doi:10.11959/j.issn.1000-436x.2015254

# 数据外包环境下一种支持撤销的属性基加密方案

闫玺玺，汤永利
(河南理工大学 计算机科学与技术学院，河南 焦作 454003)

**摘 要**：针对数据外包环境中属性的细粒度撤销，借助于数据外包管理服务器和密钥加密密钥，提出一种间接模式下支持即时撤销的属性基加密方案。首先给出外包环境中支持撤销的属性基加密定义和安全模型，其次给出具体的支持撤销的密文策略——属性基加密方案并对安全性进行证明，最后，与其他方案进行对比，该方案在密文和密钥长度方面都有所减少。另外，方案实现对用户部分属性进行细粒度撤销，支持属性的即时撤销，即使用户错过密钥即时更新的信息，也只需在解密密文前更新自己的密钥，更加贴近于实际环境。
**关键词**：数据外包；属性基加密；属性撤销；访问控制
**中图分类号**：TP309 **文献标识码**：A

# Attribute-based encryption scheme with efficient revocation in data outsourcing systems

YAN Xi-xi, TANG Yong-li
(School of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454003, China)

**Abstract:** In order to support fine-grained attribute revocation in data outsourcing systems, an attribute-based encryption scheme with efficient revocation in indirect revocation model was proposed. The model of ABE supporting attribute revocation was given, and a concrete scheme was constructed which proved its security under the standard model. Compared to the existing related schemes, the size of ciphertext and private/secret key is reduced, and the new scheme achieves fine-grained and immediate attribute revocation which is more suitable for the practical applications.
**Key words:** data outsourcing; attribute-based encryption; attribute revocation; access control

## 1 引言

目前，网络和计算机的发展导致信息和数据呈爆炸性增长，海量数据的外包存储已经成为一种趋势。然而，数据拥有者无法像管理本地文件一样直接管理自己存储在外包服务器中的数据，另外，外包服务提供商存在非法访问用户数据的可能性，因此，在外包环境下如何保护数据的隐私和安全是实践中的一个难题。

为了保护数据的安全，数据拥有者往往会对数据进行加密，将加密后的密文数据外包给数据外包提供商。采用传统的公钥密码机制只能满足一对一的数据分享环境，并且加重数据拥有者和外包服务器的工作负担，不适合数据外包环境下进行数据的细粒度访问控制问题。属性基加密机制(ABE, attribute based encryption)[1]的出现弥补了传统公钥密码体制的缺陷，一方面，数据拥有者仅需根据属性加密消息，只有符合密文属性要求的用户才能解密消息，降低了数据加密开销并保护了用户隐私；另一方面，通过属性的与、或、非和门限操作实现属性灵活的细粒度访问控制策略。ABE 方案可以很好地实现数据外包环境下数据的机密性保证和一对多数据分享模式下细粒度的访问控制，但是由于外包环境中用户会频繁地加入或者离开，如何设计一

**收稿日期**：2015-03-29；**修回日期**：2015-06-12
**基金项目**：国家自然科学基金资助项目（61300216）；河南省科技攻关计划基金资助项目（132102210123）；河南理工大学博士基金资助项目（B2013-043）
**Foundation Items:** The National Natural Science Foundation of China (61300216); The Science and Technology Project of Henan Province (132102210123); The Research Fund for the Doctoral Program of Henan Polytechnic University(B2013-043)

种支持用户或者属性级的 ABE 撤销方案就变得更加重要。

## 2 相关工作

根据撤销属性的影响范围，ABE 属性密钥撤销主要包括 3 种情况：用户撤销、用户部分属性撤销和系统属性撤销。用户撤销是针对用户所有属性的撤销，不影响其余未撤销的用户；用户部分属性撤销是针对用户属性集合中包含的某些属性的撤销，撤销后此用户失去该属性对应的权限，不影响具备该属性其余用户的权限；系统属性是针对具有该属性的所有用户进行撤销。

根据撤销执行者的不同，当前 ABE 撤销机制的研究工作主要分为直接撤销和间接撤销 2 类。直接撤销由发送方执行，发送方在信息加密阶段直接加入撤销用户的列表信息，从而实现属性密钥的撤销。文献[2]首次提出基于密文策略属性基加密（CP-ABE, ciphertext-policy ABE）的直接撤销思想，把用户标识作为一种属性，利用“非”的用户标识与密文进行关联，当用户被撤销时，他的用户标识就成为“非”，将无法解密，从而实现撤销用户或者系统属性，但是该方案增加了密文和用户密钥长度。文献[3]结合广播加密思想，实现基于密钥策略属性基加密和密文策略属性基加密的属性直接撤销，该方案降低了撤销开销，撤销的用户不会影响其余未撤销用户的权限。直接撤销方法将唯一确定用户身份的属性信息作为用户标识，仅支持整个用户的撤销，无法解决用户部分属性撤销的问题，并且用户密钥和密文长度有所增加。文献[4]基于合数阶双线性群实现了属性直接撤销的 CP-ABE 方案，该方案公钥参数与用户数量线性相关，容易造成公钥参数过长。文献[5]将属性撤销列表嵌入到密文中，实现支持属性直接撤销的 CP-ABE 方案的一般构造。

