一种基于异或运算的属性撤销 CP-ABE 方案

邱建兵，胡　勇
（四川大学网络空间安全学院，成都 610065）

摘　要：针对属性撤销 CP-ABE 方案中密钥更新时属性授权机构与用户之间的通信开销过大及密文更新时云存储中心的计算复杂度过高的问题，本文提出一种基于异或运算的、支持属性级撤销的密文策略属性基加密方案. 在该方案中，属性授权机构先将需要撤销的属性名称、被撤销用户的标识及新的时间参数发送给云存储中心，然后云存储中心根据用户标识和新的时间参数的异或结果与密文的一部分进行异或运算，得到新密文. 收到新密文后，正常用户可以利用自己的密钥解密得到原密文，进而得到明文，而被撤销用户则只能使用已撤销属性的新密钥才能解密得到原密文，从而实现属性级撤销. 理论分析和数值模拟表明，在保证系统安全性的前提下，该方案能够减少属性授权机构与用户间的通信开销，降低云存储中心的计算复杂度.

关键词：访问控制；密文策略属性基加密；异或运算；属性级撤销
中图分类号：TP391.1　　**文献标志码**：A　　DOI：10.19907/j.0490-6756.2024.013001

A CP-ABE scheme for attribute revocation based on XOR operation

QIU Jian-Bing, *HU Yong*
(School of Cyber Science and Engineering, Sichuan University, Chengdu 610065, China)

Abstract: Aiming at the problems of high communication overhead between attribute authorization authority and normal users when the key is updated, high computational complexity in cloud center when the ciphertext is updated, an attribute-based ciphertext policy encryption scheme based on XOR operation is proposed to support attribute level revocation. Attribute authorization first sends the attribute name and the user ID to be revoked and the new time parameter to the cloud center. Then the cloud center uses the XOR result of the user ID and the new time parameter to perform the XOR operation with part of the ciphertext to obtain the new ciphertext. The normal user can decrypt the original ciphertext by using his own key, and further obtain the plaintext. The revoked user can decrypt the original ciphertext only by using the new key of the revoked attribute, thereby realizing attribute level revocation. The analysis shows that under the premise of ensuring system security, this scheme reduces the communication overhead between attribute authorization and users, and reduces the computing complexity in cloud center.

Keywords: Access control; CP-ABE; XOR operation; Attribute level revokation

1　引　言

数据的云共享是共享经济的关键[1]，当前云存储在用户数据托管方面得到了广泛的应用. 但由于数据存放在用户无法控制的云端，这可能带来严重的数据隐私安全问题[2,3]. 属性基加密方法

收稿日期：2023-02-12
作者简介：邱建兵(1996－)，男，江西鹰潭人，硕士研究生，主要研究方向为数据安全. E-mail：1711130224@qq.com
通讯作者：胡勇. E-mail：huyong@scu.edu.cn

(ABE)[4]能够实现细粒度的访问控制. 其中,密文策略属性基加密(CP-ABE)方法的出现使得用户能更加灵活地设置访问策略,控制其他用户对存储于远端服务器的数据的访问[5]. 然而,在采用 CP-ABE 进行访问控制的多用户应用系统中,通常存在用户身份变化的情况. 此时为确保系统安全及用户正确解密,就需要撤销并更新相关属性的私钥组件. 按其影响范围,这些撤销可分为用户部分属性撤销、用户撤销以及系统属性撤销[6]. 其中,用户部分属性撤销即改变特定用户的特定属性值而不影响其他用户和该用户的其他属性值,也称为属性级用户撤销,是最细粒度的撤销方式.

2011 年,Hur 等[7]提出了一个利用密钥加密密钥树实现属性级用户撤销的 CP-ABE 方案. 在该方案中,若用户的某个属性被撤销,则云存储中心将生成新的密钥加密密钥,并负责重新加密密文. 由于该方案中的属性群密钥对该群的用户完全通用,因而不能抵抗撤销用户和未撤销用户的合谋攻击. 为解决这个问题,近年来学术界提出了一些方法. 例如,2013 年 Yang 等[8]提出了一种面向云存储的细粒度访问控制方案,该方案为每个属性生成两个公开参数,当进行属性撤销时由属性授权机构负责更新属性对应的公开参数,并为用户更新解密密钥,而这也增加了属性授权机构的计算开销以及其与用户之间的通信开销. 2017 年,闫玺玺等[9]提出了具有隐私保护的属性撤销方案,该方案通过设置属性陷门更新用户的属性私钥,并以令牌树机制控制数据共享的范围,该方案不能抵抗撤销用户与未撤销用户的合谋攻击. 2018 年,Xue 等[10]提出了一种基于非单调访问结构的属性撤销方案,该方案基于合数阶群构建,效率较低. 2022 年,董国芳等[11]提出了支持撤销属性的 CP-ABE 密钥更新方法,该方法有效减少了属性授权中心在更新密钥时的计算开销,但该方案由属性授权中心创建并维护用户撤销列表及属性密钥撤销列表,增加了属性授权中心的存储开销.

