属性可撤销且密文长度恒定的属性基加密方案

赵志远,朱智强,王建华,孙 磊
(信息工程大学,河南郑州 450001)

摘 要: 密文策略属性基加密(ciphertext-policy attribute-based encryption,CP-ABE)类似于基于角色访问控制,可以为云存储系统提供灵活细粒度的访问控制.但大多数 CP-ABE 方案中,密文长度与访问策略复杂度成正相关,系统属性同时被多个用户共享而导致属性难以被撤销.针对上述问题,本文提出一种支持属性撤销且密文长度恒定的属性基加密方案.该方案中每个用户的属性群密钥不能通用,可以有效抵抗撤销用户与未撤销用户的合谋攻击.为减少属性授权机构和数据拥有者的计算负担,属性撤销过程所需的计算量外包给数据服务管理者;同时该方案采用支持多值属性和通配符的"AND"门策略,实现了密文长度恒定.所提方案基于决策性 q-BDHE (q-bilinear Diffie-Hellman exponent)假设对方案进行了选择明文攻击的安全性证明.最后对方案进行了理论分析与实验验证,分析结果表明本文方案可以有效抵制用户合谋攻击,增加了方案的安全性.同时所提方案在功能和计算效率方面具有一定优势,适用于实际应用情况.

关键词: 属性基加密; 属性撤销; 合谋攻击; 密文长度恒定
中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2018)10-2391-09
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2018.10.012

Attribute-Based Encryption with Attribute Revocation and Constant-Size Ciphertext

ZHAO Zhi-yuan,ZHU Zhi-qiang,WANG Jian-hua,SUN Lei
(*Information Engineering University*,*Zhengzhou*,*Henan* 450001,*China*)

Abstract: Ciphertext-Policy Attribute-Based Encryption (CP-ABE) is similar to role-based access control, which provides flexible and fine-grained access control for cloud storage systems. However, in most of existing CP-ABE schemes, the ciphertext length is positively related to the complexity of the access structure. And the attribute level user revocation is an important challenge because the system attributes are shared by multiple users at the same time. To solve this problem, this paper presents an CP-ABE scheme that supports the attribute level user revocation and constant-size ciphertext. The attribute group key for each user in the scheme is different, so this scheme can effectively resist collusion attacks between the revoked users and the existing users. To reduce the computational burden of the attribute authority and the data owner, the amount of computation required for the attribute revocation process is outsourced to the data service manager. At the same time, the scheme adopts the AND-Gate strategy supporting multi-valued attributes and wildcards, and the ciphertext length is constant. The scheme is proved selectively secure based on Decisional q-Bilinear Diffie-Hellman Exponent (q-BDHE) assumption. Finally, the functionality and efficiency of the proposed scheme are analyzed and verified. The experimental results show that the proposed scheme can safely implement attribute level user revocation. At the same time, the proposed scheme has some advantages in terms of function and computational efficiency. It is suitable for practical application.

Key words: attribute-based encryption; attribute revocation; collusion attacks; constant-size ciphertext

收稿日期:2017-07-12;修回日期:2018-03-08;责任编辑:李勇锋
基金项目:国家 973 重点基础研究发展计划(No. 2013CB338000);国家重点研发计划(No. 2016YFB0501900)

1 引言

密文策略属性基加密方案(ciphertext-policy attribute-based encryption,CP-ABE)[1]作为一种新型的公钥加密体制,其具有"一对多"的加密特征,可以根据用户属性实现对数据灵活细粒度的访问控制,是解决当前云存储安全问题的关键支撑技术之一[2,3]. CP-ABE 在实际应用过程中,其密文长度是一个重要的技术指标. 现有大多数 CP-ABE 方案,密文长度往往随着访问策略复杂度的增加而线性增长,从而导致数据密文占用大量云存储资源及通信资源;CP-ABE 在应用过程中的另一个安全指标是属性可撤销. 因为云存储系统中拥有大量的用户,在系统运行过程中一些用户的相关属性会发生变化,或者一些私钥可能被泄露,因此撤销或更新每一个属性的私钥组件对于系统安全至关重要. CP-ABE 中每一个属性有可能被多个用户共享,这意味着撤销任何属性都有可能影响其他用户,因此属性撤销是一个极其困难的问题.

为解决密文长度问题,文献[4]第一次基于支持多值属性的"AND"门提出密文长度恒定的 CP-ABE 方案. 该方案在解密阶段所需双线性对的计算量为常数级,并且在随机预言机模型下证明了方案的安全性. 文献[5]提出一种支持灵活陷门访问策略且密文长度恒定的 CP-ABE 方案,该方案达到了标准模型下的适应性选择密文攻击安全. 文献[6]基于支持多值属性和通配符的"AND"门提出一种密文长度恒定的 CP-ABE 方案,该方案在解密阶段只需 2 个双线性对计算,并在随机预言机模型下证明了方案的安全性. 文献[7]基于"AND"门提出密文长度和密钥长度恒定的 CP-ABE 方案,并在随机预言机模型下证明了方案的选择性选择密文攻击安全. 但是上述方案没有考虑属性撤销,无法解决因用户属性变化等导致的权限变更问题.

针对属性撤销问题,文献[8]最先提出 ABE 属性撤销方案,其通过对每一个属性设定一个有效期,属性授权机构周期性地更新属性版本,通过更新某个属性的版本以此达到用户属性撤销的目的. 这种通过给每个属性设定时间周期来达到撤销目的的方法是一种粗粒度的撤销方案,其不能实现属性或用户的立即撤销. 这种撤销方案存在一个不受控制的时期被称为脆弱性窗口,其影响方案的前向安全和后向安全[9].

文献[10]提出一种具有属性和用户撤销能力的 CP-ABE 方案,该方案增强了用户访问控制的前向安全和后向安全. 但是,该方案不能抵抗撤销用户与未撤销用户的合谋攻击,其原因在于属性群密钥 KEK 对于该群的用户完全通用. 文献[11]提出一种能够实现共享数据的细粒度访问控制的可撤销属性基加密方案,且该方案基于复杂假设完成安全证明. 文献[12]提出抵抗合谋攻击的属性撤销方案,并在随机预言机模型下证明了方案的安全性,但该方案解密效率不高. 文献[13]提出具有隐私保护且支持用户撤销的属性基加密方案. 该方案采用半策略隐藏方式实现隐私保护,并且能够实现属性级用户撤销. 但是该方案与方案[10]类似,不能抵抗撤销用户与未撤销用户的合谋攻击.

文献[14]提出一种密文长度恒定且属性直接可撤销的 CP-ABE 方案,该方案的"AND"门策略支持多值属性和通配符. 该方案同时考虑了密文长度恒定和属性撤销能力. 但是该方案属于属性直接撤销方案,其数据拥有者需要维护属性撤销列表,而该工作对于数据拥有者是繁琐的.

针对上述问题,本文提出一种支持属性即时撤销且密文长度恒定的属性基加密方案. 该方案中每个用户的属性群密钥不能通用,可以有效抵抗撤销用户与未撤销用户的合谋攻击. 为减少属性权威机构和数据拥有者的计算负担,属性撤销过程所需的计算量外包给数据服务管理者(归属于云服务商);同时该方案采用支持多值属性和通配符的"AND"门策略,实现了密文长度恒定. 所提方案基于决策性 q-BDHE (q-bilinear Diffie-Hellman exponent)假设对方案进行了选择明文攻击的安全性证明. 最后对方案进行了理论分析与实验验证,分析结果表明本文方案可以有效抵制用户合谋攻击,增加了方案的安全性. 同时所提方案在功能和计算效率方面具有一定优势,适用于实际应用情况.