间接撤销由授权机构执行，授权机构周期性的更新未撤销用户的密钥，只有未撤销的用户才能更新密钥，通过新密钥解密新密文，而撤销用户将无法收到更新而导致密钥无效。文献[6]最早提出 ABE 属性撤销方法，通过给每个属性设置一个有效期，授权机构周期性地释放属性的最新版本，通过撤销用户某个属性的最新版本来实现对用户属性的撤销。文献[7]用属性的终止日期取代了有效期，来限制密钥的使用时间。这 2 种方案都不能满足实际应用需求，密钥更新过程中，授权机构密钥更新的工作量与用户数据线性相关，另外，都不支持属性的即时撤销。文献[8]采用二叉树提出可撤销的 KP-ABE 机制，支持用户撤销，并不支持属性的即时撤销。为了实现属性的即时撤销，Ibraimi 等[9]提出引入第三方扮演仲裁者，将用户密钥分别由仲裁者和用户持有，实现 CP-ABE 撤销方案。Yu 等[10]引入半可信的代理服务器，基于代理重加密技术实现可撤销的 KP-ABE 方案。这 2 种方案实现了属性的即时撤销，减轻了授权机构的工作量，但要求第三方必须保持在线。文献[11]中将数据文件分成许多小片段进行存储，当撤销事件发生时，数据拥有者对部分片段进行重加密，实现间接的用户撤销。Hur 等[12]基于二叉树，通过向合法用户分发一个 KEK 二叉树，提出一个支持完全细粒度属性撤销的 CP-ABE 方案，该方案支持属性的即时撤销，但是密钥维护代价高，且无法抵抗合谋攻击。Xie 等[13]对 Hur 的方案进行了优化，缩小了密文和密钥的尺寸，并减轻了密钥更新阶段的计算量，但是这 2 种方案都是基于一般的群假设安全问题。文献[14]利用代理重加密技术将属性撤销工作交给云服务商完成，减轻了授权机构的工作量。

本文针对已有支持属性撤销方案的不足，基于 Ibraimi 等[15]提出的 CP-ABE 方案，并借鉴 Hur 等[12]的属性撤销方案，提出一种支持完全细粒度属性撤销的 CP-ABE 方案，实现属性的即时撤销，并且无需第三方保持在线。本文将给出具体的支持属性撤销的 CP-ABE 方案构造并证明其满足选择明文攻击安全。

## 3 算法定义和安全模型

### 3.1 算法定义

支持属性撤销的 CP-ABE 方案由系统初始化 $Setup(k)$、密钥生成 $KeyGen(mk,\omega,U)$、密钥加密密钥生成 $KEKGen(U)$、加密算法 $Encrypt(m,\tau,pk)$、密文重加密算法 $ReEncrypt(c_\tau,U)$、属性群密钥解密算法 $UkDecrypt(Hdr,KEK)$ 和解密算法 $Decrypt(c_\tau',sk_\omega,pk)$ 构成。

1) $Setup(k)$：系统初始化算法由属性权威中心运行，输入安全参数 $k$，输出主密钥 $mk$ 和主公钥 $pk$。

2) $KeyGen(mk,\omega,U)$：密钥生成算法由属性权威中心运行，输入主密钥 $mk$、属性集合 $\omega$、用户

集合 $U$，输出用户的属性私钥 $sk_\omega$。

3) $KEKGen(U)$：密钥加密密钥算法由数据管理服务器运行，输入用户集，输出每个用户的二叉树 $KEK$。

4) $Encrypt(m,\tau,pk)$：加密算法由加密者运行，算法以系统公钥 $pk$、消息 $m$、访问树 $\tau$ 为输入，输出密文 $c_\tau$。

5) $ReEncrypt(c_\tau,U)$：重加密算法由数据管理服务器运行，输入密文 $c_\tau$ 和属性用户群 $U$，输出重加密后的密文 $c_\tau'$ 和头信息 $Hdr$。

6) $UkDecrypt(Hdr,KEK)$：属性用户群密钥解密算法由解密者运行，输入头信息 $Hdr$，用户的二叉树 $KEK$，只要用户未被撤销出属性群，并且被授予权限，输出属性用户群密钥 $k_j$；否则输出 $\perp$。

7) $Decrypt(c_\tau',sk_\omega,pk)$:解密算法由解密者运行，算法输入用户的私钥 $sk_\omega$、密文 $c_\tau'$、系统公钥 $pk$，输出消息明文 $m$。

### 3.2 安全模型

下面将给出支持撤销的CP-ABE方案在选择属性和明文攻击下的不可区分性（IND-sAtt-CPA, indistinguishability against selective attribute and chosen-plaintext attack）下的安全模型。这种模型中，攻击者需要提供挑战的访问树在系统进行设置之前，例如，攻击者选择访问树为 $\tau^*=(A\wedge B)\vee C$，在进行阶段一时挑战者申请属性私钥时，选择的属性集合 $w$ 必须满足 $A,B,C\notin\omega$。

攻击者$\mathcal{A}$和挑战者$\mathcal{B}$之间的具体攻击游戏如下。

准备阶段：攻击者$\mathcal{A}$选择访问树 $\tau^*$，发送给挑战者$\mathcal{B}$。

系统设置：挑战者$\mathcal{B}$运行 $Setup(k)$ 算法生成主密钥 $mk$ 和主公钥 $pk$，将公钥 $pk$ 发送给攻击者$\mathcal{A}$，自己保存主密钥 $mk$。

阶段 1：攻击者$\mathcal{A}$任意选择属性集$w=\{a_j\,|\,a_j\notin\tau^*\}$，并向挑战者$\mathcal{B}$发出属性私钥请求，挑战者$\mathcal{B}$运行 $KeyGen(mk,\omega,U)$ 得到用户的属性私钥 $sk_\omega$，运行 $KEKGen(U)$ 算法得到用户的二叉树 $KEK$，将 $sk_\omega$ 和 $KEK$ 返回给攻击者$\mathcal{A}$。

挑战：攻击者$\mathcal{A}$向挑战者$\mathcal{B}$发送 2 个等长信息 $m_0$ 和 $m_1$，挑战者$\mathcal{B}$执行公平的掷硬币游戏选取 $b\in\{0,1\}$，运行算法 $Encrypt(m_b,\tau^*,pk)$ 和 $ReEncrypt(c_\tau,U)$，将得到的挑战密文 $c_{\tau^*}$ 和头信息 $Hdr$ 返回给攻击者$\mathcal{A}$。