鉴于现有的多数方案都主要针对性能或安全性等单个问题进行研究,而在实现高性能的同时保证强的安全性却是属性级用户撤销的研究趋势,本文提出一个性能和安全性兼顾的方案. 该方案使用异或运算对密文进行更新,减少云存储中心的计算开销,同时通过只更新被撤销用户的密钥的方式达到减少属性授权机构与数据用户之间的通信开销的目的,并通过两个定理证明了该方案的安全性.

2　算法定义及安全模型

本文的方案(后文简称方案)包括四类实体:数据拥有者(Data Owner, DO),云存储中心(Cloud Center, CC),属性授权机构(Attribute Authority, AA)和数据用户(Data User, DU),各实体间的联系如图 1 所示.

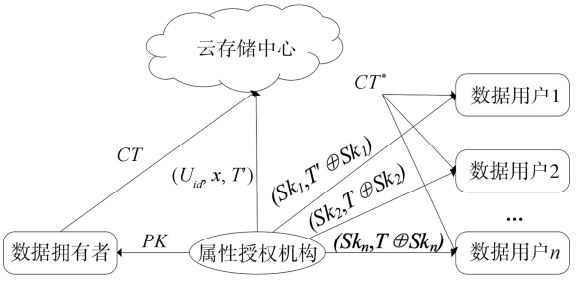


图 1　系统模型

Fig. 1　System model

2.1　算法定义

方案主要由以下几个算法构成.

(1) *Setup*():AA 运行该算法对系统进行初始化,输出系统公钥 PK 和系统主密钥 MK.

(2) *KeyGen*():AA 运行该算法以系统主密钥 MK 和属性集合 S 作为输入,输出用户私钥 Sk.

(3) *Encrypt*():DO 运行该算法,基于访问树结构 T 对数据明文 M 进行加密,并将加密得到的密文 CT 发送给 CC.

(4) *CTUpdate*():CC 收到 AA 的新的时间参数、要撤销的用户标识以及属性标识后,运行该算法进行与属性标识有关的密文更新.

(5) Decrypt():DU 获得 CC 发送的密文后,首先解密该密文得到原始密文 CT,接着使用自己的私钥解密 CT,当其拥有的属性集合满足对应的访问策略时便可解密得到明文 M.

2.2　安全模型

方案将安全模型定义为挑战者 S 与攻击者 A 间的安全游戏[12],具体规则如下.

参数设置阶段:S 运行系统初始化算法 *Setup*,生成系统主密钥 MK 和系统公开密钥 PK,并将系统公开密钥 PK 发送给 A.

密钥查询阶段 1:A 适应性地提交一系列属性集合$(S_1, S_2, \cdots, S_{q_1})$给 S,S 运行密钥生成算法 *KeyGen* 生成对应的私钥 SK,并将其发送给 A.

挑战阶段:A 提交长度相等的两个消息M_0、

M_1 和一个访问控制策略 (A^*,ρ) 给 S，S 公平地选择 $b\in\{0,1\}$，并基于访问控制策略对消息 M_b 进行加密，将生成的密文 CT^* 发送给 A.

密钥查询阶段 2：与密钥查询阶段 1 类似，A 继续适应性地提交一系列属性集合，但需满足如下限制：(1) 提交的属性集合都不能满足访问策略；(2) 更新后的解密密钥都不能解密挑战密文.

猜测阶段：A 输出值 $b'\in\{0,1\}$ 作为对 b 的猜测，如果 $b=b'$，则认为 A 赢得了该游戏，定义 A 在该游戏中的优势为 $\varepsilon=|Pr[b=b']-1/2|$. 对于一个属性撤销 CP-ABE 方案，如果在任意多项式时间内 A 的优势 ε 是可忽略的，即 A 几乎不可能赢得游戏的胜利，则称该方案是抗选择明文攻击 (IND-CPA) 安全的.

