q],i\neq q+1},\\ \left\{g^{s/b_i}\right\}_{i\in[q]},\left\{g^{sa^i b_j/b_{j'}}\right\}_{(i,j,j')\in[q+1,q,q],j\neq j'} \end{array}\right)$$

对敌手 $\mathcal{A}$ 而言, 区分 $e\left(g,g\right)^{a^{q+1}s}$ 与 R 是困难的。当 $\left|\Pr\left[\mathcal{B}\left(D,e\left(g,g\right)^{a^{q+1}s}=0\right)\right]-\Pr\left[\mathcal{B}\left(D,R\right)=0\right]\right|\geqslant\boldsymbol{\varepsilon}$ 则算法 $\mathcal{B}$ 有优势 $\boldsymbol{\varepsilon}$ 来解决群 G_T 下 q-DBPBDHE 假设。若无多项式算法以不可忽略的优势解决 q-DBPBDHE 问题, 那么 q-DBPBDHE 假设在群 G、G_T 中成立。

3 方案设计

3.1 系统模型

系统主要由半可信第三方(semi-trusted mediator, SM)、k 个属性授权机构(attribute authority, AA)、云存储服务器(cloud storage server, CSS)、代理解密服务器(proxy decryption server, PDS)、数据属主(date owner, DO)和数据用户(date user, DU) 6 个实体构成。半可信第三方 SM 调用系统初始化算法生成全局公共参数(global public parameters, GP)并将 GP 分发给属性授权机构 AA、数据属主 DO 和数据用户 DU。k 个属性授权机构 AA 负责生成用户私钥, 在方案设计中代理解密服务器 PDS 所持有的代理解密密钥并不能将密文还原成明文, 而 DO 也始终以密文的形式在 CSS 上存储数据。本文所提方案系统模型如图 1 所示。

3.2 方案构建

3.2.1 系统初始化

该阶段包含 *GlobalSetup*、*UserSetup* 和 *AASetup* 3 个算法。

1) $GlobalSetup\left(\lambda\right)\to GP$: SM 运行该算法, 输入安全参数 λ, 选择两个阶为素数 p 的乘法循环群 G 和 G_T, 且存在双线性映射 $e:G\times G\to G_T$。系统属性域用 S 表示, 系统属性授权机构集用 *Auth* 表示, 系统用户身份标识集用 U 表示, 针对大属性域属性撤销问题, 随机选择三个哈希函数 H、F 和 T。$H:S\to G$ 将属性 $att_i\in S$ 映射到 G。$F:U\to G$ 将用户身份标识 $uid\in U$ 映射到 G。$T:S\to I_{Auth}$ 将属性映射到 AA。最后输出全局公共参数 $GP=\left(G,G_T,p,g,S,I_{Auth},U,H,F,T\right)$。

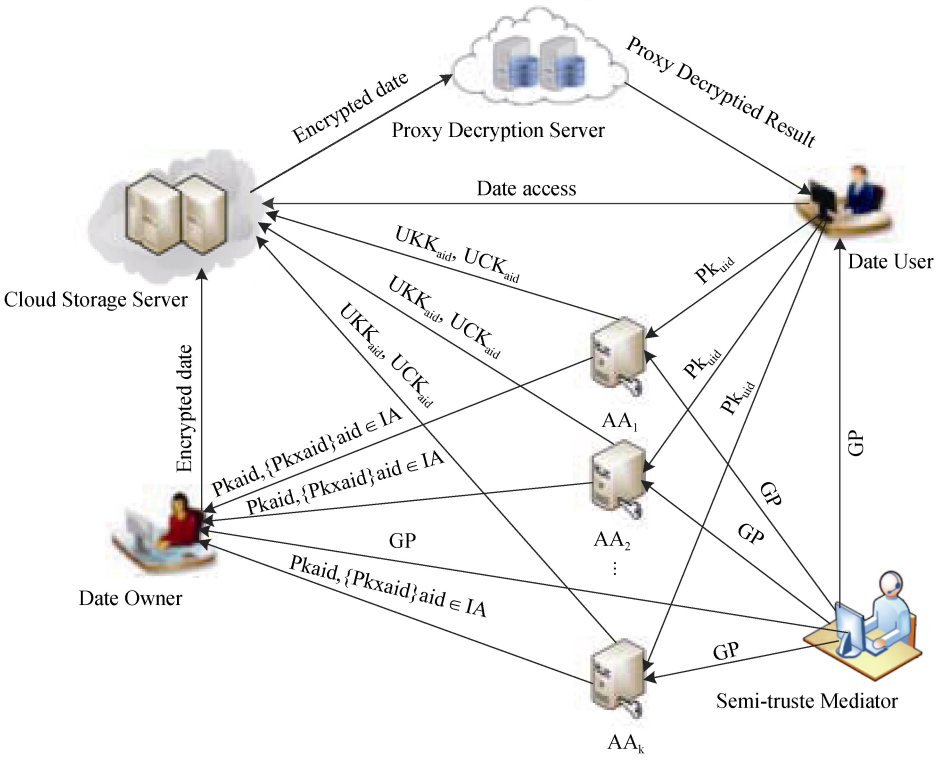

图 1　系统模型

Figure 1　System model

2) $UserSetup(uid,GP)\rightarrow(PK_{uid},\ SK_{uid})$: DU 运行该算法, 选择随机数 $z_{uid}\in Z_p^*$, 计算公钥 $PK_{uid}=\left(g^{z_{uid}},H(uid)^{\frac{1}{z_{uid}}}\right)$, 并将 $SK_{uid}=z_{uid}$ 作为私钥秘密保存。

3) $AASetup(aid,GP)\rightarrow(PK_{aid},SK_{aid},VK_{att},PK_{att})$ AA 运行该算法, 选择两个随机数 $\alpha_{aid},\beta_{aid}\in Z_p^*$, 为每个属性 att 设置属性版本 $VK_{att}=v_{att}$, 公开属性密钥 $PK_{att}=g^{v_{att}}$, 输出属性授权机构 AA 公钥 $PK_{aid}=\left(e(g,g)^{\alpha_{aid}},g^{1/\beta_{aid}},g^{\gamma_{aid}/\beta_{aid}}\right)$ 和属性授权机构 AA 私钥 $SK_{aid}=(\alpha_{aid},\beta_{aid})$。

