# 文件分层-属性基多关键字可搜索加密方案

蒋 英,陈燕俐,高诗尧
(南京邮电大学 计算机学院、软件学院、网络空间安全学院,江苏 南京 210003)

**摘 要**:针对目前的文件分层的基于属性可搜索加密方案存在不支持 LSSS 访问结构以及多关键字密文较长等问题,提出了一个云环境下灵活高效、支持 LSSS 访问结构和文件分层的多关键字可搜索方案。首先通过将秘密值分配的方法,将下层的秘密值直接嵌入上层的密文中,不仅缩短了分层加密文件的密文长度,还提高了加密、解密效率;其次通过对多关键字构造索引向量的方式,解决了目前多关键可搜索方案中关键字密文长度随关键字个数线性增加的问题,实现了关键字密文长度固定的多关键字搜索,并采用先关键字搜索,再解密的方式进一步提高了关键字的搜索效率。最后通过将部分解密工作转移到云端,从而降低了用户的计算负担。基于 q-parallel BDHE 假设下证明了该方案可抵抗选择明文安全攻击(CPA),理论分析和实验结果证明了方案的有效性。

**关键词**:属性加密;云计算;分层结构;多关键字可搜索;访问结构
**中图分类号**:TP309 **文献标识码**:A **文章编号**:1673-629X(2021)10-0098-07
doi:10.3969/j.issn.1673-629X.2021.10.017

# File Hierarchy Attribute-based Multi-keyword Searchable Encryption Scheme

JIANG Ying,CHEN Yan-li,GAO Shi-yao
(School of Computer Science,Nanjing University of Posts and Telecommunications,Nanjing 210003,China)

**Abstract**:In view of the problems of the current file hierarchy attribute based searchable encryption scheme which does not support LSSS access structure and has long keyword ciphertext,a multi keyword searchable scheme supporting LSSS access structure and file hierarchy in cloud environment is proposed. Firstly,the secret value of the lower layer is directly embedded into the upper layer ciphertext by means of secret sharing,which not only shortens the length of the ciphertext,but also improves the encryption and decryption efficiency of the hierarchical encrypted files. Secondly,by constructing index vector of multi-keywords,it solves the problem that the ciphertext length of keywords in multi-keyword searchable scheme increases linearly with the number of keywords,realizes the fixed ciphertext length of keywords,and keyword search first and then decryption is adopted to further improve the efficiency of keyword search. Finally,by transferring part of the decryption work to the cloud,the computing burden of users is reduced. Based on the assumption of q-parallel BDHE,it is proved that this scheme can resist chosen plaintext attacks (CPA). Theoretical analysis and experimental results show the effectiveness of the scheme.

**Key words**:attribute based encryption;could computing;hierarchy structure;multi-keyword searchable;access structure

## 0 引 言

近年来,云技术得到了飞速发展,已成为计算机科学中的重要领域。云技术不仅提供计算服务,还提供存储服务,可以使用户受益于解决数据共享的爆炸性增长问题。文件属主(data owner,DO)的文件可以存储在多个服务器上,并由其他数据用户共享。为了确保存储在远程服务器上的文件不会被其他数据用户或恶意服务器破坏,DO 通常会在数据上传到云端之前对数据进行加密,但是加密可能会导致用户(data user,DU)无法对密文进行搜索,在某种程度上限制了文件搜索的灵活性。为了解决搜索加密文件的问题,专家们提出了可搜索加密(search encryption,SE)技术,DO 对共享数据和文件和关键词进行加密,并上传至服务器。SE 可充分利用云服务器庞大的计算资源进行密文上的关键字查找,不仅保证了用户数据的安全和隐私,而且能够节省大量的网络传输和计算开销。随后,大量关于可搜索加密的文章被提出,目前 SE 方案根据其构造算法的不同可以分为两类:对称可搜索

**收稿日期**:2020-12-01 **修回日期**:2021-04-06
**基金项目**:国家自然科学基金资助项目(61572263,61272084)
**作者简介**:蒋 英(1996-),女,硕士研究生,研究方向为基于属性的加密;陈燕俐,博士,教授,研究方向为信息安全。