3　方案描述

方案主要包括五个算法，具体构造如下.

(1) 系统初始化算法 *Setup*.

令 G_0 为一个阶为素数 p 的双线性群，g 为 G_0 的生成元. 此外，令 $e:G_0\times G_0\to G_1$ 表示双线性映射. 群的大小由安全参数 λ 决定. 对于 $i\in Z_p$ 和 Z_p 中元素组成的集合 S，定义拉格朗日系数 $\Delta_{i,S}$ 为

$$\Delta_{i,S}(x)=\prod_{j\in S,j\neq i}\frac{x-j}{i-j} \tag{1}$$

此外，应用哈希函数 $H:\{0,1\}^*\to G_0$，并将其作为一个随机预言机. 此函数将以字符串描述的属性值映射为一个随机群元素.

属性授权机构运行 *Setup* 算法对系统进行初始化. 选择一个阶为素数 p 的双线性群 G_0，该群的生成元为 g. 然后随机选择加密指数 $\alpha,\beta\in Z_p$，输出系统公钥为

$$PK=(G_0,g,h=g^{\beta},f=g^{1/\beta},e(g,g)^{\alpha}) \tag{2}$$

系统主密钥 MK 为 (β,g^{α}).

(2) 密钥生成算法 *SkeyGen*.

属性授权机构维护初始的时间参数 T，运行密钥生成算法 *SkeyGen* 生成用户的解密密钥 Sk，将 Sk 和 $Sk\oplus T$ 发送给用户，并将该初始时间参数 T 发送给云存储中心.

该算法以系统主密钥 MK 和属性集合 S 作为输入，输出用户私钥 Sk. 该算法首先选择一个随机数 $r\in Z_p$，然后对每个属性 $j\in S$，随机选择 $r_j\in Z_p$，计算用户私钥：

$$Sk=(D=g^{(\alpha+\gamma)/\beta},\ \forall j\in S:D_j=g^{r}\cdot H(j)^{r_j},D'_j=g^{r_j}) \tag{3}$$

其中 H 为哈希函数.

(3) 数据加密算法 *Encrypt*.

数据拥有者在将数据 M 外包给云存储中心之前，需要运行数据加密算法 *Encrypt* 对数据 M 加密. 该算法基于访问树结构 T 对数据 M 进行加密. 该算法首先为树 T 中的每个节点 x (包括叶子) 选择一个多项式 q_x. 这些多项式采用自顶向下的方式从根节点 R 进行选择. 对于树中的每个节点 x，设多项式 q_x 的阶 d_x 为节点 x 的阈值 k_x 减 1，即 $d_x=k_x-1$.

本文从根节点 R 开始随机选择 $s\in Z_p$，并设置 $q_R(0)=s$. 然后，随机选择多项式 q_R 的其他 d_R 个节点来定义该多项式. 对于其他的任何节点 x，令 $q_x(0)=q_{\mathrm{parent}(x)}(\mathrm{index}(x))$ 且随机选择多项式 q_x 的其他 d_x 个点来定义该多项式.

令 Y 为树 T 的叶子节点的集合. 接着，通过访问树结构 T 构建如下的密文：

$$CT=(T,\widetilde{C}=Me(g,g)^{\alpha s},C=h^{s},\ \forall y\in Y:C_y=g^{q_y(0)},C'_y=H(att(y)^{q_y(0)})) \tag{4}$$

其中 h 的值为 g^{β}.

(4) 密文更新算法 *CTUpdate*.

一旦收到属性授权机构的新的时间参数 T'、要撤销的用户标识 U_{id} 以及属性标识 x，则运行密文更新算法 *CTUpdate* 进行与属性标识 x 有关的密文更新. 以密文 CT_1 为例，该算法选取密文 CT_1 最前面的一小部分定义为密文前部 CT_1_1，首先将要撤销的用户标识 U_{id} 与新的时间参数 T' 进行异或运算得到 $T'\oplus U_{\mathrm{id}}$，接着分别用初始的时间参数 T、$T'\oplus U_{\mathrm{id}}$ 与密文前部 CT_1_1 进行异或运算生成新的密文：

$$CT_1^*=\{T\oplus CT_1\_1,T'\oplus U_{\mathrm{id}}\oplus CT_1\_1,U_{\mathrm{id}}\}+CT'_1 \tag{5}$$

其中 CT'_1 为选取密文 CT_1 中除密文前部 CT_1_1 外剩下的部分；T 为属性授权机构维护的初始时间参数.