3.2.2　属性私钥生成

$KeyGen(GP,S_{uid,aid},PK_{uid},SK_{aid},PK_{att})\rightarrow(SK_{uid,aid})$ AA 运行该算法, 随机选择 $k_{uid,aid},t_{uid,aid}\in Z_p^*$, 设置用户版本密钥 $VK_{uid,aid}=g^{k_{uid,aid}t_{uid,aid}s/(SK_{uid}r_x)}$. $g^{s/\left(SK_{uid}\beta_{\delta(x)}r_x\right)}$ 输出属性私钥 $SK_{uid,aid}=\left(TK_{uid,aid},L_{uid,aid}\right)$, 并发送给 CCS, 其中 $TK_{uid,aid}=g^{\alpha_{\delta(x)}}\cdot\left(g^{v_{att}}H(x)\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}\cdot F(uid)^{\beta_{\delta(x)}}$, $L_{uid,aid}=g^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}$。

3.2.3　数据加密

$Encrypt(GP,PK_{aid},PK_{att},m,(M,\rho))\rightarrow CT$: DO 运行该算法, 定义函数 $\delta(x)=T(\rho(x))$, 将矩阵 $\boldsymbol{M}$ 的行映射到特定的 AA, 即 $\delta:\{1,2,...,l\}\rightarrow A$。然后随机选择 $v=(s,v_2,...,v_n)^{\mathrm{T}}$、$\omega=(0,\omega_2,...,\omega_n)^{\mathrm{T}}\in Z_p^n$ 两个向量, 对 $\forall i\in\{1,2,...,l\}$ 计算 $\lambda_i=(\boldsymbol{M}_i\cdot v)$, $\omega_i=(\boldsymbol{M}_i\cdot\omega)$ 和 $C_0=m\cdot e(g,g)^s$, 其中 $\boldsymbol{M}_i$ 是矩阵 $\boldsymbol{M}$ 的第 i 行。随机选择参数 $r_x\in Z_p$, 计算 $C_{1,i}=e(g,g)^{\lambda_x/SK_{uid}+\alpha_{\delta(x)}r_x}$, $C_{2,i}=g^{-r_x}$, $C_{3,i}=g^{\omega_x/SK_{uid}+\beta_{\delta(x)}r_x}$ $C_{4,i}=H(\rho(x))^{r_x}$ 和 $C_{5,i}=g^{v_{att}r_x}$, 最后输出密文

$$CT=\left(C_0,C_{1,i},C_{2,i},C_{3,i},C_{4,i},C_{5,i}\right)_{i\in\{1,2,\cdots,l\}}\qquad(1)$$

3.2.4　数据解密

该阶段包含 *ProxyDec*、*UserVerandDec* 和 *Decrypt* 3 个算法。

1) $ProxyDec(CT,TK_{uid,aid})\rightarrow parCT$: PDS 运行该算法, 若用户属性满足密文访问策略, 下标集合 $I=\{i:\rho(i)\in S\}$, 且满足 $I\subset\{1,2,...,l\}$, 则存在常量

$\{c_i \in Z_p\}_{i\in I}$ 使得 $\sum_{i\in I} c_i M_i=(1,0,...,0)$ 成立, 计算半解密密文 $parCT$ 。

$$ParCT=\prod_{i\in s}\left(\begin{matrix}C_{1,i}\cdot e\left(L_{uid,aid},C_{2,i}\right)\cdot e\left(F\left(uid\right),C_{3,i}\right)\\ \cdot e\left(TK_{uid,aid},C_{4,i}\cdot C_{5,i}\right)\end{matrix}\right)^{c_i} \tag{2}$$
$$=e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}s/SK_{uid}}$$

2) $UserVerandDec\left(C_{5,i},VK_{uid,aid}\right)/parCT\rightarrow CT^{'}$ CSS 运行该算法, 并将验证密文 $CT^{'}$ 发送 DU。

$$\begin{aligned}CT^{'}&=e\left(C_{5,i},VK_{uid,aid}\right)/parCT\\&=e\left(g,g\right)^{s/SK_{uid}}\end{aligned} \tag{3}$$

3) $Decrypt\left(SK_{uid},CT^{'}\right)\rightarrow m$: DU 运行该算法, 获得明文。

$$m=\frac{C_0}{CT^{'SK_{uid}}}=\frac{m\cdot e\left(g,g\right)^{s}}{\left(e\left(g,g\right)^{s/SK_{uid}}\right)^{SK_{uid}}}。 \tag{4}$$

3.2.5 用户属性撤销

用户属性撤销是对用户属性集合中的某些属性的撤销, 用户属性撤销操发生后, 该用户失去该属性的访问权限, 但不影响该用户其它属性的权限。同时, 也不影响拥有该属性但未发生撤销的其他用户对该属性的访问权限。通过 $UpkeyGen$、$UpTK_{uid,aid}$ 和 $UpCT$ 3 个算法实现用户属性撤销。

1) $UpkeyGen\left(v_{att_x},\ PK_{att_x},\right)\rightarrow\left(\begin{matrix}v^{'}_{att_x},PK^{'}_{att_x},\\ UKK_{aid},CUK_{aid}\end{matrix}\right)$

AA 运行该算法, 当用户属性 att_x (对应属性版本 $VK_{att_x}=v_{att_x}$)被撤销时, AA 随机选择 $v^{'}_{att_x}\in Z_p$ 且 $v^{'}_{att_x}\neq v_{att_x}$, 设置新的属性版本 $VK_{att^{'}_x}=v^{'}_{att_x}$, 同步更新的属性公钥为 $PK^{'}_{att_x}=g^{v^{'}_{att_x}}$。计算密钥更新密钥 $UUK_{aid}=g^{\left(v^{'}_{att_x}-v_{att_x}\right)k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}$ 和密文更新密钥 $CUK_{aid}=g^{r_x\left(v^{'}_{att_x}-v_{att_x}\right)}$ 并发送给 CCS, 在 CSP 端完成转换密钥 $TK_{uid,aid}$ 和密文组件 $C_{5,x}$ 的更新任务。

