# 基于CP-ABE和SD的高效云计算访问控制方案

陈燕俐 宋玲玲 杨 庚
(南京邮电大学计算机学院 南京 210003)

**摘 要** 存储在云端服务器中的敏感数据的保密和安全访问是云计算安全研究的重要内容。提出了一种安全、高效、细粒度的云计算访问控制方案。密文的加密采用了借助线性秘密共享矩阵的CP-ABE加密算法,并将大部分密文重加密工作转移给云服务提供商执行,在保证安全性的前提下,降低了数据属主的计算代价。该方案在用户属性撤销时,引入SD广播加密技术,有效降低了撤销时的计算开销和通信开销。理论分析表明该方案具有数据机密性、抗合谋攻击性、前向安全和后向安全,最后的实验结果验证了方案具有较高的撤销效率。

**关键词** 访问控制,云计算,子集差分,基于属性加密,撤销
**中图法分类号** TP309 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.09.029

## Efficient Access Control Scheme Combining CP-ABE and SD in Cloud Computing

CHEN Yan-li SONG Ling-ling YANG Geng
(School of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

**Abstract** The privacy and secure access of sensitive data stored in the cloud server is important content in cloud computing security research. A secure, effective, fine-grained access control scheme in cloud computing was proposed. The ciphertext encryption employs a CP-ABE with a linear secret sharing matrix, and most of the re-encryption work is transferred to the cloud service provider, so the scheme reduces the data owner's computational cost on the premise of security. When user attributes' revocation occurs, the scheme introduces SD broadcast encryption technology, effectively reducing the computing and communication overheads. The analysis shows that the scheme has the data confidentiality, collusion-resistance, backward and forward secrecy. Finally the experiment result validates the high revocation efficiency of the scheme.

**Keywords** Access control, Cloud computing, Subset difference, Attribute-based encryption, Revocation

## 1 引言

云计算是分布式计算、并行处理和网格计算的进一步发展,其依靠低成本、易于使用的接口和高扩展性的商业优势得到了学术界和工业界的广泛关注。云计算服务将数据属主(Data Owner, DO)的角色从数据提供者中分离开来,在提供数据服务时,数据属主和用户(User)不会有直接交互行为。目前云计算的访问控制场景中,访问控制策略的定义和执行是由完全可信的服务器来负责,但是不能确保云服务器在复杂的网络环境和多变的商业利益中能够始终保持安全机制的有效性。密码技术作为网络信息安全的核心技术,是保护数据最重要的工具之一,其中基于属性的加密(Attribute-based encryption, ABE)由于其独有的特性越来越受到人们的关注。

基于属性的加密是由模糊的基于身份的加密算法(Fuzzy IBE)发展而来的,很好地解决了数据访问控制领域无法进行的细粒度访问问题。基于属性的加密用属性集和访问控制策略将用户和数据关联在一起,只有属性满足访问策略,才可以将密文解密,从而进行访问控制。另外加密方可以把一个文档同时加密给多个具有相同属性的人,使得密文可以被多方共享,提高了系统的访问效率。基于属性的访问结构不仅增强了访问控制的灵活性,而且数据始终以密文形式存储也放宽了对服务器和访问存储器的安全限制。目前基于属性的加密在访问控制中已经得到了广泛应用,如Yu[1], Wan[3], Kan[16]。基于属性的加密一般分为密钥策略基于属性加密[6](Key-Policy Attribute-Based Encryption,简称KP-ABE)和密文策略基于属性加密[7](Ciphertext-Policy Attribute-Based Encryption,简称CP-ABE)。KP-ABE方案中加密者只能为数据选择描述性的属性,不能决定谁可以解密加密过的数据,只能相信密钥发布者;而CP-ABE方案中属性用来描述用户私钥,加密者可以使用访问策略来决定谁可以访问加密数据。因此,CP-ABE在访问控制中是更好的选择。基于CP-ABE的云计算访问控制方案中,用户的撤销是一个难题。因为每一个属性由多个用户共享,这就意味着任何一个用户的撤销都会影响到与他享有同样属性的所有用户。这将带来大量的

到稿日期:2013-11-08 返修日期:2014-02-15 本文受国家"九七三"重点基础研究发展规划:物联网混杂信息融合与决策研究(2011CB302903),国家自然科学基金项目:云计算环境下的新型访问控制理论与关键技术研究(61272084),江苏省自然科学基金(BK2009426)资助。

陈燕俐(1969-),女,博士生,教授,主要研究方向为计算机网络、信息安全,E-mail:chenyl@njupt.edu.cn;宋玲玲(1990-),女,硕士生,主要研究方向为信息安全;杨 庚(1961-),男,博士后,教授,博士生导师,主要研究方向为计算机通信与网络、网络安全、分布与并行计算等。

计算开销，成为系统瓶颈。

### 1.1 相关工作

目前的访问控制撤销方法可分为用户撤销和属性撤销两类。

(1)用户撤销：当一个用户离开某个系统时，该用户会失去系统中所有数据的访问权限，这称为用户撤销。目前访问控制方案中，如 Yu[1]，DO[2]等，采用的是用户撤销。Yu[1]通过更新系统属性列表中的属性版本号，生成重加密密钥，更新密文和用户私钥，但是每次有用户撤销，非撤销用户的所有密钥都需要更新。DO[2]中增设了特权管理组，撤销过程中由云服务器更新私钥，特权管理组更新密文，因此该方案并没有很好地利用云服务器中的计算和存储资源。Attrapadung[10]方案中，属主需要管理每一个属性组中所有的用户列表，以便直接进行用户撤销。这也不适合云存储环境，因为属主将加密数据交给云服务器后，不会和用户有直接的交互。Eissa[9]结合 KP-ABE 方案，用户撤销后，在其私钥的访问结构中添加非属性，密文属性集中若有满足其非属性的属性值时，用户便不能解开密文，该方案的密钥策略比较复杂，且属主需要一直在线生成代理重加密密钥。