(5) 数据解密算法 *Decrypt*.

数据用户获得云存储中心发送的密文 $CT_1^*=\{T\oplus CT_1\_1,T'\oplus U_{id}\oplus CT_1\_1,U_{id}\}+CT'_1$ 后，首先需要解密该密文得到原始密文，具体步骤为：利用属性授权机构发送的解密密钥 Sk 和 $Sk\oplus T_{\mathrm{user}}$ 进行异或运算得到时间参数 T_{user}，接着判断当前的用户标识是否为要撤销的用户标识 U_{id}. 如果是，则计算 $(T_{\mathrm{user}}\oplus U_{\mathrm{id}})\oplus(T'\oplus U_{\mathrm{id}}\oplus CT_1\_1)+CT'_1$ 作

为原密文 CT,否则计算$T_{\text{user}}\oplus(T\oplus CT_1_1)+CT_1'$作为原密文 CT,流程如图 2 所示.

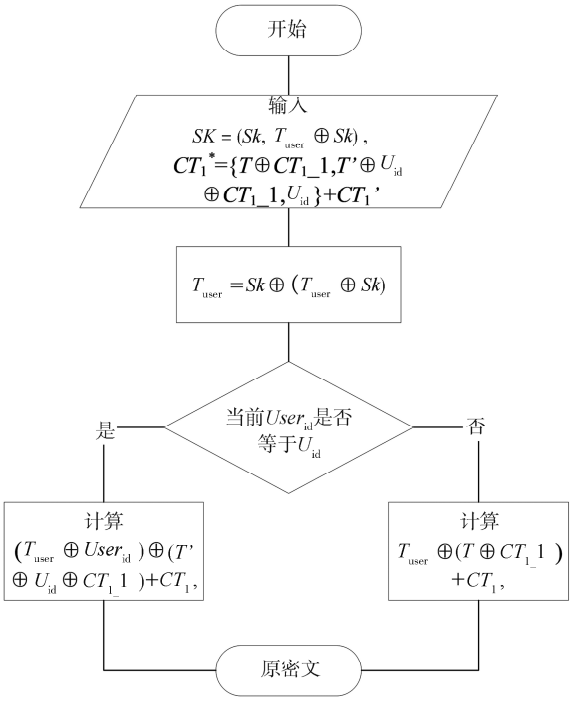


图 2　生成原密文的流程图

Fig. 2　Flow chart of generating original ciphertext

然后,定义递归算法 DecryptNode(CT, SK, x),该算法的输入为一个密文 $CT=(T,\widetilde{C},C,\forall y\in Y:C_y,C_y')$,一个基于属性集 S 的私钥 SK 和树 T 中的一个节点 x.

假如节点 x 为叶子节点,令 $i=att(x)$,假如 $i\in S$,定义

$$DecryptNode=\frac{e(D_i,C_x)}{e(D_i',C_x')}=\frac{e(g^r\cdot H(i)^{r_i},g^{q_x(0)})}{e(g^{r_i},H(i)^{q_x(0)})}=e(g,g)^{rq_x(0)}\tag{6}$$

其中D_i和D_i'是密钥组件的一部分;C_x和C_x'是密文的一部分,具体表达式在式(3)和式(4)中已给出.假如 $i\notin S$,定义 DecryptNode(CT, SK, x)$=\perp$.

现在考虑当 x 是非叶子节点的递归情况. 算法 DecryptNode(CT, SK, x)的计算情况如下:对于节点 x 的所有子节点 z,调用函数 DecryptNode(CT, SK, z)并且存储结果为F_z. 令S_x为一个任意的大小为k_x的子节点 z 的集合,且满足$F_z\neq\perp$. 如果不存在这样的集合,函数返回$\perp$. 否则计算$F_x=\prod_{z\in S_x}F_z^{\Delta_{i,S_x'}(0)}$,其中 $i=\text{index}(z)$,

$$S_x'=\{\text{index}(z):z\in S_x\}=\prod_{z\in S_x}(e(g,g)^{rq_z(0)})^{\Delta_{i,S_x'}(0)}=\prod_{z\in S_x}(e(g,g)^{rq_{\text{parent}(z)}(\text{index}(z))})^{\Delta_{i,S_x'}(0)}=\prod_{z\in S_x}(e(g,g)^{rq_x(i)})^{\Delta_{i,S_x'}(0)}=e(g,g)^{rq_x(0)}\tag{7}$$

并且返回上述计算结果.