2 理论基础知识

2.1 双线性群

双线性群是密码系统中重要的关键技术. 令 ψ 是一个群生成算法,其以安全参数 λ 作为输入,输出(p,G,G_T,e). 其中 p 是由 λ 决定的素数,G 和 G_T 是阶为素数 p 的循环群. 双线性映射 $e:G\times G\rightarrow G_T$ 满足下列性质:

(1)双线性:对于 $\forall u,v\in G$ 和 $\forall a,b\in Z_p$,均有 $e(u^a,v^b)=e(u,v)^{ab}$ 成立.

(2)非退化性:$\exists g\in G$,使得 $e(g,g)\neq 1$.

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

2.2 决策性 q-BDHE 假设

令 G 表示阶为大素数 p 的双线性群,g 和 h 为群 G 的两个独立的生成元,选择随机值 $\delta\in Z_p^*$. 定义 $\boldsymbol{y}_{g,\delta,n}=(g_1,g_2,\cdots,g_n,g_{n+2},g_{n+3},\cdots,g_{2n})\in G^{2n-1}$,其中 $g_i=g^{(\delta^i)}$. 算法 P 通过输出 $z\in\{0,1\}$ 进行猜测,若:

$$Adv_P^{q-BDHE}=|\Pr[p(g,h,\boldsymbol{y}_{g,\delta,n},e(g_{n+1},h))=0]-\Pr[p(g,h,\boldsymbol{y}_{g,\delta,n},Z)=0]|\geqslant\varepsilon \quad (1)$$

那么定义算法 P 拥有优势 ε 来解决群 G 下的决策

性 q-BDHE 问题. 若无多项式时间算法以不可忽略的优势来解决决策性 q-BDHE 问题,那么声称决策性 q-BDHE 假设在群 G 和 G_T 中是成立的.

2.3　访问策略

本文采用一种支持多值属性和通配符的“AND”门访问策略. 假设系统中共有 n 个属性,则该属性集合为 $U=\{att_1,att_2,\cdots,att_n\}$;每个属性 att_i 能够拥有多个属性值 $S_i=\{att_{i,1},att_{i,2},\cdots,att_{i,n_i}\}$,即 $n_i=|S_i|$. 假设解密者拥有属性列表 $L=[L_1,L_2,\cdots,L_n]$,加密者定义访问策略 $W=[W_1,W_2,\cdots,W_n]=\Lambda_{i\in I_W}W_i$,其中下标索引集合 $I_W=\{i|1\leqslant i\leqslant n,W_i\neq *\}$. 对于满足 $1\leqslant i\leqslant n$ 的 i,若 $L_i=W_i$ 或者 $W_i=*$,则 L 满足 W,即 $L|=W$;否则 L 不满足 W,即 $L|\neq W$. 访问策略 W 中的“$*$”意味着“不关心”值.

2.4　密钥加密密钥树

密钥加密密钥树[13]是指数据服务管理者(data service manager,DSM)基于用户集合建立的完全二叉树,为未撤销用户提供密钥更新能力,从而实现用户撤销,示意图如图 1 所示. 假设系统用户集合 $U=\{u_1,u_2,\cdots,u_N\}$,属性集合 $W=\{att_1,att_2,\cdots,att_n\}$. 设 $G_i\subset U$ 是拥有属性 att_i 的用户集合,被称为属性群. G_i 将被看作是能够访问属性 att_i 的访问列表. 设 $G=\{G_1,G_2,\cdots,G_n\}$ 是属性群集合. 例如:若 u_1,u_2,u_3 分别拥有属性 $\{att_1,att_2\}$,$\{att_1,att_2,att_3\}$,$\{att_2,att_3\}$. 那么 $G_1=\{u_1,u_2\}$,$G_2=\{u_1,u_2,u_3\}$,$G_3=\{u_2,u_3\}$.

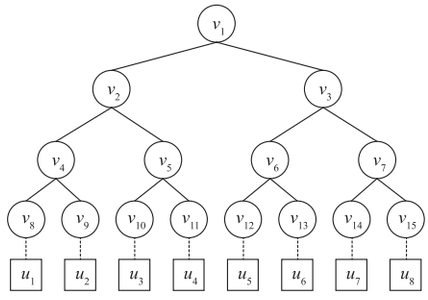


图1　密钥加密密钥树示意图

DSM 按如下过程计算构建密钥加密密钥树为用户生成属性群密钥相关参数:

(1)用户集合 U 中每一个用户被指定在二叉树的叶子节点中,每个节点 v_j 存储一个随机值 θ_j.

(2)路径节点生成算法 $Path(u_k)$. 对于每一个用户 u_k,从叶子节点到根节点上所有节点被定义为用户 u_k 的路径节点. 如 $Path(u_6)=\{v_{13},v_6,v_3,v_1\}$.

(3)最小覆盖集算法 $Mincs(G_i)$. 对于拥有属性 att_i 的属性群 G_i,树中能覆盖 G_i 的所有用户的最小节点集合为最小覆盖集. 如 $G_i=\{u_1,u_2,u_4,u_6,u_7,u_8\}$,则 $Mincs(G_i)=\{v_4,v_{11},v_{13},v_7\}$.

(4)求 $Path(u_k)$ 与 $Mincs(G_i)$ 的交集:若用户拥有属性 att_i,即 $u_k\in G_i$,则交集有且只有一个节点 v_j 存储的随机值 θ_j. 如(2)和(3)中的 u_6 只拥有节点 v_{13} 存储的随机值 θ_{13}.

3　ARCSC-ABE 方案系统及安全模型

3.1　形式化定义

本文提出的 ARCSC-ABE (ABE with attribute revocation and constant-size ciphertext)方案主要由属性授权机构(attribute authority,AA)、云服务商(包括 DSM,计算服务和存储服务)、数据拥有者和数据用户 4 部分组成. 该方案包含以下 5 个阶段:

(1)系统初始化阶段.

$AASetup(1^\lambda)\rightarrow\{PK,MSK\}$:AA 执行该算法进行系统初始化. 该算法以隐含安全参数 λ 作为输入,输出系统公钥 PK 和系统主私钥 MSK.

$DSMSetup(PK)\rightarrow\{DPK,DSK\}$:DSM 运行该算法进行初始化. 该算法以系统公钥 PK 作为输入,输出 DSM 的公钥 DPK 和主私钥 DSK. 当系统发生用户属性撤销时,DSM 的公私钥对将被更新.

(2)私钥生成阶段.

$AAKeyGen(id,PK,DPK,MSK,L)\rightarrow\{SK_L,KEK'\}$:该算法由 AA 执行,其以系统公钥 PK,DSM 公钥 DPK,系统主私钥 MSK 和用户属性集合 L 作为输入,输出用户私钥 SK_L 和属性群初始密钥 KEK'.

$DSMKeyGen(KEK',L)\rightarrow KEK$:该算法由 DSM 执行,其以 KEK' 和用户属性集合 L 作为输入,输出用户属性群密钥 KEK.

(3)数据加密阶段.

$Encrypt(PK,W,m)\rightarrow CT'$:该算法由数据拥有者执行,其以系统公钥 PK,访问策略 W 和明文消息 m 作为输入,输出中间密文 CT'.

$DSMEncrypt(PK,DSK,CT')\rightarrow\{Hdr,CT_W\}$:该算法由 DSM 执行,其以系统公钥 PK,DSM 公钥 DPK 和中间密文 CT' 作为输入,输出密文头 Hdr 和最终密文 CT_W.

(4)数据解密阶段.

$Decrypt(PK,Hdr,CT_W,SK_L,KEK)\rightarrow m$:数据用户运行该算法,其以系统公钥 PK,密文头 Hdr,密文 CT_W,用户私钥 SK_L 和 KEK 作为输入. 当数据用户的属性满足密文的访问策略时,且用户必要属性未被撤销,输出明文数据 m.

(5)用户属性撤销阶段.

$UpKEK(DSK,KEK,L_x)\rightarrow\overline{KEK}$:该算法由 DSM 执行,其以 DSM 私钥 DSK,属性群密钥 KEK 和被撤销属性 L_x 作为输入,输出新的属性群密钥 $\overline{KEK}$.