(2)属性撤销：当一个用户的级别在系统中发生改变，如等级降低时，只损失了部分属性，这称为属性撤销。目前 Wan[3]，Liang[4]，Sahai[5]等提出的 ABE 方案采用的是属性撤销方法。方案是通过更新时间机制来实现撤销，但是这些通过更改过期时间属性值的方式不能达到立即撤销的效果，只能是粗粒度的撤销。撤销过程有两个主要问题，首先是保持后向安全(backward secrecy)和前向安全(forward secrecy)[12]，由于时间窗口的脆弱性，新加入的用户可以通过持有的属性密钥访问到之前的加密数据，直到数据被周期性更新的密钥重新加密(后向安全性)；被撤销的用户可能在私钥 SK 的过期时间之前仍然能够访问加密的数据(前向安全性)。其次是可伸缩性问题，目前的撤销方案没有考虑如何灵活地将更新的属性密钥发送给非撤销用户。因此，在云计算中如何设计一个高效的细粒度属性撤销机制是本文研究的主要内容。

### 1.2 本方案的贡献

本文针对在云计算中数据访问控制策略以及数据安全性问题，提出了一个细粒度的访问控制以及撤销方案，本方案的主要优势如下：

1)本文提出的处理大规模系统中用户属性撤销问题的访问控制方案，可以表示任意的访问控制策略，加强了系统用户访问权限变动时外包数据的后向/前向安全问题，减少了密文更新过程中的计算和传输的开销。

2)用户撤销可以在每个属性级别完成而不是系统级别，使得更大程度上的细粒度访问控制成为了可能。在现实中，用户不会一直在线，可能会错过许多密钥更新信息，不能保持他们的密钥处于最新状态，这被称为接受者状态问题。在本方案中，密钥更新采用子集差分算法(Subset Difference，SD)无状态组密钥分发机制，缓解了可伸缩性问题并且不需要用户一直在线，并有效降低了传输开销。

3)本文对算法的安全性进行了理论分析，并通过仿真实验验证了方案在撤销时的高效性。

本文第 2 节讲述本文的系统模型和安全需求；第 3 节为相关基础知识；第 4 节具体描述了带有高效撤销功能的 CP-ABE 方案；第 5 节为安全性分析；第 6 节为本方案的效率分析；最后为本文的总结。

## 2 模型和假设

### 2.1 系统模型

本文构造的系统模型如图 1 所示。系统中有 4 个实体：可信机构、数据属主、云服务器、用户。

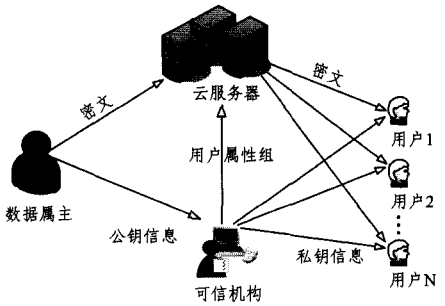


图 1 系统模型

可信机构为系统产生公钥和私钥参数，根据用户的属性授予不同的访问权限，可信机构是数据外包系统中唯一的完全可信的机构。数据属主决定访问策略，并用访问策略加密数据文件，然后将密文传送到云服务器。云服务器存储加密过的数据文件，负责管理用户对密文的访问和提供相关的数据，为每一个属性组生成组密钥，并结合广播加密技术加密属性组密钥。当用户的一个属性撤销时，云服务器重新生成该属性组的密钥，并用该密钥去更新与该属性相关的密文组件。系统中的云服务器具有“诚实但好奇”的特性，即它会诚实地执行系统中合法实体分配的任务，也会对存储的加密文件产生兴趣。用户在系统中根据身份或角色获得属性集。然而用户的属性集可能由于角色改变发生变动。例如，当一个公司职员的职位降低时，一些属性会被撤销；当职位升高时，一些新属性会被赋予。用户只有在他的有效属性满足与密文相关的访问策略的时候才可以解开密文。

### 2.2 系统威胁模型和安全需求

(1)数据机密性。未认证的用户不具有满足访问策略的属性，将会被阻止访问数据明文。此外，当“诚实但好奇”的云服务器不合法地访问它所存储的密文时，访问也会被阻止。

(2)抗合谋攻击。即使多个非授权用户合谋，也不可能通过将各自的属性组合起来的方式成功解开密文。

(3)后向和前向安全。后向安全意为任何一个用户持有一个满足密文访问策略的属性集时，他对持有该属性集之前交换的加密数据的访问是被阻止的。另外，前向安全意为任何一个用户失去一个属性后，他对随后交换的加密数据的访问是被阻止的，除非他其余的有效属性仍然满足该密文访问策略。

## 3 预备知识

### 3.1 访问结构

令 $P=\{P_1,P_2,\cdots,P_n\}$ 是参与方的集合，一个访问结构 $\mathbb{N}$ 是 $2^P$ 的一个非空子集，即 $\mathbb{N}\subseteq 2^{\{P_1,P_2,\cdots,P_n\}}\setminus\{\Phi\}$。若访问结构 $\mathbb{N}\subseteq 2^{\{P_1,P_2,\cdots,P_n\}}$ 是单调的，则有：$\forall B,C$，若 $B\in\mathbb{N}$ 且 $B\subseteq C$，则有 $C\in\mathbb{N}$。访问结构 $\mathbb{N}$ 中的集合称为授权集合，不在访

问结构$\mathbb{N}$中的集合称为非授权集合。

### 3.2 双线性配对

令$G,G_T$是两个阶为素数$p$的循环群，$g$是$G$的一个生成元。定义一个可有效计算的双线性映射$e:G\times G\to G_T$，该映射必须满足：

a)双线性：$\forall a,b\in Z_N, e(g^a,g^b)=e(g,g)^{ab}$。

b)非退化性：存在$g\in G$，使得$e(g,g)\neq 1$。

c)可计算性：$\forall u,v\in G, e(u,v)$都是可以有效计算的。

### 3.3 线性秘密共享 LSSS

一个定义在实体集$P$上的线性秘密共享方案[14](Linear Secret Sharing Scheme，LSSS)$\Pi$是指：

(1)所有实体的共享组成$Z_P$上的一个向量；

(2)存在一个$l\times n$的$\Pi$的共享生成矩阵$M$和一个从$\{1,\cdots,l\}$到$P$的映射$\rho$，随机选取$v=(s,v_2,\cdots,v_n)\in Z_p^n$，其中$s$是要共享的秘密，则$Mv^{\mathrm{T}}$就是利用$\Pi$得到的关于$s$的$l$个共享组成的向量，其中共享$(Mv^{\mathrm{T}})_i$属于实体$\rho(i)$，表示为$\lambda_i=(Mv^{\mathrm{T}})_i$。