定义函数 DecryptNode 后,解密算法通过调用树 T 的根节点 R 上的函数开始执行. 如属性集合 S 满足访问树 T,则令

$$A'=\text{DecryptNode}(CT,SK,R)=e(g,g)^{rq_R(0)}=e(g,g)^{rs}\tag{8}$$

则解密运算就是

$$\widetilde{C}/(e(C,D)/A')=\widetilde{C}/(e(h^s,g^{(\alpha+\gamma)/\beta})/e(g,g)^{rs})=M\tag{9}$$

M 即为数据拥有者外包的原始数据.

4　安全性证明

本方案的安全性证明主要基于第 2.2 节的安全模型及以下两个定理.

定理 4.1　若确定性的 q-parallel BDHE 假设成立,则不存在多项式的时间内攻击者能够选择性地攻破本方案,其中挑战矩阵规模为$l^*\times n^*$($n^*\leqslant q$)[13].

证明　假定 A 是一个在选择性安全挑战场景下拥有优势Adv_A的攻击者,他选择一个不超过 q 列的挑战矩阵A^*. 在没有一个解密密钥 Sk 能够成功解密挑战者密文的限制下,构建一个具有不可忽略优势的挑战者 S 来解决确定性的 q-parallel BDHE 问题.

定理 4.2　本方案的访问控制模式能够抵抗用户的非授权访问[14].

证明　非授权的用户访问主要包括两种情形.(1) 拥有不满足访问控制策略属性集合的用户尝试访问并解密数据;(2)当用户的某些属性被撤销后,其仍尝试用以前的解密密钥访问数据.

情形 1　拥有不满足访问控制策略属性集合的用户无法成功解密密文. 同时,还要考虑到多用户的合谋攻击. 在本方案中,所有用户都需先使用时间参数 t 和用户 id 与收到的密文进行异或运算得到原始密文后才能使用密钥进行解密,这就可以实现即使某些用户拥有相同的属性集合,他们仍无法进行合谋来解密密文.

情形 2　假设用户的某个属性被撤销,属性授权机构将生成一个新的时间参数 t 来更新被撤销

用户的密钥，并将新的时间参数 t、被撤销用户 id 发送给云存储中心，用于更新与被撤销属性相关的密文. 经过这样处理后，被撤销用户的旧密钥由于时间参数 t 不同而不能解密得到原密文，从而不能解密得到明文. 另一方面，使用新密钥能得到原密文，但由于密钥组件不匹配，同样无法解密原密文得到明文，即被撤销用户无法使用以前的密钥解密密文.

5　性能分析

通过与 Hur[7] 方案和 Yang[8] 方案的比较，我们分析了本方案的存储开销、通信开销和计算效率. 令 $|p|$ 为 Z_p 中元素的长度. 设 $|g|$ 和 $|g_T|$ 分别为 G 和 G_T 中的元素长度，n_a 和 n_u 分别表示系统中属性和用户的总数，n_r 表示要更新密钥的用户的数量. 设 $n_{a,i}$ 表示用户 i 拥有的属性数量，l 表示与密文相关的属性的数量，r 表示被撤销的属性数量，$|T|$ 表示时间参数的长度，$|u_{\mathrm{id}}|$ 表示用户标识 U_{id} 的长度.

5.1　存储开销

表 1 显示了系统中各实体的存储开销的比较. 属性授权机构的主要存储开销来自 Hur 方案中的主密钥. 除主密钥外，在 Yang 方案中属性授权机构还需要为每个属性持有一个版本密钥. 与 Yang 方案相比，本方案不需要为每个属性持有一个版本密钥，但需要维护一个时间参数.

在 Yang 方案中，公钥参数和公钥属性密钥都为数据拥有者贡献了存储开销，这与系统中的属性总数呈线性关系. 在 Hur 方案和 Yang 方案中，加密数据带来的存储开销是相同的，本方案与 Yang 方案中数据拥有者的存储开销相同.

Yang 方案只需要云存储中心存储密文，而 Hur 方案中的云存储中心需要同时存储消息头和密文，且密文与系统用户数呈线性关系. 本方案与 Yang 方案相比，还需存储被撤销属性以及被撤销用户的标识 U_{id}.