ReEncryption$(Hdr, CT_W, L_x) \rightarrow \{\overline{Hdr}, \overline{CT}\}$:该算法由DSM执行,其以密文头 Hdr,密文 CT 和被撤销属性 L_x 作为输入,输出新的密文头$\overline{Hdr}$和密文$\overline{CT}_W$.

3.2 安全模型

该方案能够抵抗撤销用户与未撤销用户的合谋攻击. 因此敌手可以询问两种类型的密钥:(1) 被撤销用户的私钥询问,该用户拥有满足访问策略的属性集合,但挑战属性一定被撤销;(2) 未被撤销用户的私钥询问,该用户的属性集合不满足访问策略,但其属性集合包含挑战属性. 安全模型如下:

系统初始化:敌手 $\mathcal{A}$ 选择一个要挑战的访问策略 W^* 和挑战属性 L_x^*,然后将它们传送给仿真者 $\mathcal{B}$. 其中,L_x^* 是一个满足 W^* 的必要属性.

系统建立:$\mathcal{B}$ 运行 AASetup 和 DSMSetup 算法获得系统公钥 PK,系统主私钥 MSK,DSM 的公钥 DPK 和主私钥 DSK. 然后,$\mathcal{B}$ 更新关联属性 L_x^* 的密钥对$\overline{DPK}$和$\overline{DSK}$. 最后,$\mathcal{B}$ 将 $PK, DPK, \overline{DPK}$发送给 $\mathcal{A}$,自己保留 $MSK, DSK, \overline{DSK}$.

询问阶段 1:除哈希询问外,$\mathcal{A}$ 可以询问两种类型密钥:

(1)Type-I 私钥询问$\langle u_I, L_I\rangle$:用户 u_I 的属性集合 L_I 满足访问策略 W^*,但是 u_I 的属性 L_x^* 已经被撤销. $\mathcal{B}$ 运行 AAKeyGen 和 DSMKeyGen 算法获得 SK_{L_I}和 KEK_I,然后将它们发送给 $\mathcal{A}$.

(2)Type-II 私钥询问$\langle u_{II}, L_{II}\rangle$:用户 u_{II}的属性集合 L_{II}不满足访问策略 W^*,但是 u_{II}拥有属性 L_x^*. $\mathcal{B}$ 运行 AAKeyGen 和 DSMKeyGen 算法获得 $SK_{L_{II}}$和 KEK_{II},然后将它们发送给 $\mathcal{A}$.

挑战阶段:$\mathcal{A}$ 提交两个等长的消息 m_0 和 m_1. $\mathcal{B}$ 随机选择 $b \in \{0,1\}$,运行 Encrypt 和 DSMEncrypt 算法产生密文头 Hdr^* 和密文 $CT_{W^*,b}$,并将其发送给 $\mathcal{A}$.

询问阶段 2:类似询问阶段 1,$\mathcal{A}$ 继续向 $\mathcal{B}$ 询问私钥.

猜测阶段:$\mathcal{A}$ 输出一个值 $b' \in \{0,1\}$ 作为对 b 的猜测. 如果 $b'=b$,我们称 $\mathcal{A}$ 赢得了该游戏. $\mathcal{A}$ 在该游戏中的优势定义为:$Adv_{\mathcal{A}} = |\Pr[b'=b] - 1/2|$.

定义 1 若无多项式时间敌手以不可忽略的优势来攻破上述安全模型,那么本文提出可撤销属性加密方案是选择明文安全的.

4 ARCSC-ABE 方案构造

4.1 具体方案

(1)系统初始化阶段. AA 和 DSM 分别进行初始化,建立系统.

AASetup$(1^\lambda) \rightarrow \{PK, MSK\}$:该算法选择两个阶为素数 p 的乘法循环群 G 和 G_T,g 是群 G 的生成元,并且存在有效的双线性映射 $e: G \times G \rightarrow G_T$. AA 随机选取 $\alpha, \beta \in Z_p^*$ 和两个抵制合谋的哈希函数 $H_0: Z_p^* \times \{0,1\}^{\log_2 n} \times \{0,1\}^{\log_2 m} \rightarrow Z_p^*$,$H_1: Z_p^* \rightarrow G$. 其中,$m = \max_{i=1}^{n} n_i$. 然后,该算法计算 $X_{i,b_i} = g^{-H_0(\alpha \| i \| b_i)}$ 和 $Y_{i,b_i} = e(g,g)^{H_0(\beta \| i \| b_i)}$. 其中,$1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i$. 输出系统公钥 $PK = (g, \{X_{i,b_i}, Y_{i,b_i}\}_{1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i})$ 和主私钥 $MSK = (\alpha, \beta)$.

DSMSetup$(PK) \rightarrow \{DPK, DSK\}$:DSM 为每一个 $\{att_{i,b_i}\}_{1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i}$ 选择一个随机指数 $t_{i,b_i} \in Z_p^*$,然后计算 $T_{i,b_i} = g^{t_{i,b_i}}$. 输出 DSM 的公钥 $DPK = \{T_{i,b_i}\}_{1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i}$ 和主私钥 $DSK = \{t_{i,b_i}\}_{1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i}$.

(2)私钥生成阶段. 首先 AA 产生与属性集合相关联的私钥,然后 DSM 产生属性群密钥 KEK.

AAKeyGen$(id, PK, DPK, MSK, L) \rightarrow \{SK_L, KEK'\}$:该算法选择 $r_{id} \in Z_p^*$. 对于 $1 \leqslant i \leqslant n$,假设 $L_i = att_{i,b_i}$. 然后 AA 计算 $\bar{\theta}_{id,i} = \theta_{id,i,b_i} = g^{H_0(\beta \| i \| b_i)} H_1(r_{id})^{H_0(\alpha \| i \| b_i)}$,$kek_{i,b_i} = T_{i,b_i}^{r_{id}}$. 最后输出私钥 $SK_L = (r_{id}, \{\bar{\theta}_{id,i}\}_{1 \leqslant i \leqslant n})$ 和属性群初始密钥 $KEK' = \{L_i, kek_{i,b_i}\}_{1 \leqslant i \leqslant n}$.

DSMKeyGen$(KEK', L) \rightarrow KEK$:该算法按 2.4 节中密钥加密密钥树为用户生成属性群密钥. 对于 $L_i = att_{i,b_i}$,DSM 计算 $\varphi_{i,b_i} = Path(u_{id}) \cap Mincs(G_{i,b_i})$. 若 $\varphi_{i,b_i} = \varnothing$,DSM 停止计算;若 $\varphi_{i,b_i} \neq \varnothing$,DSM 计算 $KEK_{i,b_i} = (kek_{i,b_i})^{1/\theta_j} = g^{t_{i,b_i} r_{id}/\theta_j}$,其中随机值 θ_j 对应节点 $v_j \in \varphi_{i,b_i}$. 然后输出 $KEK = \{L_i, v_j, kek_{i,b_i}, KEK_{i,b_i}\}_{L_i \in L}$.

(3)数据加密阶段. 该阶段分两步执行,第一步数据拥有者指定访问策略加密密文,第二步 DSM 重新加密密文并产生密文头.

Encrypt$(PK, W, m) \rightarrow CT'$:假设 $W_i = att_{i,b_i}$,为了用访问策略 $W = \wedge_{i \in I_w} W_i$ 加密明文 $m \in G_T$,数据拥有者计算$\langle X_W, Y_W\rangle = \langle \prod_{i \in I_w} X_{i,b_i}, \prod_{i \in I_w} Y_{i,b_i}\rangle$. 然后数据拥有者随机选择 $s \in z_p^*$,计算 $C'_0 = m \cdot Y_W^s$,$C'_1 = g^s$ 和 $C'_2 = X_W^s$,输出中间密文 $CT' = (W, C'_0, C'_1, C'_2)$.