按照上述方法定义的 LSSS 具有线性可重构性：假设$\Pi$是一个针对访问结构$\Lambda$的 LSSS，对授权用户集$S\in\Lambda$，定义$I=\{i:\rho(i)\in S\}\subseteq\{1,\cdots,l\}$，存在向量$w=\{\omega_i\in Z_p\}_{i\in I}$，使得$\sum_{i\in I}\omega_i M_i=(1,0,\cdots,0)$，从而得到：$\sum_{i\in I}\omega_i M_i v^{\mathrm{T}}=\sum_{i\in I}\omega_i\lambda_i=\sum_{i\in I}(\omega_i M_i)v^{\mathrm{T}}=s$。对于非授权用户集，存在向量$w\in Z_p^n$，使得$w(1,0,\cdots,0)^{\mathrm{T}}=-1, wM_i^{\mathrm{T}}=0, i\in I$。

### 3.4 困难假设

本文对判定性 q-parallel BDHE 假设[11]作如下阐述。根据安全参数选择一个阶为素数$p$的群$G$，$g$是$G$的生成元，随机值$a,s\in Z_p$。敌手获得$\vec{y}=g,g^s,g^a,\cdots,g^{a^q},g^{a^{q+2}},\cdots,g^{a^{2q}},\forall 1\leqslant j\leqslant q, g^{s\cdot b_j},g^{a/b_j},\cdots,g^{a^q/b_j},g^{a^{q+2}/b_j},\cdots,g^{a^{2q}/b_j},\forall 1\leqslant j,k\leqslant q,k\neq j, g^{a\cdot s\cdot b_k/b_j},\cdots,g^{(a^q\cdot s\cdot b_k/b_j)}$后，必须将$e(g,g)^{a^{q+1}s}\in G_T$与$G_T$中的随机值区分开来。

算法在处理$G$中判定性 q-parallel BDHE 问题时输出最后猜测$z\in\{0,1\}$的优势为$\varepsilon$，如果以下条件满足：$|\Pr[B(\vec{y},T=e(g,g)^{a^{q+1}s})=0]-\Pr[B(\vec{y},T=R)=0]|\geqslant\varepsilon$。

**定义 1** 我们认为如果在任何多项式时间内敌手都只有可忽略的优势攻破判定性 q-parallel BDHE 假设，那么判定性 q-parallel BDHE 假设成立。

### 3.5 子集差分法[13](Subset Difference，SD)

用$N$表示广播加密系统用户集合，$T$表示合法用户集合，$R=N\backslash T$表示撤销用户集合。用户被视为一棵完全二叉树的叶子节点。子集$S_1,\cdots,S_w$的集合用两个用户集的差来定义，例如一个用户群$U_1$减去另一个用户群$U_2$，其中$U_1\subset U_2$。$U_1,U_2$与两棵满二叉树的叶子节点相关。因此在 SD 方案中一个有效子集$S$由树中两个节点表示$(v_i,v_j)$，其中$v_i$为$v_j$的祖先节点，用$S_{i,j}$表示这样一个子集。如果一个用户叶子节点$u$在以$v_i$为根节点但不在以$v_j$为根节点的子树中，那么称$u\in S_{i,j}$。SD 方案最重要的特性是使用户属于更多的子集来减少通信开销。

对于给定的撤销用户集$R$，令$R=\{u_1,\cdots,u_r\}$。子集覆盖是指不相交的$N\backslash R$的分割子集$S_{i_1,j_1},S_{i_2,j_2},\cdots,S_{i_m,j_m}$(其中$S_{i,j}\subseteq N\backslash R, 1\leqslant j\leqslant m$)满足：

(1)每个子集$S_{i,j}$有一个相应的密钥$L_{i,j}$，用户$u\in S_{i,j}$可以通过其秘密信息$P_u$计算获得密钥$L_{i,j}$；

(2)对于每个撤销集合$R\subseteq N$，存在不相交的子集$S_{i1,j1}$，$S_{i2,j2},\cdots,S_{im,jm}$使得$N\backslash R=\bigcup_{j=1}^{m}S_{i,j}$。

将$N\backslash R$分割成不相交的子集$S_{i_1,j_1},S_{i_2,j_2},\cdots,S_{i_m,j_m}$，方法如下：令$ST(R)$为根节点和撤销用户集$R$生成的 Steiner 树。在建立子集集合的过程中，始终保持$ST(R)$的子树$T$使得对于任意的$T$的叶子下的$u\in U\backslash R$被覆盖。首先使得$T$与$ST(R)$相等，然后通过循环算法依次除去$T$的节点(同时将子集添加到集合中)，直到$T$只包含了一个节点。对于任意的$r$个撤销用户，覆盖算法中最多包含$2r-1$个子集。此外，如果撤销用户是随机的，那么产生的分割子集数平均为$1.25r$。

## 4 本文提出的方案

### 4.1 主要思想

为了实现细粒度访问控制，属主首先将数据文件用对称加密密钥 DEK 加密文件，然后属主对对称加密密钥 DEK 使用采用 LSSS 的 CP-ABE 算法加密[11]，这样只有用户的属性集满足 DEK 密文的访问策略时，用户才能获得 DEK，最后解出数据文件。

本文提出的访问控制方案，为系统中所有的属性创建一个属性组，每一个属性组只包括具有该属性的用户。方案为每一个属性组都分配一个组密钥，由该属性组中的用户共享。当用户发生属性撤销，即从某个属性组中移除时，系统需要更新这些撤销属性的组密钥，并结合广播加密技术加密并发布新的组密钥给请求访问的用户。为了提高系统效率，本方案将这部分计算任务交给云服务器完成。另外，密文也需要通过云服务器利用新的属性组密钥对其相关部分进行更新。这样新加入的用户以及非撤销用户使用最新版本的私钥可以解开最新版本的密文，而撤销用户由于无法获得更新后的属性组密钥，不能进行密钥更新，从而不能解开更新后的密文。