加密算法[1]和公钥可搜索加密算法[2]。前者的构造通常基于一些伪随机函数生成器、哈希算法等,更适用于单用户模型;后者主要使用双线性映射等,并且将安全问题建立在一些复杂性假设上,更适用于多用户体制。另外还包括单关键字布尔搜索方案[3-5]和多关键字布尔搜索方案[6-9]。但传统的一对一的可搜索加密不能适应云计算海量用户和数据的安全管理以及细粒度的关键字搜索,云场景下,加密者往往想将数据分享给一些的特定的、满足一些固定条件的人,如何做到云存储数据的细粒度的加密数据访问控制和密文检索,成为了研究者面临的一个关键挑战。2005 年 Sahai 和 Waters[10]提出了基于属性加密(attributed - based encryption,ABE)机制,通过引入了属性集合和访问策略的概念,将一系列的属性集合看作用户的身份标识,当属性和访问结构相匹配时,才能解密密文。2013 年,Kulvaibhavh 等人[11]首先提出了基于属性的可搜索加密方案(attribute-based search encryption,ABSE),将灵活、高效和细粒度的 ABE 加密机制和可搜索加密技术相结合,实现了云环境安全、高效、细粒度的数据共享以及密文检索。

考虑到在多数云计算实际应用环境中,共享的多个文件通常具有层次结构。以个人健康记录(personal health records,PHR)为例[12],为了在云环境中安全地共享 PHR 信息,患者将其 PHR 信息 $M$ 分为两个部分:个人信息文件 $m_1$ 和病例信息文件 $m_2$。患者根据实际需要,会通过不同的访问策略对文件 $m_1$ 和 $m_2$ 进行加密。设患者将 $m_1$ 的访问结构设置为 $P_1$:{(心脏病学 AND 研究员) AND 主治医师},$m_2$ 的访问结构被设置为 $P_2$:{心脏病学 AND 研究员}。显然,如果分别使用访问结构 $P_1$ 和 $P_2$ 对 $m_1$ 和 $m_2$ 进行加密,文件需要进行两次加密,部分密文会出现重复。考虑到通常情况下,当用户可以解密 $m_1$ 时,他必然可以解密 $m_2$,Wang 等人首次提出了文件分层-属性加密方案(file hierarchy attribute-based encryption,FH-ABE)[13],通过引入层节点和传输节点等概念,将多个访问策略集成为一个访问结构,缩短了密文长度,同时用户可以解密当层以及以下层的所有文件,从而提高了解密效率。为了实现分层文件上的可搜索加密,Miao 等人[5]将 FH-ABE 方案与 SE 方案相结合,提出了文件分层的基于属性的可搜索加密(file hierarchy attribute-based searchable encryption,FH-ABSE)等方案。然而,目前的 FH-ABE 和 FH-ABSE 方案只支持树形访问结构,解密运算由于采用递归和拉格朗日差值计算,计算效率低,并且需要逐层计算节点的信息,解密开销较大。同时目前的 FH-ABSE 方案还存在以下两个问题:(1)采用先解密,再进行关键字匹配的方式,造成关键字搜索效率较低;(2)索引密文与关键字个数相关,当关键字较多时,存储开销和计算开销较高。针对以上问题,文中以 Wang 等人的方案[13]为基础,构造了一个灵活的、高效的、支持 LSSS 访问结构的多关键字可搜索方案(file hierarchy-LSSS-attribute based multi-keyword search encryption,FH-LABMKSE)。创新点具体如下:

(1)实现了支持 LSSS 结构的文件分层-基于属性的加密。文中支持 LSSS 结构,通过将秘密因子逐步分配的方法,巧妙地避开了采用访问树结构方案在解密时的递归和拉格朗日差值计算。另外将下一层的密文信息直接嵌入上一层的密文中,实现了密文的跳跃式传递。同时,也实现了灵活的细粒度访问控制。

(2)实现了固定索引密文长度的多关键字可搜索。本方案关键字索引大小与计算开销都与关键字个数无关,和目前的多关键字可搜索方案中关键字密文大小和计算开销随着关键字个数线性增长相比,在固定了关键字密文长度的同时,还提高了计算效率。另外本方案采用先关键字匹配,然后符合访问结构的用户再解密的方式,和目前先解密再关键字匹配的方案相比,提高了关键字的搜索效率。