阶段 2：同阶段 1，攻击者可以继续向挑战者进行询问。

猜测：攻击者猜测 $b'\in\{0,1\}$，如果 $b'=b$，则说明攻击者成功。攻击者进行游戏获得成功的优势为 $Adv_{\text{CP-ABE}}^{\text{IND-CPA}}(A)=\left|\Pr[b'=b]-\frac{1}{2}\right|$，其中的概率取决于随机参数的概率分布和算法的内部随机掷币。

如果没有概率多项式时间攻击者能够以不可忽略的优势赢得 IND-sAtt-CPA 的游戏，则称支持撤销的 CP-ABE 方案是 IND-sAtt-CPA 安全的。

## 4 数据外包环境下一种支持撤销的属性基加密方案

### 4.1 基本思想

本文基于 Ibraimi 等[15]提出的 CP-ABE 方案，访问结构为由与、或节点组成的 $l$ 叉树，加密算法采用模加机制赋值给与节点的子女节点，直接将秘密值赋值给或节点的子女节点，用户的私钥与一个随机数相关，防止用户串谋，实现属性级访问控制。同时，借鉴 Hur 等[12]的属性撤销方案中 $KEK$ 二叉树生成密钥加密密钥对内容密钥密文进行代理重加密，执行用户级访问控制，通过更新密钥加密密钥实现对用户属性的撤销。

### 4.2 数据外包系统架构

一个基本的数据外包系统包括可信属性权威机构、数据管理服务器、数据服务器、数据拥有者、数据使用者 5 个实体。

1) 可信属性权威机构（TA, trusted authority）：主要负责为系统生成主公钥和主密钥，为用户生成、分发、撤销和更新属性私钥，是被用户完全可信的。

2) 数据管理服务器(DM, data service manager)：提供数据外包管理服务，控制外部用户对数据进行访问，并提供相关的服务，数据管理服务器获得的只是数据密文，当被属性权威告知需要更新密文时，其对数据密文进行重加密操作。

3) 数据服务器（DS, data service）：主要用于存储外包数据密文。

4) 数据拥有者(DO, data owner)：数据的原始拥有者，为数据定义访问控制策略，并对数据进行加

密，将密文外包给数据管理服务器。

5) 数据使用者(DU, data user)：数据的访问者，只有当其属性满足密文相对应的访问策略，以及拥有相应的权限时，才可以解密出明文数据。

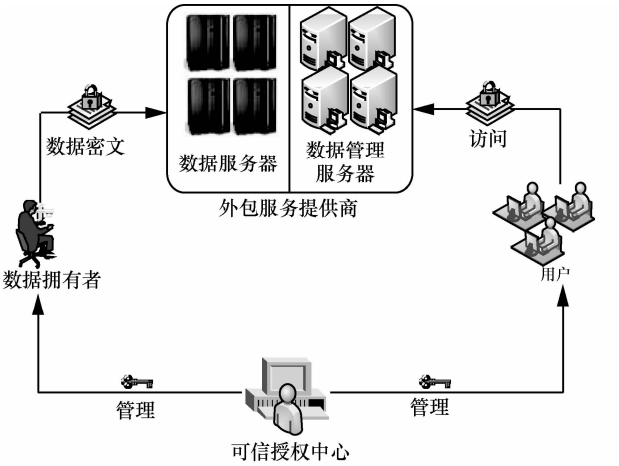


图 1　数据外包系统架构

### 4.3　方案构造

1) 系统初始化

由可信认证机构 TA 执行 $Setup(k)$ 算法，具体过程如下。

①选取一个双线性群 $G_0$，其阶为 $p$，生成元为 $g$，并且选取双线性映射 $e:G_0\times G_0\to G_1$。

②随机生成 $\alpha,t_1,t_2,\cdots,t_n\in Z_p^*$，计算 $T_j=g^{t_j}(1\leqslant j\leqslant n)$，$y=e(g,g)^{\alpha}$。

③公布公钥 $pk=\left(e,g,y,T_j\left(1\leqslant j\leqslant n\right)\right)$，可信认证机构保存主密钥 $mk=\left(\alpha,t_j\left(1\leqslant j\leqslant n\right)\right)$。

2) 密钥生成

用户集中的向认证机构申请注册，由认证机构对用户的属性进行验证，并通过 $KeyGen(mk,\omega,U)$ 算法为每个用户生成相应的私钥。

①随机为每个用户选取唯一的 $r\in Z_p^*$，计算 $d_0=g^{\alpha-r}$。

②对每个属性 $a_j\in\omega$，计算 $d_j=g^{rt_j^{-1}}$。

③发送相应的私钥给每个用户 $sk_{\omega}=(d_0,\forall a_j\in\omega:d_j)$。

执行完上述过程后，认证机构将每个属性 $a_j\in\omega$ 对应的属性用户群 $U_j$ 发送给数据管理服务器。例如，用户 $u_1$ 拥有的属性为 $\{a_{j_1},a_{j_2},a_{j_3}\}$，用户 $u_2$ 拥有的属性为 $\{a_{j_1},a_{j_2}\}$，用户 $u_3$ 拥有的属性为 $\{a_{j_2},a_{j_3}\}$，则属性 $a_{j_1}$ 所对应的属性用户群 $U_1=\{u_1,u_2\}$，属性 $a_{j_2}$ 所对应的属性用户群 $U_2=\{u_1,u_2,u_3\}$，属性 $a_{j_3}$ 所对应的属性用户群 $U_3=\{u_1,u_3\}$。