### 4.2 方案描述

令$U=\{u_1,\cdots,u_n\}$表示系统中的所有用户，$L=\{x_1,\cdots,x_p\}$表示系统中所有可描述的属性，$U_i\subset U$表示所有享有属性$x_i$的用户集合。$U_i$将作为一个属性$x_i$的访问或撤销列表。$\delta=\{U_1,\cdots,U_p\}$是以上所有属性组的集合。属性组密钥$V_{x_i}$由$U_i\subset\delta$中的每一个非撤销用户共享。

1)系统建立

可信机构运行初始化 setup 算法，初始化算法选择一个$p$阶的双线性群$G$，生成元$g$，假设属性可以用$Z_p$中的元素表示，选择两个随机数$\alpha,a\in Z_p$。然后选择一个哈希函数：$H:\{0,1\}^*\to G$。最后系统中发布的公钥如下：$PK=(g,e(g,g)^a,g^a)$，可信机构将$MK=g^\alpha$作为主密钥保存起来。

2)用户私钥的生成

该过程由两部分组成，首先由可信机构生成用户的私钥，然后由云服务器为用户生成用于解开密文头部的密钥。

用户私钥生成：当一个用户加入系统时，可信机构首先根据他的身份或角色分配一个属性集合$S$。然后可信机构通过运行密钥生成算法为用户生成私钥。密钥生成算法将主密钥和用户的属性集$S$作为输入，选择一个随机数$t\in Z_p$，然后计算出私钥如下：

$$SK=(K=g^\alpha g^{at},L=g^t,\forall x\in S:K_x=H(x)^t)$$

可信机构对为统中每一个属性 $x_j \in L$ 都建立起一个用户数组 $U_j$。例如，当用户 $u_1, u_2, u_3$ 拥有的属性集合分别为 $\{x_1, x_2, x_3\}, \{x_2, x_3\}, \{x_1, x_3\}$ 时，$U_1 = \{u_1, u_3\}, U_2 = \{u_1, u_2\}, U_3 = \{u_1, u_2, u_3\}$。这样就将所有的用户集合 $U$ 分成了不同的子集 $U_1, U_2, \cdots, U_p \subseteq U$。可信机构将属性组的信息发送给云服务器。

为用户生成秘密信息 $P_u$：根据子集差分法[13]，首先云服务器为系统中所有用户创建一棵完全二叉树，用户被视为该树的叶子节点。通过一个伪随机序列生成器[15]为树中的内部节点生成独立随机的标记值。这些标记值可以推导出所有合法子集 $S_{i,j}$ 的密钥。如果合法用户 $u$ 是子树 $T_i$ 的叶节点，$T_i$ 中的节点 $j$ 不是 $u$ 的祖先节点，$u$ 应当能通过秘密信息 $P_u$ 计算出 $L_{i,j}$。假设 $T_i$ 中节点 $v_{i1}, v_{i2}, \cdots, v_{ik}$ 与根节点 $v_i$ 到 $u$ 的路径相邻，但不是 $u$ 的祖先节点。另外，节点 $j$ 是 $v_{i1}, v_{i2}, \cdots, v_{ik}$ 中某一个节点的子孙节点。如果 $u$ 接收到节点 $v_{i1}, v_{i2}, \cdots, v_{ik}$ 的标记值作为秘密信息 $P_u$ 的一部分，则用户 $u$ 最多操作伪随机序列生成器 $\log N$ 次，可有效计算出 $L_{i,j}$。系统中用户 $u$ 属于 $\log N$ 棵这样的子树 $T_i$，因此用户 $u$ 需要存储的秘密信息即子树中节点标记值的总量为 $\frac{1}{2}\log^2 N + \frac{1}{2}\log N + 1$。

3)文件的创建

(1)属主加密数据文件：在上传数据文件给云服务器之前，属主首先对数据文件做如下操作：

· 为该数据文件选择一个唯一的 ID 号；

· 随机选择一个对称加密密钥 $DEK \xleftarrow{R} \kappa$，$\kappa$ 是密钥空间，然后使用 DEK 加密该数据文件；

· 接着加密 DEK，将公钥 $PK$、数据文件 $M$($M$ 即代表 DEK)，LSSS 访问结构 $(M, \rho)$ 作为输入。函数 $\rho$ 将矩阵 $M$ 的行矩阵与文件属性联系起来。在本构造中，我们定义 $\rho$ 是单射函数，即一个属性最多可以和 $M$ 的一个行对应。

$M$ 是一个 $l \times n$ 矩阵，算法首先选择一个随机向量 $v = (s, v_2, \cdots, v_n) \in Z_p^n$。这些值将用来共享随机值 $s$。从 $i=1$ 到 $l$，属主计算 $\lambda_i = (Mv^T)_i$。另外，算法选择一系列随机值 $r_1, \cdots, r_l \in Z_p$。

· 最后将 LSSS 的访问结构 $(M, \rho)$ 和密文提交给服务器，密文如下：$CT_0 = (C = Me(g,g)^{\alpha s}, E = g^s, C_i = g^{a\lambda_i} H(\rho(i))^{-r_i}, D_{0i} = g^{r_i}, (i = 1, \cdots, l))$。

(2)云服务器收到密文后，运行二次加密算法 SeEncrypt $(CT, U)$ 对密文进行重加密。该算法在每一个属性组的基础上加强了用户级访问控制。

算法操作如下：对密文中所有相关的属性组 $U_i \in U$，选择一个随机的 $V_{x_i} \in Z_p^*$，然后重加密密文 $CT = (C = Me(g,g)^{\alpha s}, E = g^s, C_i = g^{a\lambda_i} H(\rho(i))^{-r_i}, D_i = (g^{r_i})^{V_{x_i}}, (i = 1, \cdots, l))$。

云服务器对每一个属性组运用子集差分法。各个属性组只包括具有该属性的用户，加密算法将每个属性组中有效用户集合划分为互不相交的子集，例 $\{p_{i_1,j_1}, p_{i_2,j_2}, \cdots, p_{i_m,j_m}\}$。这些集合的并集覆盖了所有具有该属性的用户，即每个具有该属性的用户属于且仅属于这其中的某一个集合。这些划分子集的对应密钥 $\{k_{i_1,j_1}, k_{i_2,j_2}, \cdots, k_{i_m,j_m}\}$ 用 $KEK(G_{x_i})(i=1, \cdots, l)$ 表示，并用 $KEK(G_{x_i})$ 中的每一个值对属性组密钥 $V_{x_i}$ 进行加密。