(3)不仅关键字搜索功能是由云服务器完成的,并且将解密的一部分计算任务转移到云服务器,从而降低了用户的计算负担,而云服务器在整个过程中不会得到和关键字有关的有用信息。

## 1　相关工作

### 1.1　分层的基于属性的加密

为了在加密文件上实现分层的细粒度访问结构,Wang 等人首次提出了一个分层 CP-ABE 方案[14]。随后,出现了一系列具有特殊功能的分层 CP-ABE 方案[15-16]。但是,它们仅仅考虑了属性的分层,却没有考虑文件的分层。为了支持对分层文件的访问控制,Wang 等人[13]提出了 FH-CP-ABE 方案,通过将多个访问树集成为一个访问树,用户可以通过计算一次解密密钥来解密多个文件,既节省了密文存储空间又节省了加解密时间。并且,当系统的文件越多,方案的效率越高。2019 年,Fu 等人[17]提出了实用的密文策略-基于属性的文件分层加密方案。该方案首先根据文件的属性构造了一组相互独立的访问树,其次采用贪婪策略来构建合并这些树,然后通过合并各个小树来动态生长树。将集成访问树中的所有文件加密在一起,提高了加解密的效率。

### 1.2　基于属性的可搜索加密

2013 年,Wang 等人[18]结合 CP-ABE 和公钥可搜索技术,提出基于属性的可搜索加密方案。2017 年,Gowda 等[19]提出了文件分层的可搜索加密方案,该算

法具有定时启用的隐私保护关键字搜索机制。提出的方案允许数据用户基于属性集在数据拥有者提供的持续时间内执行关键字搜索。它支持在数据拥有者提供的指定时间后自动撤销访问权限。它保证了机密性,同时保持了细粒度的访问控制和通用的客户端撤销。Miao 等人提出文件分层的属性可搜索方案[5]支持细粒度的单关键字搜索和细粒度的多关键字搜索。但是,目前多关键字可搜索方案仅支持访问树结构,并且密文多数都是随着关键字的增加而线性增长,当关键字较多时,存储开销较大。

## 2 预备知识

定义 1 双线性映射[10]。设置两个阶为素数 $p$ 的乘法循环群 $G_0$ 和 $G_T$, $g$ 是 $G_0$ 的一个生成元。存在一个双线性映射 $e:G_0 \times G_0 \rightarrow G_T$,必须满足以下 3 个条件:

(a)双线性。对任意 $u,v \in G_0$,任意 $a,b \in Z_p$,都有 $e(u^a,v^b)=e(u,v)^{ab}$。

(b)非退化性。存在 $u,v \in G_0$,满足 $e(u,v) \neq 1$。

(c)可计算性。对任意 $u,v \in G_0$, $e(u,v)$ 可有效计算。

定义 2 线性秘密共享矩阵(LSSS)[20]。令 $\{P_1,P_2,\cdots,P_n\}$ 为 $n$ 个参与者的集合,这一组参与者上的秘密共享方案 $\pi$ 是线性的,如果:

(a)各方的秘密因子构成了 $Z_p$ 上的向量。

(b)存在一个 $l$ 行 $n$ 列的矩阵 $\boldsymbol{M}$,称为 $\pi$ 的共享生成矩阵。对于所有 $i=1,2,\cdots,l$, $\boldsymbol{M}$ 的第 $i$ 行为 $\boldsymbol{M}_i$,令函数 $\rho$ 将参与者所在的行 $i$ 定义为 $\rho(i)$。考虑列向量 $v=(s,r_2,\cdots,r_n)$,其中 $s \in Z_p$ 是要共享的秘密值,$r_2,\cdots,r_n \in Z_p$ 是随机选择的,则 $\boldsymbol{M}\cdot\boldsymbol{v}$ 是 $l$ 个秘密因子的向量。每个秘密因子 $\lambda_i=\boldsymbol{M}_i\cdot\boldsymbol{v}$ 属于参与者 $\rho(i)$。