2) $UpTK_{uid,aid}\left(SK_{uid,aid},UKK_{aid}\right)\rightarrow\left(TK^{'}_{uid,aid}\right)$: PDS 运行该算法, 更新未撤销该属性的用户的属性私钥组件:

$$\begin{aligned}TK^{'}_{uid,aid}&=TK_{uid,aid}\times UUK_{aid}\\&=g^{\alpha_{\delta(x)}}\cdot\left(g^{v_{att_x}}H\left(x\right)\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}\cdot F\left(uid\right)^{\beta_{\delta(x)}}\\&\times g^{\left(v^{'}_{att_x}-v_{att_x}\right)k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}\\&=g^{\alpha_{\delta(x)}}\cdot\left(g^{v^{'}_{att_x}}H\left(x\right)\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}\cdot F\left(uid\right)^{\beta_{\delta(x)}}\end{aligned} \tag{5}$$

3) $UpCT\left(CT,CUK_{aid}\right)\rightarrow\left(\overline{CT}\right)$: CCS 更新密文中关联的 att_x 的密文组件 $\overline{C}_{5,i}=C_{5,i}\times CUK_{aid}$,密文更新为

$$\overline{CT}=\left(\begin{matrix}\overline{C}_0=C_0,\\ \forall i\in\{1,2,...,l\}:\overline{C}_{1,i}=C_{1,i},\overline{C}_{2,i}=C_{2,i},\\ \qquad\qquad\overline{C}_{3,i}=C_{3,i},\overline{C}_{4,i}=C_{4,i}\\ if\quad \rho\left(i\right)\neq att_x:\overline{C}_{5,i}=C_{5,i}\\ if\quad \rho\left(i\right)=att_x:\overline{C}_{5,i}=C_{5,i}\times CUK_{aid}\end{matrix}\right) \tag{6}$$

3.2.6 用户撤销

当系统中某些用户被判定为“非法用户”或因其他原因退出系统时, 需撤销该用户在系统中的所有访问权限以保证密文的安全性。本文方案中, 当发生用户撤销时, AA 仅需对该用户版密钥进行更新, 即可完成用户撤销。CCS 运行 $UserVerandDec$ 算法生成验证密文 $CT^{'}$,

$$\begin{aligned}CT^{'}&=e\left(C_{5,i},VK^{'}_{uid,aid}\right)/parCT\\&=\frac{e\left(g,g\right)^{s/SK_{uid}}\cdot e\left(g,g\right)^{\beta_{\delta(x)}k^{'}_{uid,aid}t^{'}_{uid,aid}s/SK_{uid}}}{e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}s/SK_{uid}}}\end{aligned} \tag{7}$$

由于 $t^{'}_{uid,aid}$、$k^{'}_{uid,aid}$ 和 $t_{uid,aid}$、$k_{uid,aid}$ 值不同而不能获得 $CT^{'}$, 用户无法进一步解密获取密文。

4 安全性分析

4.1 安全性证明

定理 1. 假设 RW 方案[9]基于 q-DPBDHE 假设在随机预言机模型下是静态性安全的, 则本文提出的方案在随机预言机模型下也是静态性安全的。

假设存在敌手 $\mathcal{A}$ 以不可忽略的优势 $\varepsilon=Adv_A$ 选择性地攻破本文方案, 则便可构造一个模拟器 $\mathcal{B}$ 以同样优势攻破 RW 方案, $\mathcal{C}$ 是 RW 方案中与 $\mathcal{B}$ 交互的挑战者。

系统建立: $\mathcal{B}$ 从 RW 方案的 $\mathcal{C}$ 中获得公开参数 GP, 并发送给 $\mathcal{A}$。

询问阶段: $\mathcal{A}$ 指定腐化属性授权机构 AA, $\mathcal{B}$ 将 $\mathcal{A}$ 传来的消息发送给 $\mathcal{C}$。

AA 公钥询问: $\mathcal{A}$ 选择未被腐化的属性授权机构 AA 并询问其公钥 PK_{aid}。$\mathcal{B}$ 收到 $\mathcal{C}$ 传送的 PK_{aid}

并发送给 $\mathcal{A}$。

用户私钥询问: $\mathcal{A}$ 向 $\mathcal{B}$ 提交用户身份标识 $uid \in U$, 并询问对应的用户公/私钥对。

属性私钥询问: $\mathcal{A}$ 向 $\mathcal{B}$ 提交的查询条件 $\{uid, S_{uid,aid}\}$ 并询问属性私钥 $SK_{uid,aid}$。

密文询问: $\mathcal{A}$ 提提交两个等长的消息 m_0、m_1 及访问策略 $\mathbb{A}^*$ 询问挑战密文。

更新密钥询问: $\mathcal{A}$ 提交查询条件 $\{uid, S_{uid,aid}\}$ 以及被撤销属性 att_x, 并发起关联该的更新密钥询问。

挑战者响应阶段: 基于 RW 方案 $\mathcal{C}$ 对 $\mathcal{B}$ 返回的询问响应, 模拟器执行如下操作。

属性授权机构公钥询问响应: 对任意一个未被腐化的属性授权机构 AA, $\mathcal{B}$ 随机选择参数 $\alpha_{aid}, \beta_{aid} \in Z_p^*$, 生成公钥 PK_{aid} 并发送给 $\mathcal{A}$。

用户私钥询问响应: 依据 $\mathcal{A}$ 提交的用户身份标识 $uid \in U$, $\mathcal{B}$ 返回用户公/私钥对 $\{PK_{uid}, SK_{uid}\}$。

属性私钥询问响应: 依据查询条件 $\{uid, S_{uid,aid}\}$, $\mathcal{B}$ 随机选择 $k_{uid,aid}$, $t_{uid,aid} \in Z_p^*$ 执行算法 $KeyGen$, 输出属性私钥 $SK_{uid,aid}$ 并发送给 $\mathcal{A}$。

密文询问响应: 对于 $i \in \{1,2,...,l\}$, 模拟器计算 $C_{5,i} = C_{2,x}^{v_{att}} = g^{v_{att} r_x}$, 输出挑战密文 CT^* 发送给敌手。

更新密钥询问响应: 依据 $\mathcal{A}$ 提交的查询条件 $\{uid, S_{uid,aid}\}$ 以及被撤销属性 att_x, $\mathcal{B}$ 随机选择 $v'_{att_x} \in Z_p$, 更新 att_x 的新属性版本 $VK_{att_x'} = v'_{att_x}$ 和属性公钥 $PK'_{att_x} = g^{v'_{att_x}}$, 然后运行算法 $UpkeyGen$ 输出密钥更新密钥 UUK_{aid} 和密文更新密钥 CUK_{aid} 并发送给 $\mathcal{A}$。