(3)生成一个头部文件：$Hdr = (\forall 1 \leqslant i \leqslant l : \{E_K(V_{x_i})\}_{K \in KEK(G_{x_i})})$。其中 $E_K(f)$ 是信息 $f$ 在密钥 $K$ 下的一个对称加密算法，$E_K : \{0,1\}^k \to \{0,1\}^k$。这个加密算法用来将属性组密钥安全传送给合法用户。最后，每个数据文件以图 2 的形式存储在云服务器中。当收到用户发来的文件访问请求后，云服务器将以上文件发送给用户。

| ID | Hdr | CT | $\{DataFile\}_{DEK}$ |
|---|---|---|---|

图 2　存储在云服务器中的数据文件的格式

4)文件访问

(1)Hdr 解密：用户 $u$ 从云服务器接收到密文文件后，在文件头部中寻找到用自身所属集合的密钥加密的数据，然后用自身持有的秘密信息 $P_u$ 计算出 $k_{i,j}$，并用其对加密后的属性组密钥进行解密，即可获得属性组密钥 $V_{x_i}$。如果 $u \notin p_{i,j}$，计算将终止，因为该用户没有该属性组相应的解密信息。这里所使用的技术为完全子树[13]中查表结构的方法，查表时间复杂度为 $O(\log \log N)$。假设用户私钥的属性集 $S$ 满足密文的访问结构，将 $I \subset \{1, 2, \cdots, l\}$ 定义为 $I = \{i : \rho(i) \in S\}$。利用解出的 $V_{x_i}$ 对用户保存的私钥作如下计算：

$$SK = (K = g^{\alpha} g^{at}, L = g^t, \forall x_i \in S : K_{x_i} = (H(x)^t)^{\frac{1}{V_{x_i}}}, i \in I)$$

(2)数据文件解密：只有当用户所拥有的属性集满足密文的访问结构时，用户才可以成功地运行解密算法获得数据文件。解密算法将访问结构 $(M, \rho)$、密文和用户私钥作为输入。如果 $\{\lambda_i\}_{i \in I}$ 是任意秘密共享值 $s$ 的有效份额，那么存在 $\{w_i \in Z_p\}_{i \in I}$，使得 $\sum_{i \in I} w_i \lambda_i = s$。

解密算法首先计算出：

$$\begin{aligned} & e(E,K)/\left(\prod_{i \in I}(e(C_i, L)e(D_i, K_{\rho(i)}))^{w_i}\right) \\ &= e(g,g)^{\alpha s} e(g,g)^{ast} / \left(\prod_{i \in I}(e(g,g)^{a\alpha\lambda_i w_i})\right) \\ &= e(g,g)^{\alpha s} \end{aligned}$$

最后可以从密文组件 $C$ 中得出数据文件 $M$。

5)属性撤销

在大规模的数据存储系统中，用户的访问权限可能会动态改变。当用户的某个属性被撤销时，可信机构更新属性组的成员列表，并通知云服务器更新受影响的属性的组密钥。此时密文更新部分若由属主完成，将给属主带来巨大的计算开销。为了提高工作效率，本方案将密文更新的工作量交给云服务器完成，这样能大大降低属主的计算开销。密文更新通过代理重加密的方式完成，云服务器在更新密文之前不需要解开密文。

当属性组有用户加入或离开时，云服务器运行密文更新算法 CTUpdate，更新相应的密文组件。为了不失一般性，假设某属性 $x$ 的属性组 $U_j$ 中有成员关系变动(如在某时刻一个用户获得或撤销该属性)，则云服务器为有成员变更的属性组随机选择一个新的属性组密钥 $V_{xj}' \in Z_p^*$ 和一个随机数值 $s' \in Z_p^*$，取出 LSSS 访问控制矩阵 $M$ 的第一列 $M_{i,1}$，并将三者与密文 $CT$ 作为输入，利用公钥信息，将密文更新如下：

$$CT' = (C' = C \cdot e(g,g)^{\alpha s'}, E' = E \cdot g^{s'}, C_i' = C_i \cdot g^{a s' M_{i,1}}$$

if $\rho(i) \neq x_j : D_i' = D_i$；

if $\rho(i) = x_j : D_i' = D_{0i}^{V_{xj}'}, (i = 1, \cdots, l))$

当某属性 $x$ 的属性组 $U_j$ 中有 $R$ 个用户撤销时，调用子

集覆盖算法，将授权用户集合 $N_{xj}\backslash R$ 重新划分为互不相交的子集，并用这些新的子集对应的密钥对新的属性组密钥 $V_{xj}'$ 进行加密，对 Hdr 头部文件进行更新，如下：

$$Hdr=(\{E_K(V_{xj}')\}_{K\in KEK(G_{xj})};\ \forall 1\leqslant i\leqslant l, i\neq j:\{E_K(V_{x_i})\}_{K\in KEK(G_{xi})})$$

可以看出，当发生用户属性撤销时，属主不需要对密文进行重加密，只需云服务器利用公钥信息和密文进行更新，并且与撤销属性无关的密文组件不需要进行更新。因此，本方案大大提高了整个系统的效率。

该密钥更新过程保证了属性组中用户撤销时的细粒度访问控制，同时是通过立即生效的方式实现属性撤销，而不是通过时间机制来完成。另外，本方案中用户的撤销是基于属性级别而不是系统用户级别。因此，即使用户从某个属性组中撤销，他仍然可能通过其另外的属性去访问系统中其他的数据，只要这些有效属性满足加密数据的访问控制策略。

6)文件删除

该操作只有在数据属主向云服务商提出请求后才可以执行。属主将需要删除的文件的 ID 号和在该 ID 上的签名发送给云服务商。云服务商验证过属主签名之后返回为真，即可删除该数据文件。

## 5 安全性分析

### 5.1 数据机密性

本方案阻止未认证用户以及“诚实但好奇”的云服务器访问加密数据。方案中，新建文件的加密包括数据文件的机密性、对称密钥 DEK 的机密性以及属性组密钥的机密性。假设加密数据文件和属性组密钥的对称加密算法是安全的(例如 AES)，那么机密性就主要取决于 DEK 的安全性。DEK 的加密采用的是基于 LSSS 的 CP-ABE 的加密和解密算法，文献[11]已经证明算法是安全的，攻击者的属性集若不能满足访问策略，就无法计算出对称密钥，从而不能访问数据文件。当一个用户从某个满足密文访问策略的属性集中撤销时，除非他其余的有效属性依然能够满足该密文的访问策略，否则他将无法解开密文。因为当他从某属性组撤销时，无法再得到该属性组更新后的属性组密钥，因此也无法通过配对运算恢复出对称密钥 DEK，从而得到数据文件。