3) 密钥加密密钥生成

数据管理服务器收到属性用户群后，为 $U$ 中每一个用户生成相应的 *KEK* 二叉树，此二叉树用来为每一个用户分发属性群密钥，具体构造过程如下。

①将每个成员都分布到二叉树的叶子节点上，为树中所有的节点 $v_j$ 都随机生成一个密钥 *KEK*，记为 $KEK_j$。

②每个叶子节点到根节点所经过的节点称为路径节点，路径节点所代表的密钥集合即为每个用户 $u_t\in U$ 的专属路径密钥，记为 $PK_t$。如用户 $u_2$ 所存储的路径密钥为 $PK_2=\{KEK_9,KEK_4,KEK_2,KEK_1\}$。

③对于每个 $U_j$，存在一个相应的最小覆盖元，它可以覆盖所有和 $U_j$ 中的成员对应的叶子节点，记 $U_j$ 的最小覆盖元为 $KEK(U_j)$。假定属性用户群 $U_j=\{u_1,u_2,u_3,u_4,u_7,u_8\}$，因为节点 $v_2$ 和 $v_7$ 可以覆盖 $U_j$ 中的所有成员，所以属性用户群 $U_j$ 的最小覆盖元是 $KEK(U_j)=\{KEK_2,KEK_7\}$。

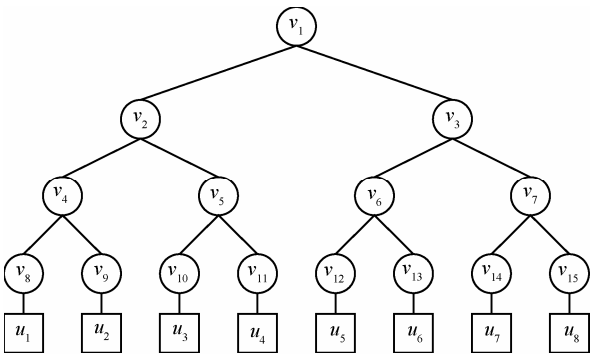


图 2　属性群密钥分发所对应的 *KEK* 二叉树

4) 数据加密

数据拥有者 DO 基于属性域指定一个访问树 $\tau$，通过 $Encrypt(m,\tau,pk)$ 算法对消息 $m$ 进行加密，具体算法如下。

①第 1 层加密：选择随机数 $s\in Z_p^*$，计算 $c_0=g^s$，$c_1=my^s=me(g,g)^{\alpha s}$。

②第 2 层加密：设置访问树 $\tau$ 根节点的值为待共享的值 $s$，将根节点置为已分配，其所有的孩子节点标记为未分配。对每个未分配的非叶子节点执行以下递归算法。

若标识符号为∧，且它的孩子节点标记为未分配，采用模加机制赋值给孩子节点。对每个孩子节点赋予一个随机数 $s_i(1 \leqslant s_i \leqslant p-1)$，最后一个孩子节点的值为 $s_t = s - \sum_{i=1}^{t-1} s_i \bmod p$，标记这些节点为已分配；若标识符号为∨，且它的孩子节点标记为未分配，设置其孩子节点为 $s$，并标记节点为已分配。

③对每个叶子节点 $a_{j,i} \in Y$（$Y$ 表示访问树 $\tau$ 的叶子节点的集合，$Y_\wedge$ 表示∧节点的 $t-1$ 个孩子叶子节点集合，$Y_\vee$ 表示∨节点的孩子叶子节点，$i$ 表示访问树中叶子节点所对应的索引值），计算 $c_{j,i} = T_j^{s_i}$。

④返回密文 $c_\tau = (\tau, c_0, c_1, \forall a_{j,i} \in Y : c_{j,i})$。

数据拥有者将密文 $c_\tau$ 安全外包给数据管理服务器。

图 3 给出了访问树 $\tau = (T_1 \wedge T_2) \vee (T_1 \vee T_2)$ 的秘密分配，具体过程如下。

①根节点（∨节点）设置为 $s$，对第 2 层非叶子节点进行秘密分配，则∧节点值为 $s$，∨节点值为 $s$。

②第 2 层∧节点值为 $s$，采用模加机制对 $s$ 进行分配，设其一个孩子节点值为 $s_1$，则另一个孩子节点值为 $s_2 = s - s_1$。

③第 2 层∨节点值为 $s$，则其孩子节点值分别为 $s_3 = s$，$s_4 = s$。

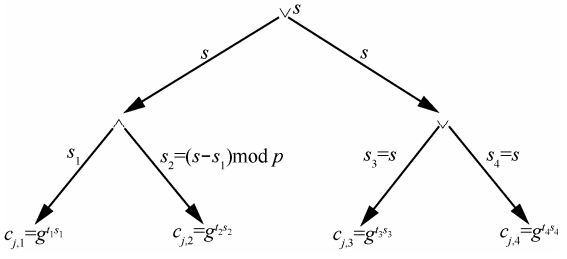


图 3 访问树 $\tau = (T_1 \wedge T_2) \vee (T_1 \vee T_2)$ 的秘密分配

5) 数据重加密

数据管理服务器收到数据密文后，基于属性用户群对数据密文进行重加密，执行用户级的访问控制。采用数据重加密算法 $ReEncrypt(c_\tau, U)$，具体过程如下。

①对于任意的 $U_j \in U$，随机选择 $k_j \in Z_p^*$，计算 $\forall a_{j,i} \in Y : c_{j,i}' = (c_{j,i})^{k_j} = (T_j^{s_i})^{k_j}$，则密文 $c_\tau' = (\tau, c_0, c_1, \forall a_{j,i} \in Y : c_{j,i}')$。

②生成头信息 $Hdr = (\forall a_{j,i} \in Y : c_{k_j} = \{E_K(k_j)\}_{K \in KEK(U_j)})$，其中 $E_K(x)$ 表示采用密钥 $K$ 对明文 $x$ 进行对称加密，最简单的实现方法是采用一种分组密码 $E_K : \{0,1\}^k \to \{0,1\}^k$，$k$ 为密钥 $K$ 的长度，通过这种方法将属性群密钥发送给有效的用户。