DSMEncrypt$(PK, DSK, CT') \rightarrow \{Hdr, CT_W\}$:对于访问策略中的 W_i,DSM 随机选择 $k_i \in Z_p$ 并调用 $Mincs(G_{i,b_i})$算法,然后重新加密中间密文 CT'获得 $C_1 = C'_1$,$C_2 = C'_2$,$C_0 = C'_0 \cdot \prod_{i \in I_w} e(g,g)^{k_i}$,输出重加密密文 $CT_W = (W, C_0, C_1, C_2)$. 另外,该算法计算密文头 $Hdr = \{v_j, E(k_i) = g^{-k_i \theta_j / t_{i,b_i}}\}_{v_j \in Mincs(G_{i,b_i}), i \in I_w}$. 最后,DSM 将 (CT_W, Hdr) 上传到云服务商进行存储.

(4)数据解密阶段. 当数据用户的属性满足密文的访问策略时,且用户属性未被撤销,可以通过以下过程计算获得明文.

Decrypt$(PK, Hdr, CT_W, SK_L, KEK) \rightarrow m$:数据用户首先计算 $\theta_{id,W} = \prod_{i \in I_w} \bar{\theta}_{id,i}$. 然后按下述公式计算获得明文消息 m:

$$m = \frac{C_0 \cdot \left(\prod_{i \in I_w} e(KEK_{i,b_i}, E(k_i))\right)^{1/r_{id}}}{e(\theta_{id,W}, C_1) \cdot e(H_1(r_{id}), C_2)} \tag{2}$$

(5)用户属性撤销阶段. 当发生用户属性撤销时,DSM 只更新用户的属性群密钥 KEK,以确保被撤销用户关联属性集合的私钥失效. 同时重加密相关密文,确保前向和后向安全.

UpKEK$(DSK, KEK, L_x) \rightarrow \overline{KEK}$:当用户 u_z 的属性 $L_x = att_{x,b_x}$ 被撤销时,DSM 随机选择 σ_{x,b_x} 并计算 $\bar{T}_{x,b_x} = T_{x,b_x}^{\sigma_{x,b_x}}$, $\bar{t}_{x,b_x} = t_{x,b_x} \cdot \sigma_{x,b_x}$,然后用 $\bar{T}_{x,b_x}$ 和 $\bar{t}_{x,b_x}$ 替代 DPK 和 DSK 中的 T_{x,b_x} 和 t_{x,b_x},最后获得新的 DSM 公钥 $\overline{DPK}$ 和主私钥 $\overline{DSK}$.

DSM 更新属性群 $\bar{G}_{x,b_x}$ 并重新计算 $Mincs(\bar{G}_{x,b_x})$. 若 $G_{x,b_x} = \{u_1, u_2, u_5, u_6, u_7, u_8\}$,则 $Mincs(G_{x,b_x}) = \{v_4, v_3\}$. 当 u_6 的属性 L_x 被撤销时,$\bar{G}_{x,b_x} = \{u_1, u_2, u_5, u_7, u_8\}$, $Mincs(\bar{G}_{x,b_x}) = \{v_4, v_{12}, v_7\}$.

对于每一个用户 $u_k \in \bar{G}_{x,b_x}$,数据服务管理者计算 $\bar{\varphi}_{x,b_x} = Path(u_k) \cap Mincs(\bar{G}_{x,b_x})$, $\overline{kek}_{x,b_x} = (kek_{x,b_x})^{\sigma_{x,b_x}}$, $\overline{KEK}_{x,b_x} = (\overline{kek}_{x,b_x})^{\frac{1}{\theta}}$. 其中,$\bar{\theta}_{j'}$ 对应节点 $\bar{v}_{j'} \in \bar{\varphi}_{x,b_x}$.

最后,DSM 用 $\{L_x, \bar{v}_{j'}, \overline{kek}_{x,b_x}, \overline{KEK}_{x,b_x}\}$ 替换 KEK 中的 $\{L_x, v_j, kek_{x,b_x}, KEK_{x,b_x}\}$.

ReEncryption$(Hdr, CT_W, L_x) \rightarrow \{\overline{Hdr}, \overline{CT}_W\}$:DSM 选择 $s', \bar{k}_x \in Z_p$,重加密密文 $\bar{C}_1 = g^{s+s'}$, $\bar{C}_2 = X_W^{s+s'}$, $\bar{C}_0 = m \cdot Y_W^{s+s'} \cdot \left(\prod_{i \in I_w} e(g,g)^{k_i}\right) \cdot e(g,g)^{(\bar{k}_x - k_x)}$. 最后,输出重加密密文 $\overline{CT}_W = (W, \bar{C}_0, \bar{C}_1, \bar{C}_2)$. 更新密文头:

$$\overline{Hdr} = \left\{ \begin{array}{l} \{\bar{v}_{j'}, E(\bar{k}_x) = g^{-\bar{k}_x \bar{\theta}_{j'}/t_{x,b_x}}\}_{\bar{v}_{j'} \in Mincs(\bar{G}_{x,b_x})}, \\ \{v_j, E(k_i) = g^{-k_i \theta_j / t_{i,b_i}}\}_{v_j \in Mincs(G_{i,b_i}), i \in I_w, i \neq x} \end{array} \right\} \tag{3}$$

4.2　安全证明

本文方案能够抵抗撤销用户与未撤销用户的合谋攻击. 本节基于 3.2 节中安全模型证明定理 1.

定理 1　若决策性 q-BDHE 假设在群 G 和 G_T 中成立,那么没有多项式时间敌手 $\mathcal{A}$ 能够以不可忽略的优势选择性地攻破本文方案.

证明　假设敌手 $\mathcal{A}$ 在进行 Type-I 和 Type-II 询问后,能以不可忽略的优势 $\varepsilon = Adv_A$ 选择性地攻破本文方案. 那么我们能够构造仿真者 $\mathcal{B}$ 以不可忽略的优势 $Adv_{\mathcal{B}} = \varepsilon/2$ 攻破决策性 q-BDHE 假设. 仿真者 $\mathcal{B}$ 输入随机决策性 q-BDHE 挑战 $(g, h, \boldsymbol{y}_{g,\delta,n}, Z)$,其中 $\boldsymbol{y}_{g,\delta,n} = (g_1, g_2, \cdots, g_n, g_{n+2}, \cdots, g_{2n}) \in G^{2n-1}$, Z 是 G_T 中的随机元素或者是 $e(g_{n+1}, h)$. 不失一般性,本文假定 $v \in \{0, 1\}$. 如果 $v = 0$,那么 $Z = e(g_{n+1}, h)$;如果 $v = 1$,那么 Z 是随机值. 在游戏交互过程中,仿真者 $\mathcal{B}$ 为敌手 $\mathcal{A}$ 提供随机预言机 H_0 和 H_1 询问,为保持一致性且抵抗合谋攻击,仿真者 $\mathcal{B}$ 保持两个列表 $\mathcal{L}_0$ 和 $\mathcal{L}_1$ 存储先前的询问结果. 仿真者 $\mathcal{B}$ 与敌手 $\mathcal{A}$ 可以按如下步骤模拟交互游戏过程.

系统初始化:敌手 $\mathcal{A}$ 选择将要挑战的访问策略 $W^* = \Lambda_{i \in I_{W^*}} W_i$, $I_{W^*} = \{1, 2, \cdots, w\}$ 为下标索引集合. 注意,$w \leqslant n$ 且 I_{W^*} 中索引不一定连续,然后选择一个挑战属性 $L_x^* = att_{x,b_x}$(其是第 x 个属性的 n_x 个属性值中的一个固定值),将它们传送给仿真者 $\mathcal{B}$. 其中 $x \in I_{W^*}$.