另一种攻击来自云服务器。因为云服务器可能会被破坏或者云服务器本身会为了谋利而试图恶意利用所存储的加密数据，所以不能被数据属主完全信任。抗服务器攻击的外包数据机密性是安全数据外包的另一个基本安全标准。本方案中即使云服务器管理分发每一个属性组密钥，它也不能解开密文，因为云服务器只被授权去利用每个属性组密钥对密文进行重加密，可信机构并没有将与属性集相关的私钥发送给云服务器，云服务器无法解开密文。因此，抗服务器攻击的数据机密性也得到了保证。

### 5.2 抗合谋攻击

基于属性的加密算法最大的挑战是防止合谋用户的攻击。在 CP-ABE 中，秘密共享值 $s$ 嵌入在密文中。为了解开密文，用户或者合谋攻击者需要将 $e(g,g)^{\alpha s}$ 恢复出来。为了恢复出 $e(g,g)^{\alpha s}$，合谋攻击者必须利用密文中的组件 $C_i$、$D_i$ 和其他合谋用户的私钥组件 $L$、$K_x$ 作相应的双线性配对运算。但是，每一个用户的私钥都通过一个随机数 $t$ 唯一生成，每个用户的 $t$ 不同，因此即使用户共谋，$e(g,g)^{\alpha s}$ 的值也不会被恢复。只有当该用户具有的属性满足访问策略时，$e(g,g)^{\alpha s}$ 值才会被恢复。

另一种合谋攻击形式为不属于某属性组的外部用户，通过合谋试图获得他们不具备的属性的有效的属性组密钥。本方案提出的属性组密钥分发协议具有密钥——不可分辨性特点，旨在使外部合谋用户看来，属性组内部用户所拥有的秘密信息值和随机值没有区别，这个特点妨碍了合谋用户去获得需要用来解开属性组密钥的秘密信息。因此，属性组外部用户的合谋攻击不能帮助他们获得并不享有的属性的属性组密钥。

### 5.3 后向和前向安全

当一个用户在某个时刻获得一个满足密文访问策略的属性集时，与这些属性相一致的属性组密钥需要进行更新，并通过安全渠道分发给这些属性组中的有效用户(包括该用户)。密文中与秘密值 $s$ 相关的密文组件都通过云服务器选择一个随机值 $s'$ 进行重加密，同时与撤销属性相关的密文组件用更新后的属性组密钥进行重加密。即使该用户保存了在他获得属性集之前系统中的密文，他也不能解开。因为，即使他可以从当前密文中计算出 $e(g,g)^{\alpha(s+s')}$ 的值，由于 $s'$ 的值无法获得，因此也不能计算出 $e(g,g)^{\alpha s}$ 的值。因此，本方案的撤销算法具有后向安全性。

当一个用户在某个时刻撤销一个属性集，即从一些属性组中移除时，与这些属性相一致的属性组密钥需要更新，并通过安全渠道分发给这些属性组中的有效用户(不包括该用户)。密文中与秘密值 $s$ 相关的密文组件都通过云服务器选择一个随机值 $s'$ 进行重加密，同时与撤销属性相关的密文组件用更新后的属性组密钥进行重加密。被撤销属性的用户无法获得更新后的属性组密钥，无法对私钥进行重计算，因而不能解开被重加密的密文。该用户即使在撤销属性之前成功地进行过解密，计算并保存了 $e(g,g)^{\alpha s}$ 的值，也由于随后的密文是通过一个随机值 $s'$ 重加密过的而无法计算出 $e(g,g)^{\alpha(s+s')}$ 的值，因此无法解密。所以，本方案的撤销算法具有前向安全性。

## 6 性能分析

### 6.1 理论分析

本方案和 Hur[8] 方案的性能比较见表 1，其中 $n$ 为一个属性组中的用户数量，$l$ 为密文中相关的属性数量，$r$ 为撤销的属性数量。

表 1 两种访问控制方案的比较

| 方案 | 安全模型 | 访问结构 | 密文长度 | 撤销时计算复杂度 |
|---|---|---|---|---|
| Hur[8] 方案 | 通用群模型 | 访问树 | $O(l\log n)+O(1)$ | $O(3l+1)$ |
| 本方案 | 标准安全模型 | LSSS | $O(l)+O(l)$ | $O(l+r)$ |

本方案中的访问控制使用了 LSSS 秘密共享体制，可以表示任意类型的访问结构，并且方案的安全性更高。本方案用户撤销时，属性组用户子集的划分采用子集差分算法，子集覆盖过程中减少了被分割成的子集树，其被分割成的子集树最多为 $2r-1$(平均情况下为 $1.25r$)，从而减少了通信开销。用户属性撤销的开销主要包括对数据密文的重加密和对属性组密钥的重加密。由于本方案与 Hur 方案在对属性组密钥

重加密算法上没有较大区别，这里主要考虑密文重加密的计算开销。Hur 方案中计算开销主要为 1 次 $G_T$ 上的指数运算，1 次 $G_T$ 上的乘法运算，$3l+1$ 次 $G$ 上的指数运算，$2l+1$ 次 $G$ 上的乘法运算，$G$ 上的乘法运算开销远远小于 $G$ 上的指数运算，因此计算复杂度为 $O(3l+1)$。本方案中计算开销主要为 1 次 $G_T$ 上的指数运算，1 次 $G_T$ 上的乘法运算，$l+r$ 次 $G$ 上的指数运算（由于访问控制矩阵 $M$ 第一列 $M_{i,1}$ 中的 $l$ 个值不一定全为 1，1 的个数与访问策略中或门的个数有关，并且 1 的个数会影响到密文重加密时 $G$ 上的指数运算个数，这里是按最复杂的情况作比较，即 $l-1$ 个），$l+1$ 次 $G$ 上的乘法运算，$G$ 上的乘法运算开销远远小于 $G$ 上的指数运算，因此计算复杂度为 $O(l+r)$。一般情况下，一次撤销属性的数量远小于密文中属性的数量。