用户私钥更新和密文更新询问响应: 依据 $\mathcal{A}$ 获得的撤销属性 att_x 关联的 UUK_{aid} 和 CUK_{aid}, $\mathcal{B}$ 运行 $UpTK_{uid,aid}$ 算法和 $UpCT$ 算法, 将最终结果均发送给 $\mathcal{A}$。

猜测阶段: $\mathcal{A}$ 最终输出对 b 的猜测 $b' \in \{0,1\}$, $\mathcal{B}$ 同样输出 b'。若 $b'=b$, 模拟器输出 0, 表示猜测 $T = e(g,g)^{a^{q+1}s}$; 否则输出 1, 表示猜测 T 为群 G_T 中的随机元素 R。根据安全模型中 $\mathcal{A}$ 的优势定义 $\left|\Pr\left[b' = b\right] - 1/2\right|$, 因此有:

$$\begin{aligned} Adv_A &= \left|\Pr\left[b' = b\right] - 1/2\right| \\ &= \left|\Pr\left[\mathcal{B}\left(D, T = e(g,g)^{a^{q+1}s} = 0\right) - \frac{1}{2}\right]\right| \end{aligned} \tag{8}$$

当 T 为群 G_T 中的随机元素 R 时, m_b 对敌手是完全随机的, 可得出 $\Pr\left[\mathcal{B}\left(D, T = R\right) = 0\right] = \frac{1}{2}$, 因此模拟器的优势定义为

$$\begin{aligned} Adv_B &= \frac{1}{2}\Pr\left[\mathcal{B}\left(D, T = e(g,g)^{a^{q+1}s} = 0\right)\right] \\ &\quad + \frac{1}{2}\Pr\left[B\left(D, T = R\right) = 0\right] - \frac{1}{2} \\ &= \frac{1}{2}\left(\frac{1}{2} + Adv_A\right) + \frac{1}{4} - \frac{1}{2} \\ &= \frac{\varepsilon}{2} \end{aligned} \tag{9}$$

因此, 若 $\mathcal{A}$ 能以不可忽略优势攻破本文方案, 则 $\mathcal{B}$ 也能以不可忽略的优势攻破 RW 方案, 即可构建 $\mathcal{B}$ 能以不可忽略的优势解决 q-DPBDHE 问题。证毕。

定理 2. 系统隐私安全性证明

该系统为每个属性授权机构 AA 合理地分配该属性授权机构所管理的属性域, 每个 AA 均独立运行, 无需与其他的 AA 交互。不同的 AA 管理不同的属性域, 用户在不同机构为自己的不同属性去申请属性私钥, 不同授权机构只有用户的部分零碎属性, 对用户的全部属性不得而知, 因此, 能够在一定程度上保护用户个人隐私安全。其次, 多授权机构把单授权机构的信任和工作量进行了分散承担, 有效减轻了传统的单属性授权机构的计算负载和安全瓶颈, 可防止单属性授权机构因遭受恶意攻击造成密钥泄露, 导致云端数据被非法解密的风险。

定理 3. 前向安全和后向安全证明

前向安全和后向安全是 CP-ABE 属性撤销方案, 最基本的安全需求。前向安全是指属性被撤销的用户不能访问基于该属性加密的密文。后向安全是指新准入的用户在满足访问策略的情况下依然能够正确解密先前发布的密文。

1) 用户撤销时, AA 只需更新被撤销用户的版本密钥 $VK_{uid,aid}$, 用户未被撤销时 $ParCT$ 和 $VK_{uid,aid}$ 处于同周期, 可执行用户验证算法 $UserVerandDec$ 获得验证密文 CT' 进行进一步解密操作, 获取明文。当发生用户撤销时, AA 更新后的 $VK_{uid,aid}$ 与 $ParCT$ 不在同一周期, 无法完成用户验证。由于用户撤销不涉

及其他用户, 可以保证前后向安全。

2)用户属性撤销时, AA 更新撤销属性 att_x 的属性版本 $VK_{att_x}=v'_{att_x}$, 为包含该属性但未撤销的用户生成升级密钥 UUK_{aid}, UUK_{aid} 中包含用户私钥 SK_{uid} 除属性授权机构外其他任何用户均无法获得, 因此撤销该属性的用户不能使用其他用户的 UUK_{aid} 更新自己的属性私钥组件 $TK_{uid,aid}$。CCS 为涉及 att_x 的密文生成密文更新密钥 CUK_{aid} 进行密文组件进行同步更新。

若被撤销属性 att_x 的用户仍输入未更新的属性私钥组件 $TK_{uid,aid}$ 运行 Pr*oxyDecrpty* 算法, 由于 $TK_{uid,aid}$ 中的属性版本和密文组件 $C_{5,x}$ 中属性的版本号不同无法得到 $parCT$, 满足方案的前向安全。当新用户加入系统时, 其属性私钥 $SK_{uid,aid}$ 和用户版本密钥 $VK_{uid,aid}$ 均基于最新属性版本生成。假设其属性集合满足密文的访问策略便可解密。因此, 保证了本文方案的后向安全。

定理 4. 抵抗合法用户之间的合谋攻击证明

AA 选择随机数 $k_{uid,aid}$, $t_{uid,aid}\in Z_p^*$ 运行 *KeyGen* 算法为每个用户颁发属性私钥 $SK_{uid,aid}$, 共谋用户被随机变量所蒙蔽, 身份标识 uid 不同的用户无法选择性的替换身份标识 aid 不同 AA 为其颁发的 $SK_{uid,aid}$。当多个不具备解密权限的用户通过组合 $SK_{uid,aid}$ 的方式发动合谋攻击时, 即使属性集合满足访问策略, PDS 只对认证身份标识 uid 的用户的 $TK_{uid,aid}$ 执行 ProxyDec 算法, 因此, 合谋攻击密失败。