系统建立:$\mathcal{B}$ 选择 $\alpha, \alpha', \beta, \beta' \in Z_p^*$. 然后计算:

(1)对于 θ_j,假设 $W_x = att_{x,b_x}$,然后 $\mathcal{B}$ 计算

$$(X_{x,b_x}, Y_{x,b_x}) = \left(g^{H_0(\alpha \| x \| b_x)} \prod_{t \in I_{W^*} - |x|} g_{n+1-t}, e(g,g)^{H_0(\beta \| x \| b_x)} e(g,g)^{\alpha^{x+1}}\right) \tag{4}$$

若 $b \neq b_x$,仿真者 $\mathcal{B}$ 计算

$$(X_{x,b}, Y_{x,b}) = \left(g^{-H_0(\alpha' \| x \| b)}, e(g,g)^{H_0(\beta' \| x \| b)}\right) \tag{5}$$

(2)若 $i \in I_{W^*} - \{x\}$,假设 $W_i = att_{i,b_i}$,$\mathcal{B}$ 计算

$$(X_{i,b_i}, Y_{i,b_i}) = \left(g^{-H_0(\alpha \| i \| b_i)} g_{n+1-t}^{-1}, e(g,g)^{H_0(\beta \| i \| b_i)}\right) \tag{6}$$

若 $b \neq b_i$,仿真者 $\mathcal{B}$ 计算

$$(X_{i,b_i}, Y_{i,b_i}) = \left(g^{-H_0(\alpha \| i \| b)}, e(g,g)^{H_0(\beta \| i \| b)}\right) \tag{7}$$

(3)若 $i \notin I_{W^*}$,对于 $1 \leqslant b_i \leqslant n_i$,仿真者 $\mathcal{B}$ 计算

$$(X_{i,b_i}, Y_{i,b_i}) = \left(g^{-H_0(\alpha \| i \| b_i)}, e(g,g)^{H_0(\beta \| i \| b_i)}\right) \tag{8}$$

设置公钥 $PK = (g, X_{x,b_x}, Y_{x,b_x}, \{X_{i,b_i}, Y_{i,b_i}\}_{att_i \in U, i \neq x})$ 和主私钥 $MSK = (\alpha, \beta)$

对于每一个属性值 $\{att_{i,b_i}\}_{1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i}$,$\mathcal{B}$ 选择一个随机指数 $t_{i,b_i} \in Z_p^*$,然后计算 $T_{i,b_i} = g^{t_{i,b_i}}$,最后输出系统公钥 $DPK = \{T_{i,b_i} | 1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i\}$ 和系统主私钥 $DSK = \{t_{i,b_i} | 1 \leqslant i \leqslant n, 1 \leqslant b_i \leqslant n_i\}$. $\mathcal{B}$ 更新 $W_x^* = att_{x,b_x}$ 的相关密钥 $\bar{T}_{x,b_x} = (T_{x,b_x})^{\delta^n} = (g_n)^{t_{x,b_x}}$,设置 $\bar{t}_{x,b_x} = t_{x,b_x}\delta^n$. 然后 $\mathcal{B}$ 用 $\bar{T}_{x,b_x}$ 和 $\bar{t}_{x,b_x}$ 替换 DPK 和主私钥 DSK 中的 T_{x,b_x} 和 t_{x,b_x},更新 DSM 的公钥 $\overline{DPK}$ 和 DSM 的主私钥 $\overline{DSK}$.

询问阶段 1:敌手 $\mathcal{A}$ 可以进行哈希询问和两种类型密钥询问.

(1)$\mathcal{O}_{H_0}(\cdot)$ 询问:当输入“·”询问 H_0 时,仿真者 $\mathcal{B}$ 首先查询“·”是否已经存在于列表 $\mathcal{L}_0$ 中. 若是,则将先前存放的值返回,否则随机选择 $\eta \in Z_p^*$ 并在列表 $\mathcal{L}_0$ 中增加实体 $<\cdot, \eta>$,然后返回 η.

(2)$\mathcal{O}_{H_1}(r_{id})$ 询问:当输入 r_{id} 询问 H_1 时,仿真者 $\mathcal{B}$ 首先查看 r_{id} 是否在列表 $\mathcal{L}_1$ 中. 若是,则将先前存放的值返回,否则按如下过程计算:

①若 r_{id} 与在身份密钥询问过程中的 L_I 相匹配,仿真者 $\mathcal{B}$ 在列表 $\mathcal{L}_1$ 中增加 $<r_{id}, g_x g^z>$ 并返回 $g_x g^z$. 其中,$z \in Z_p^*$,x 是 L 的下标,且 $x \in I_{W^*}$,但属性 $L_x^* = att_{x,b_x}$ 已经被撤销.

②若 r_{id} 与在身份密钥询问过程中的 L_{II} 相匹配,$\mathcal{B}$ 在列表 $\mathcal{L}_1$ 中增加 $<r_{id}, g_y g^z>$ 并返回 $g_y g^z$. 其中,$z \in Z_p^*$,y 是 L 的下标,且 $y \notin I_{W^*}$.

③否则 $\mathcal{B}$ 随机选择 $i\in(1,2,\cdots,n)$ 和 $z\in Z_p^*$，在列表 $\mathcal{L}_1$ 中增加 $<r_{id},g_ig^z>$ 并返回 g_ig^z.

(3)Type-I 私钥询问$[u_I,L_I]$：当 $id=I$ 时，用户 u_I 的属性列表 L_I 满足访问策略 W^*，但是 u_I 的属性 $L_x^*=att_{x,b_x}$ 已经被撤销. $\mathcal{B}$ 随机选取 $r_I,z\in Z_p^*$：

①对于 $t=x$，$L_x^*=att_{x,b_x}$，然后仿真者 $\mathcal{B}$ 计算 $\bar{\theta}_{I,x}=\theta_{I,x,b_x}=g^{H_0(\beta\|x\|b_x)}g_x^{H_0(\alpha\|x\|b_x)}\left(\prod_{k\in I_{W^*}-|x|}g_{n+1-k}^{-1}\right)X_{x,b_x}^{-z}$.

②对于 $t\in I_{W^*}-\{x\}$，假设 $L_t=att_{t,b_t}$，仿真者 $\mathcal{B}$ 计算 $\bar{\theta}_{I,t}=\theta_{I,t,b_t}=g^{H_0(\beta\|t\|b_t)}(g_x)^{H_0(\alpha\|t\|b_t)}g_{n+1-t}(X_{t,b_t})^{-z}$.

③若 $t\notin I_{W^*}$，假设 $L_t=att_{t,b_t}$，仿真者 $\mathcal{B}$ 计算 $\bar{\theta}_{I,t}=\theta_{I,t,b_t}=g^{H_0(\beta\|t\|b_t)}(g_xg^z)^{H_0(\alpha\|t\|b_t)}$. 因为 L_I 满足访问策略 W^*，所以这种询问在解密时无用，其作用只是返回给敌手 $\mathcal{A}$ 一个值.

对于 $L_i\in L_I$，计算 $kek_{i,b_i}=T_{i,b_i}^{r_I}$，仿真者 $\mathcal{B}$ 计算 $\varphi_{i,b_i}=Path(u_I)\cap Mincs(G_{i,b_i})$ 和 $KEK_{i,b_i}=(kek_{i,b_i})^{1/\theta_j}=g^{t_{i,b}r_I/\theta_j}$. 其中，随机值 θ_j 所对应节点 $v_j\in\varphi_{i,b_i}$. 该情况 $L_x\in L_I$ 已经被撤销，不能进行更新询问. 最后，仿真者 $\mathcal{B}$ 将 $SK_{L_I}=(r_I,\{\bar{\theta}_{id,i}\}_{L_i\in L_I})$ 和 $KEK=\{L_i,v_j,kek_{i,b_i},KEK_{i,b_i}\}_{L_i\in L_I}$ 发送给 $\mathcal{A}$.