### 6.2 实验仿真

为了评估本方案的效率，将其与 Hur 方案在以下平台上进行了仿真实验。硬件：Intel T5850 @ 2. 2GHz，2GBytes RAM，平台：Windows，代码库：Miracl。我们使用的椭圆曲线为 512bit，该椭圆曲线群的阶数为 160bit，即 $p$ 是一个 160bit 长的大素数。

本方案与 Hur 方案的加密、解密计算效率仿真结果如图 3 和图 4 所示。仿真结果表明，两种方案的加密时间与密文属性数量呈一定的线性关系，由于本方案在加密时比 Hur 方案多 $l$ 个群 $G$ 上的指数运算，因此加密效率略低于 Hur 方案。CP-ABE 方案的解密时间与私钥中属性数量和密文策略相关，由于 Hur 方案在解出秘密值 $s$ 时需要递归运算，本方案使用 LSSS 秘密共享方案则不需要，因此解密效率略高于 Hur 方案。

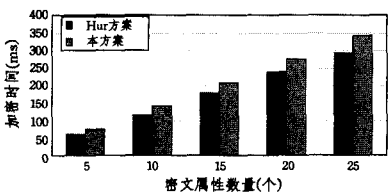


图 3 加密运算

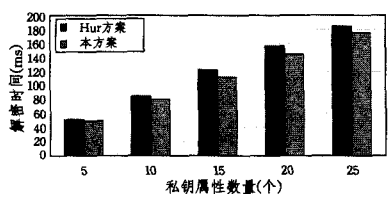


图 4 解密运算

将本方案与 Hur 方案在属性撤销时密文重加密的计算效率进行比较，对于具有不同属性数量的密文，假设撤销属性数量为固定个数，仿真结果如图 5 所示。仿真结果表明本方案在属性撤销时具有更高的效率。

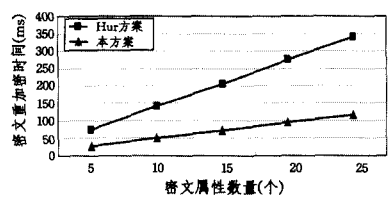


图 5 密文重加密

**结束语** 本方案的访问控制采用基于 LSSS 结构的 CP-ABE 算法，结合子集差分法使得该细粒度访问控制具有用户属性级撤销能力，同时运用代理重加密技术将用户属性撤销时的大部分计算开销转移给云服务器完成，降低了数据属主的计算开销。本方案的访问控制策略，采用秘密共享方案，能够表示任意访问结构，且解密效率较高，撤销时的计算复杂度与撤销属性数量和密文访问策略相关，计算效率较高。本方案具有数据机密性、抗合谋攻击性、前向安全和后向安全。本方案中属性组用户子集的划分采用子集差分法，减少了被分割成的子集树，从而减少了通信开销，但该算法中的建树过程需要较大计算开销，如何减少建树的计算开销是今后需要进一步研究的问题。

## 参 考 文 献


[1] Yu Shu-cheng, Wang Cong, Ren Kui, et al. Achiving secure, scalable, and fine-grained data access control in cloud computing [C] // INFOCOM, 2010 Proceedings IEEE. San Diego, CA, 2010: 1-9

[2] Do Jeong-min, Song You-jin, Park N. Attribute based Proxy Re-Encryption for Data Confidentiality in Cloud Computing Environments[C] // 2011 First ACIS/JNU International Conference on Computers, Networks, Systems and Industrial Engineering (CNSI). IEEE, Jeju Island, 2011: 248-251

[3] Wan Zhi-guo, Liu Jun'e, Deng R H. HASBE: A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing[J]. IEEE Transaction on Information Forensics and Security, 2012, 7(2): 743-754

[4] Liang Xiao-hui, Lu Rong-xing, Lin Xiao-dong, et al. Ciphertext Policy Attribute Based Encryption with Efficient Revocation [R]. Technical Report, University of Waterloo, 2010

[5] Sahai A, Seyalioglu H, Waters B. Dynamic Credentials and Ciphertext Delegation for Attribute-Based Encryption[M] // Advances in Cryptology-CRYPTO 2012. Springer Berlin Heidelberg, 2012: 199-217

[6] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data[C] // Proceedings of the 13th ACM conference on Computer and communications security. ACM, New York, NY, USA, 2006: 89-98

[7] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption[C] // IEEE Symposium on Security and Privacy, 2007(SP'07). Berkeley, CA, United states, 2007: 321-334

[8] Hur J, Noh D K. Attribute-based access control with efficient revocation in data outsourcing systems[J]. IEEE Transaction on Parallel and Distributed Systems, 2011, 22(7): 1214-1221

[9] Eissa T, Cho G-H. A Fine Grained Access Control and Flexible Revocation Scheme for Data Security on Public Cloud Storage Services[C] // 2012 International Conference on Cloud Computing Technologies, Applications and Management (ICCCTAM). Dubai, 2012: 27-33

[10] Attrapadung N, Imai H. Conjunctive broadcast and attribute-based encryption [M] // Pairing-Based Cryptography-Pairing 2009. Springer Berlin Heidelberg, 2009: 248-265

[11] Waters B. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization[M] // Public Key Cryptography-PKC 2011. Springer Berlin Heidelberg, 2011: 53-70


（下转第 168 页）

比较分析(见表1)时，主要考虑以下指标：公钥长度、私钥长度、签名时是否需要高斯抽样运算、消息与签名的总长度、安全证明是否使用随机预言机模型。表1中长度单位均为比特，正整数 $n$ 为安全参数，$m=O(n\log n)$，$q=O(n^2)$，$k=O(\log n)$，消息记为 $\mu$，$|\mu|$ 表示消息的比特长。为方便比较，尽可能统一参数，以格中短基为私钥的格是满秩的，文献[14]格签名方案中 $d=1$。