数据管理服务器收到用户发出的数据使用请求后，将 $(Hdr, c_\tau')$ 发送给用户。

6) 数据解密

数据解密阶段包括属性群密钥解密和消息解密 2 部分。

**属性群密钥解密**

用户收到后，只要未被撤销出属性群，并且被授予权限，可以随时从头信息 $Hdr$ 中解密出属性群密钥，即使用户未能实时更新自己的密钥。

①用户首先解密属性 $\omega$ 中所有属性对应的属性群密钥，如果用户 $u_t \in U_j$，那么他将借助于自己所存储的路径密钥来解密属性群密钥 $k_j$：$k_j = \{D_K(c_{k_j})\}_{K \in PK_t}$。如属性用户群 $U_j = \{u_1, u_2, u_3, u_4, u_7, u_8\}$，$KEK(U_j) = \{KEK_2, KEK_7\}$，用户 $u_3$ 的路径密钥为 $PK_3 = \{KEK_{10}, KEK_5, KEK_2, KEK_1\}$，则用户可以通过 $KEK_2 \in PK_3$ 来解密 $k_j$。

②用户通过属性群密钥更新自己的私钥：$sk_\omega = \left(d_0, \forall a_j \in \omega : d_j' = (d_j)^{k_j^{-1}}\right)$。

**信息解密**

用户输入自己的属性列表 $\omega' = (a_1, a_2, \cdots, a_k)$，执行 $Decrypt(c_\tau', sk_\omega, pk)$ 算法，计算 $m = \dfrac{c_1}{e(c_0, d_0) \prod\limits_{a_j \in \omega'} e(c_{j,i}', d_j')}$。

①当 $\omega' \subseteq \omega$ 时，对每个属性 $a_j \in \omega'$，

$$\begin{aligned}
&\prod_{a_j \in \omega'} e\left(c_{j,i}', d_j'\right) \\
&= \prod_{a_j \in \omega'} e\left(\left(T_j^{s_i}\right)^{k_j}, \left(d_j\right)^{k_j^{-1}}\right) \\
&= \prod_{a_j \in \omega'} e\left(\left(g^{t_j s_i}\right)^{k_j}, \left(g^{r t_j^{-1}}\right)^{k_j^{-1}}\right) \\
&= e(g, g)^{rs}
\end{aligned}$$

②计算

$$
\begin{aligned}
&e(c_0,d_0)e(g,g)^{rs}\\
&=e\left(g^{s},g^{\alpha-r}\right)e(g,g)^{rs}\\
&=e(g,g)^{\alpha s}e(g,g)^{-rs}e(g,g)^{rs}\\
&=e(g,g)^{\alpha s}
\end{aligned}
$$

③计算

$$
m'=\frac{c_1}{e(g,g)^{\alpha s}}=\frac{me(g,g)^{\alpha s}}{e(g,g)^{\alpha s}}=m
$$

7) 密钥更新

当某个用户属性发生变化时，即需要加入或离开某个属性群时，这就意味着属性群密钥需要更新。当可信认证机构收到用户属性发生变化时，更新属性用户群，并将更新的用户名单通知给数据管理服务器，由数据管理服务器执行属性群密钥更新。

假设属性用户群 $U_f$ 中成员发生变化，密钥更新过程如下。

① 数据管理服务器随机选择 $s'\in Z_p^*$ 和 $k_f'\in Z_p^*\left(k_f'\neq k_f\right)$，执行重加密操作

$$
c_0'=c_0g^{s'}=g^{s+s'}
$$

$$
c_1'=c_1y^{s'}=me(g,g)^{\alpha s}e(g,g)^{\alpha s'}=me(g,g)^{\alpha(s+s')}
$$

$$
c_{f,i}=\left(T_f^{s_i'}\right)^{k_{f'}},\ \forall a_{j,i}\in Y\setminus\{f\}:c_{j,i}'=\left(T_j^{s_i'}\right)^{k_j}
$$

其中，$\begin{cases}s_i'=s_i+s' & a_{j,i}\in Y\setminus Y_\wedge\\ s_i'=s_i & a_{j,i}\in Y_\wedge\cup Y_\vee\end{cases}$，则新的密文

$$
c_\tau'=\left(\tau,c_0',c_1',c_{f,i},\forall a_{j,i}\in Y\setminus\{f\}:c_{j,i}'\right)
$$

②更新完密文后，数据管理服务器为 $U_f$ 选择一个新的最小覆盖元，并且生成新的头信息

$$
Hdr=\begin{pmatrix}\left\{E_K\left(k_f'\right)\right\}_{K\in KEK(U_f)},\\ \forall a_{j,i}\in Y\setminus\{f\}:c_{k_j}=\left\{E_K\left(k_j\right)\right\}_{K\in KEK(U_j)}\end{pmatrix}
$$

当用户需要访问数据时，接收到的是更新后的 $\left(Hdr,c_\tau'\right)$。

### 4.4 正确性检验

当撤销事件发生时，用户收到 $\left(Hdr,c_\tau'\right)$，首先进行属性群密钥解密从头信息 $Hdr$ 中解密出属性群密钥，并更新自己的私钥

$$
sk_\omega=\left(d_0,d_f'=\left(d_f\right)^{k_f^{-1}},\forall a_j\in\omega\setminus\{f\},:d_j'=\left(d_j\right)^{k_j^{-1}}\right)
$$