在方案[21]中表明,根据上述定义,每个线性秘密共享方案也都具有线性重构属性,定义如下:假设 $\pi$ 是访问结构 $A$ 的 LSSS。$S \in A$ 为任意授权集合,令 $I \in \{1,2,\cdots,l\}$ 定义为 $I=\{i:\rho(i) \in S\}$。存在常数 $\{\omega_i \in Z_p\}_{i\in I}$,满足 $\sum_{i\in I}\omega_i\cdot\boldsymbol{M}_i=(1,0,\cdots,0)$,使得如果 $\{\lambda_i\}$ 是根据 $\pi$ 的任何秘密值的有效秘密因子,则有 $\sum_{i\in I}\omega_i\cdot\lambda_i=s$

定义 3 判定平行双线性 Diffie-Hellman 指数问题(decision parallel bilinear Diffie-Hellman exponent problem,q-parallel BDHE)[14]。根据系统安全参数选择一个阶为素数 $p$ 的乘法循环群 $G_0$, $g$ 是 $G_0$ 的一个生成元。随机选择 $a,s,b_1,\cdots,b_q \in Z_p$,假设 q-parallel BDHE 问题是给定多元组: $y=(g,g^s,g^a,\cdots,g^{(a^{2q})},\forall 1\leqslant j\leqslant q\, g^{s\cdot b_j},\ g^{\frac{s}{b_j}},\cdots,g^{\frac{a^{2q}}{b_j}},\forall 1\leqslant j\leqslant q,k\neq j\, g^{\frac{a\cdot s\cdot b_k}{b_j}},\ \cdots,\ g^{\frac{a^q\cdot s\cdot b_k}{b_j}})$。如果不存在一种算法能够在多项式时间内以不可忽略的概率区分 $e(g,g)^{a^{q+1}s} \in G_T$ 和 $R \in G_T$,则 q-parallel BDHE 假设成立。

## 3 FH-LABMKSE 方案的安全模型

本方案的安全模型与原始的 LSSS-CP-ABE 方案[20]相同。文中在敌手 A 和挑战者 B 之间定义 CPA(chosen plaintext attacks)安全游戏,如下所示。

系统建立。B 运行 FH-LABMKSE 的 Setup 阶段,输出公开参数 PK,并将 PK 发送给 A。

询问阶段 1。A 重复地对属性集 $S_1$, $S_2$,…, $S_{q_1}$ 进行私钥查询。

挑战。A 选择两个要受到挑战的等长的信息 $m_0$ 和 $m_1$。同时,A 给定一个受到挑战的访问结构 $A^*$ 使得没有任何询问阶段 1 的集合 $S_i$ 满足 $A^*$。B 选择一条信息 $m_\rho$, $\rho \in \{0,1\}$,并且用 $A^*$ 加密。B 将密文 $CT^*$ 返回给 A。

询问阶段 2。A 重复询问阶段 1 的步骤,但是,属性集 $S_{q_1+1}$,…, $S_q$ 都不满足与挑战相对应的访问结构。

猜测阶段。A 输出一个 $\rho$ 的猜想 $\rho'$,若 $\rho=\rho'$,则攻击者 A 赢得游戏。上述游戏中的优势 A 可以描述为 $\mathrm{Adv}_A^{CCA}(1^\lambda)=|\Pr[\rho=\rho']-\frac{1}{2}|$。

定义 4 如果没有敌手能够在多项式时间内以不可忽略的优势赢得上述安全游戏,则 FH-LABMKSE 方案是 CPA 安全的。

## 4 FH-LABMKSE 方案

### 4.1 系统模型

设关键字集 $W=\{w_1,w_2,\cdots,w_m\}$,DO 首先从文件集 $F=\{F_1,F_2,\cdots,F_k\}$ 中提取关键字,并为每个文件构造关键字索引向量 $D_i=\{d_{i,1},d_{i,2},\cdots,d_{i,m}\}$(若 $d_{i,\theta}$ 为 1,表示 $F_i$ 中有第 $\theta$ 个关键字,否则为 0)。接着,DO 利用不同的对称密钥 $ck_i$ 分别加密文件 $F_i$,并加密对称密钥集 $ck=\{ck_1,ck_2,\cdots,ck_k\}$ 和关键字索引向量。当 DU 想要访问包含其预计关键字的密文时,必须将根据查询的关键字生成的陷门传送给云服务提供商(cloud server provider,CSP)。此后,当且仅当其陷门与访问结构匹配时,CSP 才会返回相关的密文。