在 Yang 方案中，每个用户的存储开销与它拥有的属性数量相关. 而在 Hur 方案中，每个数据用户的存储开销不仅与它拥有的属性数量呈线性关系，而且与系统中的用户数量呈线性关系. 本方案中数据用户的存储开销与 Yang 方案相同. 通常，用户的数量远大于系统中属性的数量，这意味着本方案和 Yang 方案中数据用户产生更少的存储开销.

表 1　不同方案的存储开销对比

Tab. 1　Comparison of storage overhead for different schemes

	Hur 方案	Yang 方案	本方案
AA	$2\|p\|$	$(4+n_a)\cdot\|p\|$	$4\|p\|+\|T\|$
DO	$2\|g\|+\|gT\|$	$(2+n_a)\|g\|+\|gT\|$	$(2+n_a)\|g\|+\|gT\|$
CC	$2\|gT\|+(3l+3)\|g\|+\frac{l\cdot\|n_u\|\cdot\|p\|}{2}$	$\|gT\|+(3l+1)\|g\|$	$\|gT\|+r\|u_{id}\|+(3l+1)\|g\|$
DU	$(2n_{a,i}+1)\cdot\|g\|+log(n_u+1)\|p\|$	$(2+n_{a,i})\cdot\|g\|$	$(2+n_{a,i})\cdot\|g\|$

5.2　通信开销

如表 2 所示，系统的通信开销主要是由密钥和密文造成的. 在 Yang 方案中，属性授权机构与数据用户之间的通信开销来自于数据用户的密钥和更新密钥. 而在 Hur 方案中，属性授权机构与数据用户之间的通信开销只与密钥有关. 本方案中属性授权机构与数据用户之间的通信开销同样来自数据用户的密钥和更新密钥，与 Yang 方案不同的是，本方案只更新被撤销用户的密钥，通信开销小很多.

在 Hur 方案中，公钥耗费了属性授权机构与数据拥有者之间的大部分通信开销. 在 Yang 方案中，当需要进行属性撤销时数据拥有者需要获取被撤销属性的最新公共属性密钥. 本方案与 Hur 方案相同，主要是分发密钥贡献了属性授权机构与数据拥有者之间的通信开销.

在 Yang 方案中，云存储中心和数据用户之间的通信开销来自于密文. 但在 Hur 方案中，除了密文之外，消息头也会增加云存储中心与数据用户之间的通信开销，该开销与系统中所有用户的数量呈线性关系. 本方案无额外的消息头，故而通信开销与 Yang 方案处于一个数量级.

密文是云存储中心和数据拥有者之间通信的主要开销. 由于本方案与 Yang 方案中密文的大小比 Hur 方案中的密文小得多，因此云存储中心与数据拥有者之间的通信开销比 Hur 方案中的密文小得多.

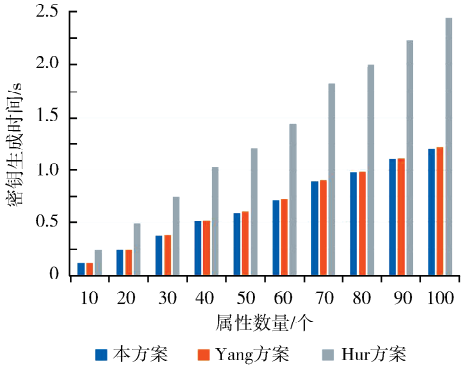
表 2　不同方案的通信开销比较

Tab. 2　Comparison of communication cost for different schemes

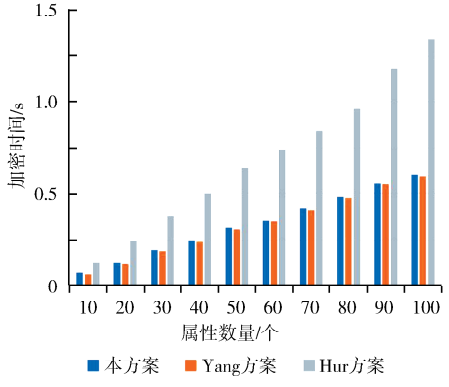
	Hur 方案	Yang 方案	本方案
AA&DU	$\|g\|+2n_{a,i}\|g\|$	$n_{a,i}\|g\|+2\|g\|+2n_r\|g\|$	$4\|g\|+n_{a,i}\|g\|$
AA&DO	$2\|g\|+\|g_T\|$	$2\|g\|+\|g_T\|+n_a\|g\|$	$2\|g\|+\|g_T\|$
CC&DU	$\|g_T\|+(2l+1)\|g\|+(l\cdot\|n_u\|/2+log(n_u+1))\|p\|$	$(3l+1)\|g\|+\|g_T\|$	$(3l+1)\|g\|+\|g_T\|$
CC&DO	$(l+1)\|g_T\|+2l\|g\|$	$(3l+1)\|g\|+\|g_T\|$	$(3l+1)\|g\|+\|g_T\|$