定理 5. 抵抗非法用户和代理解密服务器之间和合谋攻击证明

PDS 持有的属性私钥组件 $TK_{uid,aid}$ 为半解密密钥, 执行 ProxyDec 算法生成 $parCT$ 仍以密文的形式存储。合法的 ProxyDec 算法执行过程:

$$ParCT=\prod_{x\in s}\left(\begin{matrix}C_{1,x}\cdot e\left(L_{uid,aid},C_{2,x}\right)\cdot e\left(F\left(uid\right),C_{3,x}\right)\\ \cdot e\left(TK_{uid,aid},C_{4,x}\cdot C_{5,x}\right)\end{matrix}\right)^{c_x}$$

$$=\prod_{x\in s}\left(e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}\lambda_x/SK_{uid}}\cdot e\left(F\left(uid\right),g\right)^{\omega_x/SK_{uid}}\right)^{c_x}$$

$$=e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}s/SK_{uid}} \tag{10}$$

当用户属性撤销时, AA 为涉及撤销属性 att_x 的密文颁发更新密钥 CUK_{aid}, CCS 同步更新相关密文, 但被属性撤销的用户失去对该属性关联的属性私钥组件 $TK_{uid,aid}$ 的更新权限。当属性撤销的用户与 PDS 之间和合谋攻击时, ProxyDec 算法运行过程中

$$ParCT=\prod_{x\in s}\left(\begin{matrix}C_{1,x}\cdot e\left(L_{uid,aid},C_{2,x}\right)\cdot e\left(F\left(uid\right),C_{3,x}\right)\\ \cdot e\left(TK_{uid,aid},C_{4,x}\cdot C_{5,x}\right)\end{matrix}\right)^{c_x}$$

$$=\prod_{x\in s}\left(\begin{matrix}e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}\lambda_x/SK_{uid}}\cdot e\left(F\left(uid\right),g\right)^{\omega_x/SK_{uid}}\\ \cdot e\left(g,g\right)^{\left(t_{uid,aid}k_{uid,aid}\beta_{\delta(x)}/SK_{uid}\right)\cdot\left(v'_{att_x}r_x-v_{att_x}r_x\right)}\end{matrix}\right)^{c_x} \tag{11}$$

由于被撤销属性的属性版本 $VK_{att_x}=v_{att_x}$ 和密文中涉及被撤销属性的 $VK_{att_x}=v'_{att_x}$ 不同, 无法消去 $e\left(g,g\right)^{\left(t_{uid,aid}k_{uid,aid}\beta_{\delta(x)}/SK_{uid}\right)\cdot\left(v'_{att_x}r_x-v_{att_x}r_x\right)}$ 完成代理解密。属性撤销用户和代理解密服务器之间和合谋攻击失败。

4.2 正确性分析

假设用户属性集合满足密文访问策略, 令 $I=\left\{i:\rho(i)\in S\right\}$, 则存在常量 $\left\{c_i\in Z_p\right\}_{i\in I}$ 满足 $\sum_{i\in I}\lambda_x c_x=s$ 和 $\sum_{i\in I}\omega_x c_x=0$, 可以递归地恢复出 s, 进一步计算解出明文, 具体推导过程如下:

$$\prod_{x\in s}\left(\begin{matrix}C_{1,x}\cdot e\left(K_{uid,aid},C_{2,x}\right)\cdot e\left(F\left(uid\right),C_{3,x}\right)\\ \cdot e\left(L_{uid,aid},C_{4,x}\cdot C_{5,x}\right)\end{matrix}\right)^{c_x}$$

$$=\prod_{x\in s}\left(\begin{matrix}e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}\lambda_x/SK_{uid}+\alpha_{\delta(x)}\gamma_x}\\ \cdot e\left(g^{\alpha_{\delta(x)}}\cdot\left(g^{v_x}H\left(x\right)\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}/SK_{uid}}\cdot F\left(uid\right)^{\beta_{\delta(x)}},g^{-r_x}\right)\\ \cdot e\left(F\left(uid\right),g^{\omega_x/SK_{uid}+\beta_{\delta(x)}r_x}\right)\\ \cdot e\left(g^{t_{uid,aid}k_{uid,aid}\beta_{\delta(x)}/SK_{uid}},H\left(\rho(x)\right)^{r_x}\right)\\ \cdot e\left(g^{t_{uid,aid}k_{uid,aid}\beta_{\delta(x)}/SK_{uid}},g^{v_xr_x}\right)\end{matrix}\right)^{c_x}$$

$$=\prod_{x\in s}\left(e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}\lambda_x/SK_{uid}}\cdot e\left(F\left(uid\right),g\right)^{\omega_x/SK_{uid}}\right)^{c_x}$$

$$=e\left(g,g\right)^{k_{uid,aid}t_{uid,aid}\beta_{\delta(x)}\lambda_x s/SK_{uid}} \tag{12}$$

$$e\left(C_{5,x},VK_{uid,aid}\right)/\ parCT$$

$$=e\left(g,g\right)^{s/SK_{uid}} \tag{13}$$

$$\text{用户解密 } m=\frac{C_0}{CT'^{SK_{uid}}}=\frac{m\cdot e\left(g,g\right)^s}{\left(e\left(g,g\right)^{s/SK_{uid}}\right)^{SK_{uid}}} \tag{14}$$

5 性能分析及实验验证

本节在功能特性、存储开销、通信开销和计算开销方面将本文方案与已有几种撤销方案进行分析对比。为了描述方便, 描述符定义如下: $|G|$ 表示 G 中数据元素的长度; $|G_T|$ 表示 G_T 中数据元素的长度; $|p|$ 表示 Z_P 中数据元素的长度; n_a 表示系统中属性的总个数; n_u 表示系统中用户的总个数; n_c 表示与密文相关的属性个数; n_k 表示私钥中的属性个数; n_A 表示系统中 AA 的总个数; n_r 表示系统中撤销属性的个数; n_i 表示由 AA_{aid} 管理的属性个数; E 和 E_T 示 G 和 G_T 中模指数运算, P 表示双线性对运算。乘法运算开销远小于幂指数运算和双线性对运算, 因此, 忽略乘法运算开销。

5.1 功能特征

从表 1 中可以看出, 文献[19]仅支持用户撤销。文献[20-21, 24]仅支持用户属性撤销。文献[16]和本文方案既支持用户属性撤销又支持用户撤销, 但是文献[16]不支持大属性域。

5.2 存储开销

表 2 将本文方案与其他相关文献的存储开销进行对比。属性授权机构 AA 的存储开销主要来自于主密钥, 所有方案主密钥开销都较少。数据属主 DO 的存储开销主要来自于公钥。文献[16]在初始化过程中, AA 为系统属性集中每个属性设置公开属性密钥, 公钥随属性总数 n_a 呈线性增长。文献[20]中的公钥随 n_u 呈斜率为常数的线性增长, 文献[19]和文献[24]中每个 AA 可管理多个属性, 公钥长度只随系统中 AA 的总个数 n_A 呈线性增长, 文献[21]公钥为常数, 本文方案中公钥随属性总数 n_a 与 AA 总数 n_A 呈斜率为常数的线性增长。云存储服务商 CCS 的存储开销主要来自于密文与密文头。表 2 中所有方案的密文储开销都随与密文相关的属性个数 n_c 呈线性增长, 因此, CCS 端的存储开销相近。数据用户 DU 的存储开销主要来自于用户私钥。文献[16, 19]和本文方案的用户私钥长度仅为 $|p|$。