(4)Type-II 私钥询问$[u_{II},L_{II}]$：当 $id=II$ 时，用户 u_{II} 的属性集合 L_{II} 不满足访问策略 W^*，但是 u_{II} 拥有属性 $L_x^*=att_{x,b_x}$. 因为 $L_{II}|\neq W^*$，所以一定存在 $y\in I_{W^*}$ 使得 $L_y\notin W_y$. 不失一般性，假设 $L_y=att_{y,b_y}$ 和 $W_y=att_{y,b_y}$. 仿真者 $\mathcal{B}$ 选取一个随机值 $r_{II}\in Z_p^*$，然后计算 $\bar{\theta}_{II,y}=\theta_{II,y,b_y}=g^{H_0(\beta'\|y\|b_y)}(g_yg^z)^{H_0(\alpha'\|y\|b_y)}$.

对于 $t\neq y$，$\mathcal{B}$ 选择 $z\in Z_p^*$，按如下方式计算 $\bar{\theta}_{II,t}$：

①对于 $t=x$，则 $L_x^*=att_{x,b_x}$，仿真者 $\mathcal{B}$ 计算 $\bar{\theta}_{II,x}=\theta_{II,x,b_x}=g^{H_0(\beta\|x\|b_x)}g_y^{H_0(\alpha\|x\|b_x)}X_{x,b_x}^{-z}\prod_{k\in I_{W^*}-|x,y|}g_{n+1-k+y}^{-1}$.

②若 $t\in I_{W^*}-\{x\}$，假设 $L_t=att_{t,b_t}$，然后 $\mathcal{B}$ 计算 $\bar{\theta}_{II,t}=\theta_{II,t,b_t}=g^{H_0(\beta\|t\|b_t)}(g_y)^{H_0(\alpha\|t\|b_t)}g_{n+1-t+y}(X_{t,b_t})^{-z}$.

③若 $t\notin I_{W^*}$，假设 $L_t=att_{t,b_t}$，仿真者 $\mathcal{B}$ 计算 $\bar{\theta}_{II,t}=\theta_{II,t,b_t}=g^{H_0(\beta\|t\|b_t)}(g_yg^z)^{H_0(\alpha\|t\|b_t)}$.

对于 $i\neq x$ 且 $L_i\in S_{II}$，计算 $kek_{i,b_i}=T_{i,b_i}^{r_{II}}$，$\mathcal{B}$ 计算 $\varphi_{i,b_i}=Path(u_{II})\cap Mincs(G_{i,b_i})$ 和 $KEK_{i,b_i}=(kek_{i,b_i})^{1/\theta_j}=g^{t_{i,b}r_{II}/\theta_j}$. 其中，随机值 θ_j 所对应节点 $v_j\in\varphi_{i,b_i}$；对于 $L_x^*\in L_{II}$，计算 $kek_{x,b_x}=(\bar{T}_{x,b_x})^{r_{II}}$，$\varphi_{x,b_x}=Path(u_{II})\cap Mincs(G_{x,b_x})$，$KEK_{x,b_x}=(kek_{x,b_x})^{1/\theta_j^*}=g_n^{t_{x,b}r_{II}/\theta_j^*}$. 其中，随机值 θ_j^* 对应节点 $v_j^*\in\varphi_{x,b_x}$. 最后，仿真者 $\mathcal{B}$ 将 $SK_{L_{II}}=(r_{II},\{\bar{\theta}_{id,i}\}_{L_i\in L_{II}})$ 和 $KEK=\{L_i,v_j,kek_{i,b_i},KEK_{i,b_i}\}_{L_i\in L_{II}}$ 发送给 $\mathcal{A}$.

挑战阶段：$\mathcal{A}$ 提交两个等长的消息 m_0 和 m_1. 仿真者 $\mathcal{B}$ 随机选择 $b\in\{0,1\}$ 生成挑战密文 $C'_1=h$，$C'_2=h^{-\alpha_{W^*}}$，$C'_0=m_bY_{W^*}^s=m_bZe(g,h)^{\beta_{W^*}}$. 最后，输出中间密文 $CT'=(W^*,C'_0,C'_1,C'_2)$. 其中，

$$\alpha_{W^*}=\sum_{t\in I_{W^*}}H_0(\alpha\|t\|b_t)\tag{9}$$

$$\beta_{W^*}=\sum_{t\in I_{W^*}}H_0(\beta\|t\|b_t)\tag{10}$$

$$\begin{aligned}X_{W^*}&=X_{x,b_x}\prod_{t\in I_{W^*}-|x|}X_{t,b_t}\\&=g^{-H_0(\alpha\|x\|b_x)}\prod_{t\in I_{W^*}-|x|}g_{n+1-t}\prod_{t\in I_{W^*}-|x|}g^{-H_0(\alpha\|t\|b_t)}g_{n+1-t}^{-1}\\&=g^{-\alpha_{W^*}}\end{aligned}\tag{11}$$

$$\begin{aligned}Y_{W^*}&=Y_{x,b_x}\prod_{t\in I_{W^*}-|x|}Y_{t,b_t}\\&=e(g,g)^{H_0(\beta\|x\|b_x)}e(g,g)^{\delta^{n+1}}\prod_{t\in I_{W^*}-|x|}e(g,g)^{H_0(\beta\|t\|b_t)}\\&=e(g,g)^{\sum_{t\in I_{W^*}}H_0(\beta\|t\|b_t)+\delta^{n+1}}\end{aligned}\tag{12}$$

对于 $i\neq x$ 且 $W_i\in W^*$，$\mathcal{B}$ 随机选择 $k'_i\in Z_p$，并设置 $k_i=s\cdot k'_i$；对于 $W_x\in W^*$，$\mathcal{B}$ 随机选择 $k'_x\in Z_p$ 并设置 $k_x^*=s\delta^nk'_x$. 然后重新加密中间密文 CT' 获得：

$$\begin{aligned}C_0^*&=C'_0\cdot e(g_1,g)^{s\cdot k_x\delta^n}\cdot\prod_{i\in I_{W^*}-|x|}e(g,g)^{s\cdot k_i}\\&=C'_0\cdot Z^{k_x}\cdot\prod_{i\in I_{W^*}-|x|}e(g,h)^{k_i}\\C_1^*&=C'_1\\C_2^*&=C'_2\end{aligned}\tag{13}$$

最后，输出 $CT_{W^*,b}=(W^*,C_0^*,C_1^*,C_2^*)$. 另外，对于 $W_i\in W^*$，$\mathcal{B}$ 调用 $Mincs(G_{i,b_i})$ 算法，并计算密文头：

$$Hdr^*=\left\{\begin{matrix}\{v_j,E(k_i)=h^{-k'_i\theta_j/t_{i,b_i}}\}_{v_j\in Mincs(G_{i,b_i}),W_i\in W,i\neq x}\\\{v_j^*,E(k_x)=h^{-k'_x\theta_j^*/t_{x,b_x}}\}_{v_j^*\in Mincs(G_{x,b_x})}\end{matrix}\right\}\tag{14}$$

最后，$\mathcal{B}$ 将$(CT_{W^*,b},Hdr^*)$发送给 $\mathcal{A}$.

当 $Z=e(g_{n+1},h)$ 时，密文 $CT_{W^*,b}=(W^*,C_0^*,C_1^*,C_2^*)$ 是明文消息 m_b 的合法密文；当 Z 是 G_T 中随机元素时，在敌手眼里 $CT_{W^*,b}$ 是随机消息的密文.

询问阶段 2：类似询问阶段 1，$\mathcal{A}$ 继续向 $\mathcal{B}$ 询问私钥.