5.3　计算效率

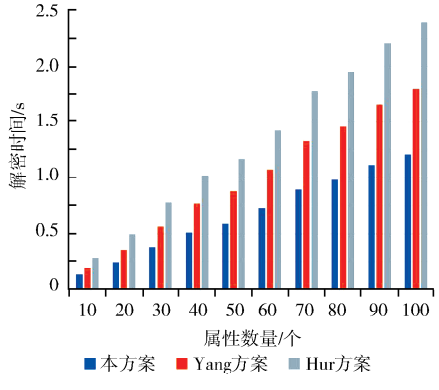
我们在 Windows 系统上实现本文方案、Hur 方案[7]和 Yang 方案[8]. 该系统采用 AMD Ryzen 54 600 G CPU,主频 3.7 GHz,内存 16 GB. 代码使用 JPBC 库实现上述方案. 使用 512 bit 有限域上的超奇异曲线 $y^2=x^3+x$ 中的 160 bit 椭圆曲线群. 实验数据取运行 20 次所得的平均值. 在实验中,JPBC 库进行一次双线性对运算的时间大约为 4.6 ms,进行一次群 G 中指数运算的时间大约为 6.0 ms,进行一次群 G_T 中指数运算的时间大约为 0.6 ms. 我们主要对本方案与现有方案在密钥生成时间、加密时间、解密时间与重加密时间方面进行了比较.



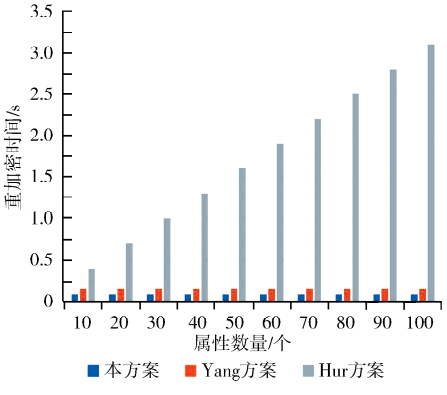
(a)密钥生成时间
(a) Key generation time



(b)加密时间
(b) Encryption time



(c)解密时间
(c) Decryption time



(d)重加密时间
(d) Re-encryption time

图 3　各阶段不同方案的仿真时间对比

Fig. 3　Comparison of the simulation time at each stage for different schemes

如图 3a 所示,密钥生成时间随着用户的属性个数呈线性增长,Hur 方案中的密钥生成包括属性授权机构执行的属性密钥生成算法和云存储中心执行的 KEK 生成算法,故其密钥生成时间比 Yang 方案以及本方案要长.

如图 3b 所示,加密时间与系统中总数呈线性关系. Yang 方案中的加密阶段所需时间比 Hur 方案更少. 这是由于在 Hur 方案中,数据所有者首先运行 CP-ABE 方案中的数据加密算法加密数据,然后将密文发送到云存储中心. 云存储中心收到数据所有者发送的密文后,会使用随机生成的加密指数对密文进行重新加密. 然后,云存储中心用广播

加密方法以一组属性组密钥加密该指数. 本方案与 Yang 方案的加密算法类似，故加密时间与 Yang 方案相差不大.

如图 3c 所示，各方案的解密时间都随着解密属性数量的增长而增长. 其中，Hur 方案的数据解密阶段包含了对消息头 *Hdr* 的属性组密钥解密和对密文的解密，所需时间较长. Yang 方案中的密文在密文更新阶段使用代理重加密技术，而本方案使用的是异或运算进行密文更新，故在解密时所需解密时间小于 Yang 方案.