5.3 通信开销

表 3 将本文方案与其他相关文献的通信开销进行对比。通信开销主要是由密钥和密文产生的。属性授权机构 AA 与数据用户 DU 之间的通信开销主要是由密钥产生的。属性授权机构 AA 与数据属主 DO 之间的通信开销主要由公钥产生。文献[21]和本文方案在撤销属性时, 需要给 DO 颁发撤销属性相应的更新属性公钥而产生一个 $n_r|g|$ 的通信开销。云服存储务器 CCS 与数据用户 DU 之间通信成本主要是由密

表 1 性能特征对比

Table 1 Comparison of performance characteristics

对比方案	授权机构	大属性域	安全假设	安全模型	撤销粒度
文献[16]	单授权机构	否	q - parallel BDHE	标准模型	用户属性/用户撤销
文献[19]	多授权机构	是	q - DPBDHE	随机模型	用户撤销
文献[20]	单授权机构	是	q - type	标准模型	用户属性撤销
文献[21]	单授权机构	是	q - parallel BDHE	标准模型	用户属性撤销
文献[24]	多授权机构	是	q - type	随机模型	用户属性撤销
本文方案	多授权机构	是	q - DPBDHE	随机模型	属性/用户撤销

表 2 存储开销对对比

Table 2 Comparison of storage cost

对比方案	AA	DO	CCS	DO
文献[16]	$\|g\|+\|p\|$	$(2+n_a)\|g\|+\|g_T\|$	$(2n_c+1)\|g\|+\|g_T\|$	$\|p\|$
文献[19]	$2\|p\|$	$n_A(\|g\|+\|g_T\|)$	$3n_c\|g\|+(n_c+1)\|g_T\|$	$\|p\|$
文献[20]	$3\|p\|$	$(2n_u+6)\|g\|+\|g_T\|$	$(1+3n_c)\|g\|+\|g_T\|$	$(2n_c+1)\|g\|+\|g_T\|$
文献[21]	$2\|p\|+\|g\|$	$2\|g\|+\|g_T\|$	$2(n_c+1)\|g\|+\|g_T\|$	$(3+n_k)\|g\|$
文献[24]	$2\|p\|$	$n_A(\|g\|+\|g_T\|)$	$3n_c\|g\|+(n_c+1)\|g_T\|$	$2n_k\|g\|+n_A\log(N+1)\|p\|$
本文方案	$2\|p\|$	$(2n_A+n_a)\|g\|+n_A\|g_T\|$	$(n_c+1)\|g_T\|+4\|g\|$	$\|p\|$

表 3 通信开销对比

Table 3 Comparison of communication overhead

对比方案	AA & DU	AA & DO	CCS & DU	CSS & DO
文献[16]	$(n_k+1)\|g\|+p$	$(2+n_a+n_r)\|g\|+\|g_T\|$	$(2n_c+1)\|g\|+(n_c+1)\|g_T\|$	$(2n_c+1)\|g\|+\|g_T\|$
文献[19]	$2n_i\|g\|+\|p\|$	$n_A(\|g\|+\|g_T\|)$	$3n_c\|g\|+3(n_c+1)\|g_T\|$	$(3n_c+1)\|g_T\|+3\|g\|$
文献[20]	$(2n_k+3)\|g\|+\|p\|$	$(2n_u+6)\|g\|+\|g_T\|$	$(1+3n_c)\|g\|+(4n_r+1)\|g_T\|$	$(3n_c+1)\|g\|+\|g_T\|$
文献[21]	$(4+n_k)\|g\|$	$2\|g\|+\|g_T\|$	$2(n_c+1)\|g\|+\|g_T\|$	$\|g\|+\|g_T\|$
文献[24]	$2n_k\|g\|+n_A\log(N+1)\|p\|$	$n_A(\|g\|+\|g_T\|)$	$(3n_c+1)\|g\|+(n_c+1)\|g_T\|$	$3n_c\|g\|+(n_c+1)\|g_T\|$
本文方案	$\|p\|+2n_k\|g\|$	$(2n_A+n_a+n_r)\|g\|+n_A\|g_T\|$	$(2n_c+1)\|g_T\|+4\|g\|$	$(n_c+1)\|g_T\|+4\|g\|$

文产生的。云存储服务器 CCS 与数据属主 DO 之间的通信主要是由 DO 生成的密文产生的。

5.4 计算开销

表 4 将本文方案与其他相关文献的计算开销进行对比。文献[21]中的方案的加密计算开销最少, 但是文献[21]方案基于单属性授权机构, 其安全性高度依赖于属性授权中心的可靠性, 在属性单授权机构不可信或遭受恶意攻击的情况下, 可能造成密钥泄露而导致数据被非法解密。此外, 在用户属性撤销时需同时通过代理重加密技术更新密文, 且没有考虑用户撤销的情形。本文方案在加密计算开销上略大于文献[16, 19, 21]。与文献[20, 24]相比, 本文的方案在用户解密的计算代价上存在显著优势, 相比于其他文献方案, 本方案在综合计算开销上具有较大优势。

5.5 实验分析

实验环境配置: 64 bit Ubuntu 20.04 操作系统、Intel® CoreTM i5-7400U(3.0GHz)、内存 8GB、基于 Standford Pairing-Based Crypto library[25]中的 Charm[26]进行实验仿真, 程序在基于 512 bit 有限域