其次进行信息解密计算，当 $\omega'\subseteq\omega$ 时，对每个属性 $a_j\in\omega'$，有

$$
\begin{aligned}
&\prod_{a_j\in\omega'}e\left(c_{j,i}',d_j'\right)\\
&=\prod_{a_j\in\omega'\setminus\{f\}}e\left(\left(T_j^{s_i'}\right)^{k_j},\left(d_j\right)^{k_j^{-1}}\right)e\left(\left(T_f^{s_i'}\right)^{k_f},\left(d_f\right)^{k_f^{-1}}\right)\\
&=\prod_{a_j\in\omega'\setminus\{f\}}e\left(\left(g^{t_js_i'}\right)^{k_j},\left(g^{rt_j^{-1}}\right)^{k_j^{-1}}\right)e\left(\left(g^{t_fs_i'}\right)^{k_f},\left(g^{rt_f^{-1}}\right)^{k_f^{-1}}\right)\\
&=\prod_{a_j\in\omega'}e(g,g)^{rs_i'}\\
&=e(g,g)^{r(s+s')}\\
&e\left(c_0',d_0\right)e(g,g)^{r(s+s')}\\
&=e\left(g^{s+s'},g^{\alpha-r}\right)e(g,g)^{r(s+s')}\\
&=e(g,g)^{\alpha(s+s')}e(g,g)^{-r(s+s')}e(g,g)^{r(s+s')}\\
&=e(g,g)^{\alpha(s+s')}
\end{aligned}
$$

则 $m'=\frac{c_1'}{e(g,g)^{\alpha(s+s')}}=\frac{me(g,g)^{\alpha(s+s')}}{e(g,g)^{\alpha(s+s')}}=m$。

### 4.5 安全性分析

本方案基于判定双线性 Diffie-Hellman 问题（DBDH, decisional bilinear Diffie-Hellman problem）：给定 $\left(g,g^a,g^b,g^c,Z\right)$，其中 $a,b,c,\theta\in Z_p^*$，$Z=e(g,g)^\theta$，计算 $e(g,g)^{abc}$，判断 $Z=e(g,g)^{abc}$ 是否成立。一个概率性多项式时间算法 $B$ 能够以优势 $\varepsilon$ 求解 DBDH 问题，当且仅当满足

$$
\begin{aligned}
&\Pr\left[B\left(g,g^a,g^b,g^c,e(g,g)^{abc}\right)=1\right]\\
&-\Pr\left[B\left(g,g^a,g^b,g^c,e(g,g)^{\theta}\right)=1\right]\geqslant\varepsilon
\end{aligned}
$$

**定理 1**　假设 DBDH 成立，如果没有敌手可以在多项式时间内选择访问树 $\tau^*$ 下攻破支持撤销的 CP-ABE 方案，那么该方案是 IND-sAtt-CPA 安全的。

**证明**　如果存在一个攻击者 $\mathcal{A}$ 可以攻破本方案，则存在一个算法 $B$ 可以攻破 DBDH 问题，即输入 $\left(g,g^a,g^b,g^c,Z_\mu=e(g,g)^\theta\right)$，算法 $B$ 决定等式 $Z_\mu=e(g,g)^{abc}$ 是否成立。

攻击者 $\mathcal{A}$ 和挑战者 $\mathcal{B}$ 按照如下操作，执行 IND-sAtt-CPA 游戏。

初始化：攻击者$\mathcal{A}$选择访问树$\tau^*$，发送给挑战者$\mathcal{B}$。

系统设置：挑战者$\mathcal{B}$运行$Setup(k)$算法如下。

①选择随机数$x'\in Z_p^*$，设置$e(g,g)^{\alpha}=e(g,g)^{ab}e(g,g)^{x'}$使$\alpha=ab+x'$。

②对每一个属性$a_j\in\Omega(1\leqslant j\leqslant n)$，选择随机数$s_j\in Z_p^*$，如果$a_j\notin\tau^*$，则设$T_j=B^{s_j^{-1}}$，因此$t_j=bs_j^{-1}$；如果$a_j\in\tau^*$，则设$T_j=g^{s_j}$，因此$t_j=s_j$。

③计算$y=e(g,g)^{\alpha}$，则公钥$pk=(e,g,y,T_j(1\leqslant j\leqslant n))$，$mk=\left(\alpha,t_j(1\leqslant j\leqslant n)\right)$。

挑战者$\mathcal{B}$将公钥$pk$发送给攻击者$\mathcal{A}$，自己保存主密钥$mk$。

阶段1：攻击者$\mathcal{A}$任意选择属性集$w=\left\{a_j\middle|a_j\notin\tau^*\right\}$，并向挑战者$\mathcal{B}$发出属性私钥询问请求。

①挑战者$\mathcal{B}$随机选择$r'\in Z_p^*$，计算$d_0=g^{x'-r'b}$，因为$x'=\alpha-ab$，所以$d_0=g^{x'-r'b}=g^{\alpha-ab-r'b}=g^{\alpha-(a+r')b}$，由此可以得知$r=ab+r'b$。

②对每个属性$a_j\in\omega$，因为$r=ab+r'b$，$t_j=bs_j^{-1}$，计算$d_j=g^{(ab+r'b)b^{-1}s_j}=g^{as_j}g^{r's_j}=A^{s_j}g^{r's_j}$；

③运行$KEKGen(U)$算法得到用户的二叉树$KEK$；

挑战者$\mathcal{B}$将属性私钥$sk_{\omega}=\left(d_0,\forall a_j\in\omega:d_j\right)$和用户的二叉树$KEK$返回给攻击者$\mathcal{A}$。

挑战：攻击者$\mathcal{A}$向挑战者$\mathcal{B}$发送2个等长信息$m_0$和$m_1$，挑战者$\mathcal{B}$执行公平的掷硬币游戏选取$b\in\{0,1\}$，进行以下操作。

①第一层加密：计算$c_0=g^c$，

$$c_1=m_be(g,g)^{\alpha c}=m_be(g,g)^{(ab+x')c}$$
$$=m_be(g,g)^{abc}e(g,g)^{x'c}=m_bZe\left(g^c,g^{x'}\right)$$