猜测阶段：$\mathcal{A}$ 输出一个值 $b'\in\{0,1\}$ 作为对 b 的猜测. 如果 $b'=b$，$\mathcal{B}$ 输出 0 表示猜测 $Z=e(g_{n+1},h)$；否则输出 1 表示猜测 Z 为群 G_T 中的随机元素. 这两种情况如下所述：

情况 1：当 $Z=e(g_{n+1},h)$ 时，即 $v=0$. 这种情况下 $CT_{W^*,b}=(W^*,C_0^*,C_1^*,C_2^*)$ 是一个可用的密文，$\mathcal{B}$ 能够提供一个有效的仿真. $\mathcal{A}$ 的优势为 $\varepsilon=Adv_{\mathcal{A}}$. 因此，$\Pr[\mathcal{B}(g,h,\boldsymbol{y}_{g,\delta,n},e(g_{n+1},h))=0]=1/2+Adv_{\mathcal{A}}$；

情况 2：当 Z 为群 G_T 中的随机元素时，即 $v=1$. 这时 m_b 对于敌手来说是完全随机的，因此我们可以得出：$\Pr[\mathcal{B}(g,h,\boldsymbol{y}_{g,\delta,n},Z)=0]=1/2$.

最后，我们能够得到：

$$\begin{aligned}Adv_{\mathcal{B}}&=\frac{1}{2}\Pr[\mathcal{B}(g,h,\boldsymbol{y}_{g,\delta,n},e(g_{n+1},h))=0]\\&\quad+\frac{1}{2}\Pr[\mathcal{B}(g,h,\boldsymbol{y}_{g,\delta,n},Z)=0]-\frac{1}{2}\end{aligned}$$

$$=\frac{1}{2}\left(\frac{1}{2}+Adv_A\right)+\frac{1}{2}\cdot\frac{1}{2}-\frac{1}{2}=\frac{Adv_A}{2}=\frac{\varepsilon}{2} \quad (15)$$

即,仿真者 $\mathcal{B}$ 能够以不可忽略的优势 $\varepsilon/2$ 攻破决策性 q-BDHE 假设. 基于上述过程,完成了本文方案的选择明文攻击的安全性证明. 证毕

5　方案分析及实验验证

5.1　理论分析

本节主要在功能性、计算成本和存储成本方面将本文方案与已有几种撤销方案进行对比. 对比过程中所使用描述符定义如下:$|p|$ 表示 Z_p 中数据元素的长度;$|g|$ 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;$|C_k|$ 表示密文头长度;$|K_k|$ 表示属性群密钥长度;l 表示访问策略 W(包括 $LSSS$)中属性的数量,Y 表示访问树 $\mathcal{T}$ 的叶子节点数,为简单起见,l 和 Y 统一用 n_c 表示;n_k 表示用户密钥中属性的个数;n_a 表示整个系统中属性的总个数;n_u 表示整个系统中用户的总个数;N 表示系统中属性取值的总数;n_d 表示解密所需属性的个数;r 表示撤销事件的数目;E_G 和 E_{G_T} 分别表示 G 和 G_T 中模指数计算,P 表示双线性对计算. 在计算成本对比中,模指数和双线性对的计算量相对于其他计算需要更多的计算时间,因此本文忽略了其他次要因素.

5.1.1　功能及计算成本对比

从表 1 可以看出,文献[11 ~ 13]方案具有较灵活的访问策略,文献[14]和本文方案采用支持多值属性和通配符的"AND"门访问策略(AND_m^*),其表达能力有所欠缺,但依然能满足大部分应用. 文献[12,13]方案基于简单假设,而文献[11,14]和本文方案基于复杂假设. 文献[14]和本文方案具有恒定密文长度,有效减少存储和通信成本,同时具有较小的计算量. 文献[11 ~ 13]和本文方案属于间接撤销,而文献[14]属于直接撤销. 间接撤销相对于直接撤销的优势在于数据拥有者不需要维护属性撤销列表. 另外文献[13]方案不能够抵抗撤销用户与未撤销用户的合谋攻击. 文献[14]和本文方案在加密过程中,数据拥有者只需恒定计算就可完成加密任务,与访问结构复杂度无关,有效减少数据拥有者的计算负担. 文献[14]和本文方案在解密过程中所需计算量同样小于其他方案. 综合分析,本文方案在表达能力方面有所欠缺,但在功能和计算效率方面有较大优势. 冯登国研究员指出"在目前属性密码构造中,由于访问策略的复杂性,方案的计算代价和通讯代价往往都比较高. 可以通过适当降低原有表达能力和安全需求来提高效率,并且这种情况在实际应用中是可以接受的[15]. 所以本文方案是实用的.

表 1　几种撤销方案功能及计算成本对比

方案	功能性				计算成本	
	访问策略	安全假设	撤销机制	定长密文	加密	解密
文献[11]方案	LSSS	决策性 q-Parallel BDHE	间接撤销	否	$(3n_c+1)E_G+E_{G_T}$	$n_dE_{G_T}+(2n_d+1)P$
文献[12]方案	Tree	CDH	间接撤销	否	$(2n_c+1)E_G+E_{G_T}$	$n_dE_{G_T}+(3n_d+1)P$
文献[13]方案	Tree	DBDH	间接撤销	否	$(4n_c+2)E_G+2E_{G_T}$	$n_kE_G+(4n_d+1)P$
文献[14]方案	AND_m^*	决策性 q-BDHE	直接撤销	是	$\leqslant 3E_G+2E_{G_T}$	$(4r+1)P$
本文方案	AND_m^*	决策性 q-BDHE	间接撤销	是	$2E_G+1E_{G_T}$	$n_dE_{G_T}+(n_d+2)P$

5.1.2　存储成本

表 2 将本文方案与其他相关方案进行了存储成本的对比. 本文属性授权机构的存储成本和其他方案属性授权机构的存储成本主要来自于主密钥. 文献[11,14]和本文方案使用了较少的主密钥,文献[12]方案的主密钥随着属性总数 n_a 成线性增长,文献[13]方案的主密钥与属性总数 n_a 和属性取值总数 N 之和成线性正相关. 数据拥有者的存储成本主要来自于公钥. 方案[12]使用了最少的公钥,其他 4 种方案数据拥有者的存储成本随着属性总数 n_a、用户总数 n_u 或者属性取值总数 N 的增长成线性增长. 云服务商的存储成本主要来自于密文,密文头. 方案[14]需要存储较少的密文. 而方案[12,13]和本文方案不仅要存储密文,还要存储密文头,所以本文方案虽然实现了密文长度恒定,但是密文头的长度与访问策略中属性数量 n_c 成线性正相关. 数据用户的存储成本主要来自于其拥有的私钥. 文献[12,13]和本文方案中,每个用户都要存储一定的属性群密钥和属性相关的私钥进行解密操作. 方案[14]需要额外存储一个大小为 $\log n_u$ 的序列号. 综合分析,本文方案在存储成本上与其他方案相当.