表 4 计算开销对比

Table 4 Comparison of Computational Cost

对比方案	Encryption	Decryption		User Attribute Revocation	User Revocation
		CCS-Decrypt	DU-Decrypt		
文献[16]	$(4n_c+1)E_G+E_{G_T}$	$2n_cP$	E_{G_T}	$2n_rE_G$	$2E_G$
文献[19]	$(2n_c+1)E_{G_T}+4n_cE_G$	$2n_cE_{G_T}+3n_cP$	E_G	—	0
文献[20]	$(2+4n_c+3n_r)E_G+E_{G_T}$	$(3n_r+2)p+n_rE_G$	$(3n_r+1)E_{G_T}$	$2n_rE_{G_T}+(n_r+1)E_G$	—
文献[21]	$(2n_c+2)E_G+E_{G_T}$	$E_{G_T}+(n_k+3)E_G$	$E_{G_T}+P$	$(5n_r+4)E_G$	—
文献[24]	$5n_cE_G+(2n_c+1)E_{G_T}$	$(2n_c+n_k+2)E_G+(2n_c+1)E_{G_T}+3n_{ep}$	$n_kE_G+n_cp+n_cE_{G_T}$	$(6n_r+3n_c)E_G+E_{G_T}$	—
本文方案	$5n_cE_G+(2n_c+1)E_{G_T}$	$3n_kP+n_kE_{G_T}$	E_{G_T}	$2n_rE_G$	$2E_G$

上的超奇异曲线 $y^2=x^3+x$ 中的 160bit 椭圆曲线群上运行。所有实验数据均为 20 次试验的平均值。具体实验结果如图 2、图 3 和图 4 所示。

如图 2 所示, 数据加密过程中加密时间与访问策略属性数量成线性增长关系。文献[20]需要通过代理重加密为每个属性额外计算 2 个用于撤销后更新密文的组件, 而其他方案撤销计算由第三方执行, 所以文献[20]所需执行时间长一些。本文方案与文献[16]加密时间相近。

如图 3 所示, 解密时间与解密所需属性数量呈线性增长关系, 文献[16, 20]和本文方案解密过程均由代理解密服务器进行部分解密和用户最终解密两部分构成, 时间开销都比较小。文献[18]代理解密服

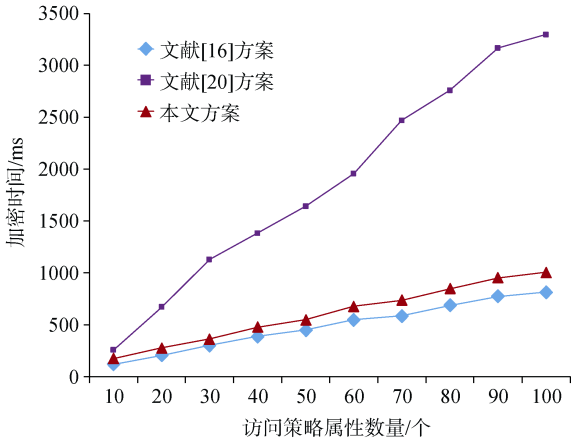

图 2 加密时间对比

Figure 2 Comparison of encryption time

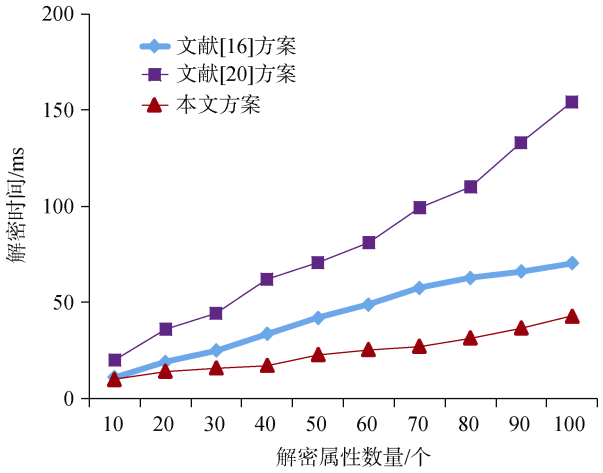

图 3　解密时间对比

Figure 3　Comparison of decryption time

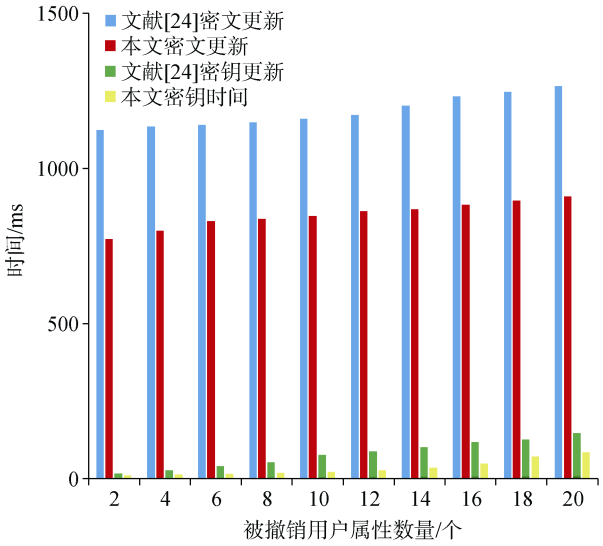

图 4　用户属性撤销对比

Figure 4　Comparison of user attribute revocation

务器需要计算重加密密文，增加了解密时间开销。文献[16]和本文方案在用户解密阶段只需计算一个 G_T 中的指数运算，即可完成解密操作，且与属性数量无关。

图 4 描述了用户属性撤销过程中随着撤销属性数量的增加，更新密文和密钥时间开销的变化况。在本实验中，系统中 AA 数量 n_A 设置为 5，每个 AA 管理的属性个数 n_i 设置为 10。当用户属性被撤销时，文献[24]方案和本文方案只更新涉及到被撤销属性的密钥和密文，由于密钥和密文的更新复杂的计算由 CSP 完成，AA 只需要进行少量计算就能够完成密钥更新，因此所需计算时间均较少。但是文献[24]方案引入 KEK 树密钥，比本文方案的版本号密钥需要的计算量多。在密文更新方面，文献[24]方案需要 CSP 通过代理重加密完成密文更新，比本文仅需通过更新属性版本密钥完成密文更新所需时间长。

6　结论

针对云环境中一些用户的相关属性变更引起的用户权限动态变更问题，本文中提出一种大属性域版本控制的云安全用户属性动态撤销策略，该策略通过更新属性版本密钥实现用户属性撤销，更新用户版本密钥实现用户撤销。为提高系统效率，将复杂的解密计算外包给云服务商。理论分析与实验验证表明，本文方案实现了动态的用户属性撤销与用户撤销，且可以有效抵制合谋攻击，增强了方案的安全性。在功能和计算效率方面具有一定优势。此外，该方案支持大属性域，具有良好的扩展性，可更加灵活的应用于实际环境

参考文献

[1] Mell P. The NIST Definition of Cloud Computing[J]. *Communications of the ACM*, 2010, 53(6): 50.