②第二层加密：设访问树$\tau^*$根节点的值为$g^c$，标记所有的孩子节点为未分配，根节点为已分配，对每一个未分配的非叶子节点执行以下操作。

若标识符号为$\wedge$，且它的孩子节点标记为未分配。除最后一个孩子外，对每个孩子节点选择一个随机数$h_i(1\leqslant h_i\leqslant p-1)$，并计算$g^{h_i}$，则最后一个孩子节点的值为$g^{h_t}=\dfrac{g^c}{\sum_{i=1}^{t-1}g^{h_i}}$，标记这些节点为已分配；

若标识符号为$\vee$，且它的孩子节点标记为未分配，为孩子节点选择$g^c$，并标记节点为已分配。

③对每个叶子节点$a_{j,i}\in\tau^*$，计算$c_{j,i}=g^{h_is_j}$。

④对于任意的$U_j\in U$，随机选择$k_j\in Z_p^*$，计算$\forall a_{j,i}\in\tau^*:c_{j,i}'=\left(c_{j,i}\right)^{k_j}=\left(g^{h_is_j}\right)^{k_j}$，则密文$c_{\tau^*}'=\left(\tau^*,c_0,c_1,\forall a_{j,i}\in\tau^*:c_{j,i}'\right)$。

⑤生成头信息

$$Hdr=\left(\forall a_{j,i}\in Y:c_{k_j}=\left\{E_K\left(k_j\right)\right\}_{K\in KEK(U_j)}\right)。$$

挑战者$\mathcal{B}$将密文$c_{\tau^*}'$和头信息$Hdr$返回给攻击者$\mathcal{A}$。

阶段2：同阶段1，攻击者$\mathcal{A}$可以继续向挑战者$\mathcal{B}$进行询问。

猜测：攻击者$\mathcal{A}$输出猜测$b'\in\{0,1\}$。

如果$b'=b$，则挑战者输出1，表示DBDH成立，$Z=e(g,g)^{abc}$。否则，输出0，表明$Z=e(g,g)^{\theta}$。

当$Z=e(g,g)^{abc}$，攻击者获得的是有效的密文，攻击者的优势为$\Pr\left[b'=b\middle|Z=e(g,g)^{abc}\right]=\dfrac{1}{2}+\varepsilon$。

当$Z=e(g,g)^{\theta}$，攻击者获得的密文是随机的，并不能获得明文的任何信息，$\Pr\left[b'\neq b\middle|Z=e(g,g)^{\theta}\right]=\dfrac{1}{2}$。

因此，

$$\Pr\left[B\left(g,g^a,g^b,g^c,e(g,g)^{abc}\right)=1\right]$$
$$-\Pr\left[B\left(g,g^a,g^b,g^c,e(g,g)^{\theta}\right)=1\right]\geqslant\varepsilon$$

成立，即假定攻击者$\mathcal{A}$能够以优势$\varepsilon$求解DBDH。

综上分析，假设DBDH成立，如果没有敌手可以在多项式时间内选择访问树$\tau^*$下攻破支持撤销的CP-ABE方案，那么该方案是IND-sAtt-CPA安全的。

## 5 性能分析

ABE撤销机制的研究工作大部分采用间接撤销模式，因此本文方案仅对间接撤销方案中几个方案从密文长度、属性私钥大小、系统公钥大小、解密过程的计算量、属性撤销机制是否满足属性即时撤销、对授权机构以及第三方是否要求保持在线几个方面进行比较，具体比较结果如表1所示。其中，

**表 1**　　**各种间接撤销机制效率对比**

| 方案 | 密文长度 | 私钥长度 | 公钥长度 | 解密代价 | 属性即时撤销 | 在线要求 | 安全性假设 | 安全模型 |
|---|---|---|---|---|---|---|---|---|
| 文献[7] | $2(t+1)L_0+L_1$ | $(2k+1)L_0+L_{kek}$ | $L_0+L_1$ | $2k+1$ | × | 授权方 | group model | CPA 安全 |
| 文献[10] | $(n+1)L_0+L_1$ | $(2n+1)L_0+L_k$ | $(3n+1)L_0+L_1$ | $n+1$ | √ | 授权方，第三方 | DBDH | CPA 安全 |
| 文献[12] | $(2t+1)L_0+L_1$ | $(2k+1)L_0+(\log m)L_{kek}$ | $2L_0+L_1$ | $2k+1$ | √ | 授权方 | group model | 无安全模型 |
| 文献[13] | $(t+1)L_0+L_1$ | $(k+1)L_0+(\log m)L_{kek}$ | $L_0+L_1$ | $k+1$ | √ | 授权方 | group model | 无安全模型 |
| 文献[14] | $(2t+1)L_0+L_1$ | $(k+2)L_0$ | $(2n+2)L_0+L_1$ | $2k+1$ | √ | 第三方 | $q$-parallel BDHE | CPA 安全 |
| 本文方案 | $(t+1)L_0+L_1$ | $(k+1)L_0+(\log m)L_{kek}$ | $(n+1)L_0+L_1$ | $k+1$ | √ | 授权方 | DBDH | CPA 安全 |

$L_0$ 和 $L_1$ 分别表示群 $G_0$ 和群 $G_1$ 中一个元素的比特长度，$L_k$ 表示用户的属性私钥对应的属性集的比特大小，$L_{kek}$ 表示密钥加密密钥的比特长度，$n$ 表示属性域中的属性总个数，$t$ 表示加密时访问策略中出现的属性的个数，$k$ 表示用户所拥有的属性个数，$m$ 表示系统中用户的最大数目，解密代价指的是双线性对计算次数。