系统模型如图 1 所示,文中系统主要涉及四种不同类型的实体:文件属主(DO)、用户(DU)、云服务提供商(CSP)和授权中心(attribute authority,AA)。假设 DO 拥有 $k$ 个文件,并且文件集 $F=\{F_1,F_2,\cdots,F_k\}$ 在云存储中共享。AA 是一个完全受信任的实体,并

且接受云储存中的用户注册,为用户生成属性私钥。CSP 是一个半可信的实体。CSP 会好奇存储在云上的数据,但是绝对忠诚,会严格履行特定的服务,不会恶意删除数据或者拒绝响应用户的请求。在本系统它提供密文存储、关键字检索和部分解密工作。

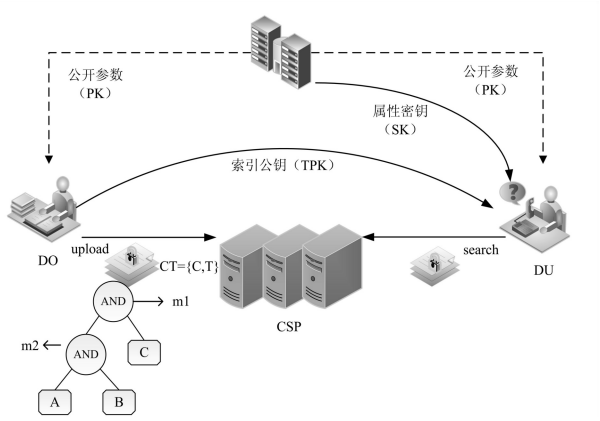


图 1　系统模型

### 4.2　FH-LABMKSE 方案具体构造

(a)初始化算法 Setup 。

由 AA 执行,输入安全参数 $\lambda$ 。生成阶为素数 $p$ 的乘法循环群 $G$ 、$G_T$ 。给定双线性映射 $e:G\times G\to G_T$ , $g$ 为 $G$ 的生成元。设系统中属性集合 $U$ ,系统属性个数为 $|U|$ 。随机生成群元素 $h_{\mathrm{attr}(1)},\cdots,h_{\mathrm{attr}(|U|)}\in G$ 。AA 定义两个哈希函数 $H:\{0,1\}^*\to G$ , $H_1:\{0,1\}^*\to Z_p$ 。最后,随机选择 $\alpha,a,b\in Z_p$ ,生成 PK 和 MSK , PK 发布在 AA 的公共布告板上,MSK 自行秘密保存。

$$\mathrm{PK}=\{e(g,g)^{\alpha},g,g^{a},g^{b},h_{\mathrm{attr}(1)},\cdots,h_{\mathrm{attr}(U)},H,H_1\} \tag{1}$$

$$\mathrm{MSK}=\{g^{\alpha},b\} \tag{2}$$

(b)用户密钥生成算法 Kengen 。

由 AA 执行,当用户加入时,根据用户属性集 $S$ , AA 选择一个随机值 $t\in Z_p$ ,生成 DU 的属性密钥 SK ,并传送给用户。

$$\begin{aligned}&\mathrm{SK}=\{K=g^{\alpha}g^{at},L_1=g^{t},L_2=g^{1/b};\\&\forall\,\mathrm{attr}(i)\in S,K_{\mathrm{attr}(i)}=h_{\mathrm{attr}(i)}^{t}\}\end{aligned} \tag{3}$$

(c)索引密钥生成算法 TKengen 。

用户随机选择 $z\in Z_p$ ,生成一对索引公私钥对, $\mathrm{TPK}=e(g,g)^{z}$ 为索引公钥, $\mathrm{TSK}=z$ 为索引私钥。

(d)关键字和文件加密算法 Encrypt 。

由 DO 执行,假设文件集 $F=\{F_1,F_2,\cdots,F_k\}$ ,对称密钥集 $\mathrm{ck}=\{\mathrm{ck}_1,\mathrm{ck}_2,\cdots,\mathrm{ck}_k\}$ 分别用来加密 $k$ 个文件,文件等级由 1 到 $k$ 递减。