[2] Wang Y D, Yang J H, Xu C, et al. Survey on Access Control Technologies for Cloud Computing[J]. *Journal of Software*, 2015, 26(5): 1129-1150.
(王于丁，杨家海，徐聪，等. 云计算访问控制技术研究综述[J]. *软件学报*, 2015, 26(5): 1129-1150.)

[3] Sahai A, Waters B. Fuzzy Identity-Based Encryption[C]. *The 24th annual international conference on Theory and Applications of Cryptographic Techniques*, 2005: 457-473.

[4] Waters B. Ciphertext-Policy Attribute-Based Encryption: An Expressive, Efficient, and Provably Secure Realization[M]. Public Key Cryptography - PKC 2011. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 53-70.

[5] Goyal V, Pandey O, Sahai A, et al. Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data[C]. *The 13th ACM conference on Computer and communications security*, 2006: 89-98.

[6] Sookhak M, Yu F R, Khan M K, et al. Attribute-Based Data Access Control in Mobile Cloud Computing: Taxonomy and Open Issues[J]. *Future Generation Computer Systems*, 2017, 72: 273-287.

[7] Liu Z, Cao Z F, Wong D S. White-Box Traceable Ciphertext-Policy Attribute-Based Encryption Supporting any Monotone Access Structures[J]. *IEEE Transactions on Information Forensics and Security*, 2013, 8(1): 76-88.

[8] Lewko A, Waters B. Unbounded HIBE and Attribute-Based Encryption[M]. Advances in Cryptology - EUROCRYPT 2011. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 547-567.

[9] Rouselakis Y, Waters B. Efficient Statically-Secure Large-Universe Multi-Authority Attribute-Based Encryption[M]. Financial Cryptography and Data Security. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015: 315-332.

[10] Zhang Y H, Zheng D, Deng R H. Security and Privacy in Smart

Health: Efficient Policy-Hiding Attribute-Based Access Control[J]. *IEEE Internet of Things Journal*, 2018, 5(3): 2130-2145.

[11] Pirretti M, Traynor P, McDaniel P, et al. Secure Attribute-Based Systems[C]. *The 13th ACM conference on Computer and communications security*, 2006: 99-112.

[12] Bethencourt J, Sahai A, Waters B, et al. Ciphertext-policy attribute-based encryption[C]. *2007 IEEE Symposium on Security and Privacy*, 2007: 321-334.

[13] Rafaeli S, Hutchison D. A Survey of Key Management for Secure Group Communication[J]. *ACM Computing Surveys*, 2003, 35(3): 309-329.

[14] Wu X L, Jiang R, Bhargava B. On the Security of Data Access Control for Multiauthority Cloud Storage Systems[J]. *IEEE Transactions on Services Computing*, 2017, 10(2): 258-272.

[15] Zhao Z Y, Zhu Z Q, Wang J H, et al. Revocable Attribute-Based Encryption with Escrow-Free in Cloud Storage[J]. *Journal of Electronics & Information Technology*, 2018, 40(1): 1-10.
(赵志远, 朱智强, 王建华, 等. 云存储环境下无密钥托管可撤销属性基加密方案研究[J]. *电子与信息学报*, 2018, 40(1): 1-10.)

[16] Yan X C, Chen Y, Ba Y, et al. Updatable Attribute-Based Encryption Scheme Supporting Dynamic Change of User Rights[J]. *Journal of Computer Research and Development*, 2020, 57(5): 1057-1069.
(严新成, 陈越, 巴阳, 等. 支持用户权限动态变更的可更新属性加密方案[J]. *计算机研究与发展*, 2020, 57(5): 1057-1069.)

[17] Miklau G, Suciu D. Controlling Access to Published Data Using Cryptography[M]. Proceedings 2003 VLDB Conference. Amsterdam: Elsevier, 2003: 898-909.

[18] Bagga W, Molva R, Crosta S. Policy-Based Encryption Schemes from Bilinear Pairings[C]. *The 2006 ACM Symposium on Information, computer and communications security*, 2006: 368.

[19] Zhang K, Ma J F, Li H, et al. Multi-Authority Attribute-Based Encryption with Efficient Revocation[J]. *Journal on Communications*, 2017, 38(3): 83-91.
(张凯, 马建峰, 李辉, 等. 支持高效撤销的多机构属性加密方案[J]. *通信学报*, 2017, 38(3): 83-91.)

[20] Lian H J, Wang Q X, Wang G B. Large Universe Ciphertext-Policy Attribute-Based Encryption with Attribute Level User Revocation in Cloud Storage[J]. *The International Arab Journal of Information Technology*, 2020, 17(1): 107-117.

[21] Zhao J, Zeng P, Choo K K R. An Efficient Access Control Scheme with Outsourcing and Attribute Revocation for Fog-Enabled E-Health[J]. *IEEE Access*, 9: 13789-13799.

[22] Hur J, Noh D K. Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems[J]. *IEEE Transactions on Parallel and Distributed Systems*, 2011, 22(7): 1214-1221.

[23] Li J G, Yao W, Han J G, et al. User Collusion Avoidance CP-ABE with Efficient Attribute Revocation for Cloud Storage[J]. *IEEE Systems Journal*, 2018, 12(2): 1767-1777.

[24] Liu Z C, Jiang Z L, Wang X, et al. Practical Attribute-Based Encryption: Outsourcing Decryption, Attribute Revocation and Policy Updating[J]. *Journal of Network and Computer Applications*, 2018, 108: 112-123.

[25] Akinyele J A, Garman C, Miers I, et al. Charm: A Framework for Rapidly Prototyping Cryptosystems[J]. *Journal of Cryptographic Engineering*, 2013, 3(2): 111-128.

[26] Lynn, B. *http://crypto.stanford.edu/pbc/The Pairing-based Crypto Library.May* 2016

党鲜玲 于 2016 年在太原理工大学现代科技学院应用化学专业获得学士学位。太原科技大学计算机技术专业攻读硕士学位。研究领域为云安全与系统安全。研究兴趣包括: 人工智能、系统安全与隐私保护、网络安全。Email: 2496994304@qq.com

郭银章 于 2011 年在太原科技大学机械设计及理论专业获得博士学位。现任太原科技大学物联网与云计算实验室主任, CCF 协同计算专委会常委, CCF 教育专委会常委, CCF 太原分部主席, 研究领域为协同计算与云计算、系统安全与隐私保护。Email: guoyinzhang@ 263.net