从表 1 可以看出，本文所提出的方案在密文长度、属性私钥大小、系统公钥大小，解密过程的计算量方面性能都是最优的。文献[7]无法支持属性即时撤销。文献[10]中密文长度、属性私钥大小和系统公钥大小以及解密代价均和属性域中的属性总个数成线性增长，且要求授权方和代理第三方同时在线，不适用于数据外包环境。文献[12]仅仅满足一般群模型下保证安全，并没有给出具体的安全模型，此外，其解密代价与用户所拥有的属性个数相关联。文献[13]在密文长度、属性私钥大小、系统公钥大小，解密过程的计算量方面性能都是最优的，但是和文献[12]一样，仅仅满足一般群模型下保证安全，并没有给出具体的安全模型。文献[14]将所有的属性认证工作交给第三方工作，对第三方有更高的安全性需求，并不是很实用。本文方案在密文长度、属性私钥大小、解密过程的计算量方面性能都达到了优化，同时本方案支持属性的即时撤销，并不需要第三方时刻保持在线。此外，本文方案满足 DBDH 假设下选择明文攻击安全（CPA, chosen-plaintext attack）。因此，结合工作效率和安全性能来看，本文所提方案与其他几个间接撤销方案相比是更优的。

## 6　结束语

针对数据外包环境下数据的机密性和细粒度的访问控制需求，本文提出一种支持属性撤销的 CP-ABE 方案，并给出其安全性证明。该方案具备以下优点：1）保证数据机密性，只有当用户的属性集满足访问控制策略时才能正常解密；2）支持细粒度属性撤销，实现对用户的某一个属性进行单独的撤销；3）数据拥有者只需关注属性访问策略，由数据管理服务端实施用户级访问控制；4）无状态接收问题，即使用户错过密钥即时更新的信息，只需在解密密文前更新自己的密钥。本文方案中密钥维护代价方面还需要继续改进，如何降低密钥更新的工作量以及如何实现属性的直接撤销将是下一步工作所要考虑的问题。

## 参考文献：


[1] SAHAI A, WATERS B. Fuzzy identity-based encryption[A]. Advances in Cryptology - EUROCRYPT 2005[C]. Berlin, Heidelberg, Springer - Verlag, 2005. 457-473.

[2] OSTROVSKY R, SAHAI A, WATERS B. Attribute-based encryption with non-monotonic access structures[A]. Proc of the ACM Conf on Computer and Communications Security[C]. New York, 2007. 195-203.

[3] ATTAPADUNG N, IMAI H. Conjunctive broadcast and attribute-based encryption[A]. Proc of the Pairing-Based Cryptography-Pairing 2009[C]. Berlin, Heidelberg: Springer-Verlag, 2009. 248-265.

[4] 王鹏翩,冯登国,张立武.一种支持完全细粒度属性撤销的 CP-ABE 方案[J].软件学报, 2012, 23(10): 2805-2816.
WANG P P, FENG D G, ZHANG L W. CP-ABE scheme supporting fully fine-grained attribute revocation[J]. Journal of Software, 2012, 23(10): 2805-2816.

[5] WU Q X. A generic construction of ciphertext-policy attribute-based encryption supporting attribute revocation [J]. In: China communications, 2014, 11(1):93-100.

[6] PIRRETTI M, TRAYNOR P, MCDANIEL P, *et al*. Secure attribute-based systems[A]. Proc of the ACM Conf. on Computer and Communications Security[C]. New York, 2006. 99-111.

[7] BETHENCOURT J, SAHAI A, WATERS B. Ciphertext-policy attrib-

ute-based encryption[A]. Proc of the 2007 IEEE Symp on Security and Privacy[C]. Washington, IEEE Computer Society, 2007. 322-334.

[8] BOLDYREVA A, GOYAL V, KUMAR V. Identity-based encryption with efficient revocation[A]. Proc of the ACM Conf on Computer and Communications Security[C]. New York, 2008. 417-426.

[9] IBRAIMI L, PETKOVIC M, NIKOYA S, *et al*. Mediated ciphertext-policy attribute-based encryption and its application[A]. Proc of the 10th Int'l Workshop on Information Security Applications-WISA 2009[C]. LNCS 5932, Berlin, Heidelberg: Springer-Verlag, 2009. 310-322.

[10] YU S C, WANG C, REN K, *et al*. Attribute based data sharing with attribute revocation[A]. Proc of the ASIAN ACM Conf on Computer and Communications Security (ASIACCS 2010)[C]. New York, ACM Press, 2010. 262-270.

[11] CHENG Y, WANG Z Y, MA J, *et al*. Efficient revocation in ciphertext-policy attribute-based encryption based cryptographic cloud storage [J]. Journal of Zhejiang University-Science C, 2013, 14(2): 85-97.

[12] 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.

[13] XIE X X, MA H, LI J, *et al*. An efficient ciphertext-policy attribute-based access control towards revocation in cloud computing[J]. Journal of Universal Computer Science, 2013, (16):2349-2367.

[14] TAKERU N, MASAMI M, YOSHIAKI S. Provably secure attribute-based encryption with attribute revocation and grant function using proxy re-encryption and attribute key for updating. Human-Centric Computing and Information Sciences, 2015, (8):1-13.

[15] IBRAIMI L, TANG Q, HARTEL P, *et al*. Efficient and provable secure ciphertext-policy attribute-based encryption schemes [A]. Proc of the Information Security Practice and Experience[C]. Berlin, Heidelberg: Springer-Verlag, 2009. 1-12.

**作者简介：**

**闫玺玺**（1985-），女，河南灵宝人，河南理工大学讲师，主要研究方向为数字版权管理、数字内容安全、计算机网络安全。

**汤永利**[通信作者]（1972-），男，河南孟州人，博士，河南理工大学副教授，主要研究方向为计算机网络安全、数字内容安全、信息安全测评。E-mail:yltang@hpu.edu.cn。