表 2 存储成本对比

方案	属性授权机构	数据拥有者	云服务商	数据用户
文献[11]方案	$\|p\|$	$(n_a+2)\|g\|+\|g_T\|$	$n_a\|p\|+(2n_c+1)\|g\|+\|g_T\|$	$(n_k+2)\|g\|$
文献[12]方案	$\|p\|+(n_a+1)\|g\|$	$3\|g\|+\|g_T\|$	$(2n_c+1)\|g\|+\|g_T\|+\|C_k\|$	$(2n_k+1)\|g\|+\|K_k\|$
文献[13]方案	$(n_a+N+1)\|g\|$	$(n_a+N+1)\|g\|+\|g_T\|$	$(4n_c+2)\|g\|+\|g_T\|+\|C_k\|$	$(n_k+1)\|g\|+\|K_k\|$
文献[14]方案	$3\|p\|$	$(2n_u+N+1)\|g\|$	$\leqslant 4\|g\|+\|g_T\|$	$(n_k+1)\|g\|+\|p\|+\log n_u$
本文方案	$2\|p\|$	$(2N+1)\|g\|$	$2\|g\|+\|g_T\|+\|C_k\|$	$n_k\|g\|+\|p\|+\|K_k\|$

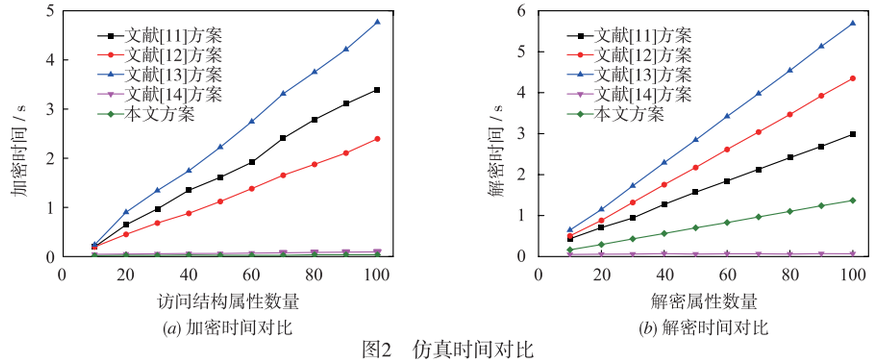
5.2 实验分析

实验环境为 64 bit Ubuntu 14.04 操作系统、Intel ® CoreTM i5-6200U(2.3 GHz)、内存 8GB,实验代码基于 Pairing-based Cryptography Library (PBC-0.5.14)与 cpabe-0.11 进行修改与编写,并且使用基于 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.

在本文方案中,由属性撤销导致的属性群密钥更新和密文更新操作都由具有强大计算能力的云服务商中的 DSM 完成,因此本文重点仿真由用户完成计算的加密和解密操作. 仿真中,本文设置了 10 种不同的访问策略,所涉及属性数量以 10 为增量,从 10 增加到 100. 对于每个访问策略,本文重复了 10 次实验,然后以平均值作为实验结果. 并保持每次实验都是完全独立的.

如图 2(*a*)所示,加密时间与访问策略中的属性数量成线性增长关系. 文献[11~13]方案中加密阶段需要为每个属性计算两个密文组件,导致这种线性增长关系. 这对于只有少量计算资源的用户来说是不可接受的. 而文献[14]和本文方案采用了定长密文技术,加密阶段只需要较少的计算量,且受访问策略中属性数量的增长影响非常小. 如图 2(*b*)所示,解密时间与解密所需属性数量成线性增长关系. 方案[14]在解密阶段用户只需承担较小的计算量就可完成解密任务(假设 $r=1$). 文献[11~13]方案的解密计算量与解密所需的属性数量成线性关系. 而本文方案在解密过程中,计算量虽然没有达到固定值,但是随属性数量变化的斜率较小.

综合分析,本文由于属性撤销导致的计算全部由云服务中的 DSM 完成,极大减轻了 AA 和用户的计算负担,数据拥有者只需定量计算就能完成数据加密任务,而数据用户需要较少计算就能够完成数据解密任务. 因此本文方案更加适用于属性频繁变化且计算资源有限的情况.



(*a*) 加密时间对比 (*b*) 解密时间对比

图2 仿真时间对比

6 结束语

本文提出一种支持属性即时撤销且密文长度恒定的属性基加密方案. 该方案中每个用户的属性群密钥不能通用,可以有效抵抗撤销用户与未撤销用户的合谋攻击. 为减少属性授权机构和数据拥有者的计算负担,属性撤销过程所需的计算量外包给数据服务管理者(归属于云服务商);同时该方案采用支持多值属性和通配符的"AND"策略,实现了密文长度恒定. 所提方案基于决策性 q-BDHE 假设对方案进行了选择明文攻击的安全性证明. 最后对方案进行了理论分析与实验验证,分析结果表明本文方案可以有效抵制用户合谋攻击,增加了方案的安全性. 同时所提方案在功能和计算效率方面具有一定优势,适用于实际应用情况.

参考文献

[1] BETHENCOURT J, SAHAI A, WATERS B. Ciphertext-policy attribute-based encryption[A]. Proceedings of the IEEE Symposium on Security and Privacy[C]. Washington: IEEE Computer Society, 2007. 321 – 334.

[2] 刘梦君,刘树波,等. 基于 LSSS 共享矩阵无授权策略的属性密码解密效率提高方案[J]. 电子学报,2015,43(6):1065 – 1072.
LIU Meng-jun, LIU Shu-bo, et al. Optimizing the decryption efficiency in LSSS matrix-based attribute-based en-

cryption without given policy[J]. Acta Electronica Sinica, 2015,43(6):1065 – 1072. (in Chinese)
[3] SOOKHAK M, YU F R, et al. Attribute-based data access control in mobile cloud computing: Taxonomy and open issues[J]. Future Generation Computer Systems, 2017, 72(C):273 – 287.
[4] EMURA K, MIYAJI A, et al. A ciphertext-policy attribute-based encryption scheme with constant ciphertext length [A]. Proceedings of the International Conference on Information Security Practice and Experience [C]. Berlin: Springer, 2009. 13 – 23.
[5] GE A, ZHANG R, et al. Threshold ciphertext policy attribute-based encryption with constant size ciphertexts [A]. Proceedings of the Australasian Conference on Information Security and Privacy [C]. Berlin: Springer, 2012. 336 – 349.
[6] ZHANG Y, ZHENG D, et al. Computationally efficient ciphertext-policy attribute-based encryption with constant-size ciphertexts[A]. Proceedings of the International Conference on Provable Security[C]. Berlin: Springer, 2014. 259 – 273.
[7] ODELU V, DAS A K, et al. Pairing-based CP-ABE with constant-size ciphertexts and secret keys for cloud environment[J]. Computer Standards & Interfaces, 2016, 54(P1): 3 – 9.
[8] PIRRETTI M, TRAYNOR P, et al. Secure attribute-based systems [A]. Proceedings of the ACM Conference on Computer and Communications Security[C]. New York: ACM, 2006. 99 – 112.
[9] RAFAELI S, HUTCHISON D. A survey of key management for secure group communication[J]. ACM Computing Surveys, 2003, 35(3): 309 – 329.
[10] 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.
[11] SHIRAISHI Y, NOMURA K, et al. Attribute revocable attribute-based encryption with forward secrecy for fine-grained access control of shared data[J]. IEICE Transactions on Information and Systems, 2017, 100(10): 2432 – 2439.
[12] LI Ji-guo, YAO Wei, et al. User collusion avoidance CP-ABE with efficient attribute revocation for cloud storage [J]. IEEE Systems Journal, 2017, PP(99): 1 – 11.
[13] 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(6): 14 – 21.
YAN Xi-xi, YE Qing, LIU Yu. Attribute-based encryption scheme supporting privacy preserving and user revocation in the cloud environment[J]. Netinfo Security, 2017(6): 14 – 21. (in Chinese)
[14] 张应辉,郑东,等. 密文长度恒定且属性直接可撤销的基于属性的加密[J]. 密码学报, 2014, 1(5): 465 – 480.
ZHANG Ying-hui, ZHENG Dong, et al. Attribute directly-revocable attribute-based encryption with constant ciphertext length[J]. Journal of Cryptologic Research, 2014, 1(5): 465 – 480. (in Chinese)
[15] 冯登国,陈成. 属性密码学研究[J]. 密码学报, 2014, 1(1): 1 – 12.
FENG Deng-guo, CHEN Cheng. Research on attribute-based cryptography[J]. Journal of Cryptologic Research, 2014, 1(1): 1 – 12. (in Chinese)

作者简介

赵志远　男,1989 年生于吉林磐石. 解放军信息工程大学三院博士生. 研究方向为云计算安全、公钥密码学.
E-mail: zzy_taurus@foxmail.com

朱智强　男,1961 年生于吉林长春. 武汉大学博士. 现为解放军信息工程大学三院教授,硕士生导师. 研究方向为云计算、信息安全.