如图 3d 所示，用户的某个属性被撤销时，需要对相应的密文进行更新. Yang 方案只需更新与被撤销属性相关的密文组件，而 Hur 方案需要重新加密密文的所有组件. 此外，Hur 方案中的重加密需要产生新的加密指数，故其重加密时间比 Yang 方案所需时间长. 本方案也只需更新与被撤销属性相关的密文. 不同的是，本方案使用异或运算进行密文更新，所需时间比 Yang 方案更少.

6　结　论

本文提出一个基于异或运算的属性撤销 CP-ABE 方案，通过使用被撤销用户的标识以及新的时间参数对密文进行异或运算实现了细粒度的属性撤销. 通过挑战者与攻击者之间的安全博弈，本文证明了方案的安全性. 理论分析和实验结果表明，该方案能有效降低密钥更新时的通信开销以及密文更新时的计算开销. 未来我们将继续优化云存储中心所需的存储开销.

参考文献：

[1] Qin J C, Wang Y, Dong Y C. Investigating the impact of risk attitude and privacy protection on consumers' data sharing behavior[J]. J Sichuan Univ (Nat Sci Ed), 2021, 58: 067002.[秦军昌，王渊，董玉成. 风险态度和隐私保护对消费者共享数据行为影响的机制研究[J]. 四川大学学报(自然科学版)，2021，58：067002.]

[2] Rivera D, García A, Martín-Ruiz M L, *et al*. Secure communications and protected data for an internet of things smart toy platform [J]. IEEE Int Things J, 2019, 1: 1.

[3] Zhang J, Wang B, Wang X A, *et al*. New group user based privacy preserving cloud auditing protocol [J]. Future Gener Comput Syst, 2020, 106: 585.

[4] Wang Y D, Yang J H, Xu C, *et al*. Survey on access control technologies for cloud computing[J]. J Software, 2015, 26: 1129.[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，2015，26：1129.]

[5] Shao L, Niu W N, Zhang X S. Cybersecurity threats assessment of self-service terminals in IoT application scenarios and corresponding countermeasures[J]. J Sichuan Univ(Nat Sci Ed), 2023, 60: 013004.[邵林，牛伟纳，张小松. 物联网应用场景下自助终端网络安全威胁评估与应对[J]. 四川大学学报(自然科学版)，2023，60：013004.]

[6] Fang L, Yin L H, Guo Y C, *et al*. A survey of key technologies in attribute-based access control scheme [J]. Chin J Comp, 2017, 40: 1680.[房梁，殷丽华，郭云川，等. 基于属性的访问控制关键技术研究综述[J]. 计算机学报，2017，40：1680.]

[7] Hur J, Dong K N. Attribute-based access control with efficient revocation in data outsourcing systems [J]. IEEE Trans Parallel Distrib Syst, 2011, 22: 1214.

[8] Yang K, Jia X, Ren K. Attribute-based fine-grained access control with efficient revocation in cloud storage systems [C]//Acm SIGSAC Symposium on Information. New York: ACM, 2013.

[9] Yan X X, Ye Q, Liu Y. Attribute-based encryption scheme supporting privacy preserving and user revocation in the cloud environment [J]. Netinfo Secur, 2017, 17: 14.[闫玺玺，叶青，刘宇. 云环境下支持隐私保护和用户撤销的属性基加密方案[J]. 信息网络安全，2017，17：14.]

[10] Xue L, Yu Y, Li Y, *et al*. Efficient attribute-based encryption with attribute revocation for assured data deletion [J]. Inform Sci, 2018, 2018: 640.

[11] Dong G F, Lu Y K, Zhang C W, *et al*. CP-ABE key update method supporting revocation attribute [J]. Appli Res Comp, 2023, 40: 6.[董国芳，鲁烨堃，张楚雯，等. 支持撤销属性的 CP-ABE 密钥更新方法[J]. 计算机应用研究，2023，40：6.]

[12] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption [C]//IEEE Symposium on Security & Privacy. Piscataway: IEEE, 2007.

[13] Zhao Z Y, Zhu Z Q, Wang J H, *et al*. Revocable attribute-based encryption with escrow-free in cloud storage[J]. J Electr Inform Technol, 2018, 40: 1.[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.]

[14] Fan Y D, Wu X P. Cloud storage access control scheme based on policy hiding attribute encryption [J]. Comp Eng, 2018, 44: 139.[范运东，吴晓平. 基于策略隐藏属性加密的云存储访问控制方案[J]. 计算机工程，2018，44：139.]