从表1可以看出，文献[9-13]中格签名方案消息和签名总长度较长，且需要用到昂贵的高斯原像抽样运算，效率较低，其中仅文献[9,12,13]中方案在标准模型下可证明安全。由于新方案与文献[14,15]格签名方案中 $\mathbf{z}$ 的每个分量都不超过 $12\sigma$，所以其长度可以表示为 $m\log(12\sigma)$，同时它们都在随机预言机模型下可证明安全。进一步地，与文献[14,15]中的方案相比，我们的新方案缩短了消息和签名的总长度，具有优势。

表1 性能比较

| 方案 | 公钥长度 | 私钥长度 | 消息-签名总长度 | 是否抽样 | 预言机模型 |
|---|---|---|---|---|---|
| 文献[9] | $(\|\mu\|+1)nm\log q$ | $m^2\log q$ | $\|\mu\|+m(\|\mu\|/2+1)\log q$ | 是 | 标准模型 |
| 文献[10] | $nm\log q$ | $m^2\log q$ | $\|\mu\|+2m\log q$ | 是 | 随机预言机模型 |
| 文献[11] | $nm\log q$ | $m^2\log q$ | $\|\mu\|+n+m\log q$ | 是 | 随机预言机模型 |
| 文献[12] | $(2\|\mu\|+1)nm\log q$ | $m^2\log q$ | $\|\mu\|+m(\|\mu\|+1)\log q$ | 是 | 标准模型 |
| 文献[13] | $(nm+(\|\mu\|+2)n^2k+n)\log q$ | $mnk\log q$ | $\|\mu\|+(m+2nk)\log q$ | 是 | 标准模型 |
| 文献[14] | $2nm\log q$ | $m^2\log 3$ | $\|\mu\|+m\log 3+m\log(12\sigma)$ | 否 | 随机预言机模型 |
| 文献[15] | $nm\log(2q)$ | $nm\log(2q)$ | $\|\mu\|+n\log 2+m\log(12\sigma)$ | 否 | 随机预言机模型 |
| 新方案 | $2nm\log q$ | $m^2\log 3$ | $\|\mu_2\|+\|\mu\|+m\log(12\sigma)$ | 否 | 随机预言机模型 |

**结束语** 本文在Lyubashevsky的无陷门签名方案的基础上，提出了一个高效的具有消息恢复功能的格签名方案，同时新方案可以看作是具有消息恢复功能Abe、Okamato的签名的格密码版本，能够抵抗未来量子计算机攻击。在随机预言机模型下，基于格上小整数解困难问题假设证明新方案是不可伪造的。新方案仅需要简单的矩阵与向量乘法运算，没有使用高斯原像抽样作为签名，提高了签名效率，同时缩短了传输消息-签名的总长度。另外，如何在标准模型下构造具有消息恢复功能的高效格签名方案是下一步值得研究的工作。

## 参考文献


[1] Boneh D, Lynn B, Shacham H. Short signatures from the weil pairing [J]. Journal of Cryptology, 2004, 17(4): 297-319

[2] Nyberg K, Rueppel R A. A new signature scheme based on the DSA giving message recovery [C] // CCS 1993. ACM, New York, 1993: 58-61

[3] Abe M, Okamoto T. A signature scheme with message recovery as secure as discrete logarithm [C] // ASIACRYPT 1999. LNCS 1716, Springer, Berlin, 1999: 378-389

[4] 陈辉焱，吕述望. 基于身份的具有部分消息恢复功能的签名方案[J]. 计算机学报，2006，29(9)：1622-1627

[5] ISO/IEC 9796-3: Information technology-Security techniques-Digital signature schemes giving message recovery-Part 3: Discrete logarithm based mechanisms(2nd Edition)[S]. JTC 1/SC 27. 2006

[6] ISO/IEC 9796-2: Information technology-Security techniques-Digital signature schemes giving message recovery-Part 2: Integer factorization based mechanisms(3nd Edition)[S]. JTC 1/SC 27. 2010

[7] Yang J H, Lin I C. A source authentication scheme based on message recovery digital signature for multicast[J]. International Journal of Communication Systems, 2013

[8] Ajtai M. Generating hard instances of lattice problems [C] // STOC 1996. ACM, New York, 1996: 99-108

[9] 王凤和，胡予濮，贾艳艳. 标准模型下的格基数字签名方案[J]. 西安电子科技大学学报，2012，39(4)：57-61

[10] 谢璇，喻建平，王廷，等. 基于格的变色龙签名方案[J]. 计算机科学，2013，40(2)：117-119

[11] Gentry C, Peikert C, Vaikuntanathan V. Trapdoors for hard lattices and new cryptographic constructions[C] // STOC 2008. ACM, New York, 2008: 197-206

[12] Cash D, Hofheinz D, et al. Bonsai trees, or how to delegate a lattice basis[C] // EUEOCRYPT 2010. LNCS 6110, Springer, Berlin, 2010: 523-552

[13] Micciancio D, Peikert C. Trapdoors for lattices: Simpler, tighter, faster, smaller[C] // EUROCRYPT 2012. LNCS 7237, Springer, Berlin, 2012: 700-718

[14] Lyubashevsky V. Lattice signatures without trapdoors [C] // EUROCRYPT 2012. LNCS 7237, Springer, Berlin, 2012: 738-755

[15] Ducas L, Durmus A, Lepoint T, et al. Lattice signatures and bimodal Gaussians [C] // Crypto 2013. LNCS 8042, Springer, Berlin, 2013: 40-56

[16] Bellare M, Neven G. Multi-signatures in the plain public-key model and a general forking lemma[C] // CCS 2006. ACM, New York, 2006: 390-399


(上接第157页)


[12] Rafaeli S, Hutchison D. A survey of key management for secure group communication[J]. ACM Computing Surveys (CSUR), 2003, 35(3): 309-329

[13] Naor D, Naor M, Lotspiech J. Revocation and tracing schemes for stateless receivers[M] // Advances in Cryptology-CRYPTO 2001. Springer Berlin Heidelberg, 2001: 41-62

[14] Beimel A. Secure Schemes for Secret Sharing and Key Distribution[D]. Israel Institute of Technology, Technion, Haifa, Israel, 1996

[15] Goldreich O, Goldwasser S, Micali S. How to Construct Random Functions[J]. JACM, 1986, 33(4): 792-807

[16] Yang Kan, Jia Xiao-hua, Kui Ren. Attributed-based fine-grained access control with efficient revocation in cloud storage systems [C] // Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security. ACM, New York, NY, USA, 2013: 523-528