DO 采用一个分层的访问结构加密 $k$ 个对称密钥。在分层访问结构中,每一层只加密一个对称密钥。分层的访问结构实际上是多个访问策略的聚类,它包含很多个访问策略,每个访问策略对应一个对称密钥。这些访问策略存在从属关系,即 $P_k\subset P_{k-1}\subset\cdots\subset P_1$。若 DU 能够解密 $P_i$ ,那它必然可以解密 $P_{i+1}$, $i\in[1,k-1]$ 。DO 根据 $P_1$,生成一个访问矩阵 $(\boldsymbol{M},\rho)$ , $\boldsymbol{M}$ 是 $l\times n$ 的矩阵, $l$ 是 $P_1$ 中的属性个数。选择一个向量 $v=\{s_1,y_2,\cdots,y_n\}\in Z_p$ , $s_1$ 是加密 $\mathrm{ck}_1$ 的秘密因子。DO 通过计算 $\boldsymbol{v}\times\boldsymbol{M}_i$ 得到每个属性对应的秘密因子 $\lambda_i$ ,其中 $\boldsymbol{M}_i$ 是矩阵 $\boldsymbol{M}$ 的第 $i$ 行向量。根据文献[21]中的访问矩阵构成方式,不难看出,或门的孩子节点的秘密因子等于该或门节点的秘密因子,而无论一个与门节点有多少孩子节点,它的秘密因子都等于它的左孩子节点的秘密因子减去右边所有孩子节点秘密因子的和。因此当 DO 得到每个属性的秘密因子和 $s_1$ 后,便可自上而下地重构每个子访问策略 $P_i$ 的秘密值 $s_i$ , $i\in[2,k]$ 。

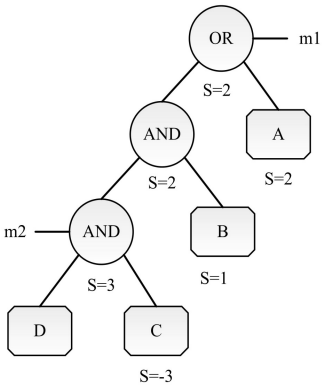


图 2　分层访问结构中秘密因子的分配

如图 2 所示,假设第一个文件 $m_1$ 的访问策略为 $P_1=((D\wedge C)\wedge B)\vee A$ ,第二个文件 $m_2$ 的访问策略为 $P_2=(D\wedge C)$ ,显然 $P_2\subset P_1$。根据 $P_1$ 构造出的 LSSS 访问矩阵 $\boldsymbol{M}$ 是 4 × 3 的矩阵,根据文献[22], $P_1$ 对应的 LSSS 矩阵 $(\boldsymbol{M},\rho)=(\begin{bmatrix}1&1&1\\0&0&1\\0&1&0\\1&0&0\end{bmatrix}\begin{matrix}D\\C\\A\\B\end{matrix})$ 。

为 $P_1$ 随机选择一个秘密值 $s_1=2$,接着选择两个随机数,设 $y_2=1$, $y_3=-3$,构成向量 $v=(2,1,-3)$ ,图 2 给出了各节点分配的秘密因子,可知 $P_2$ 的秘密值为 3。

DO 随机选择 $r_1,r_2,\cdots,r_l\in Z_p$ ,生成文件密文和关键字密文的步骤如下:

• 文件密文的生成。

用对称密钥 $\mathrm{ck}_i$ 加密每个文件 $F_i$ ,得到数据密文 $\mathrm{CF}_i=\mathrm{Enc}(F_i,\mathrm{ck}_i)$ (此处 Enc 是一种对称加密算法)。

再用文件分层的属性加密机制对 $\mathrm{ck}_i$ 进行加密,生成密文如下:

$$C_i = \mathrm{ck}_i e(g,g)^{\alpha s_i}, C_i' = g^{s_i}, i \in [1,k] \quad (4)$$

$$\Lambda_i = e(g,g)^{\alpha(s_i+s_{i+1})} \cdot H(e(g,g)^{\alpha s_i}) \quad i \in [1,k-1] \quad (5)$$

$$\mathbb{C}_x = g^{a\lambda_x} h_{\rho(x)}^{-r_x}, \mathbb{C}_x' = g^{r_x}, x \in [1,l] \quad (6)$$

得到与文件有关的密文 $C = \{\mathrm{CF}_i, C_i, C_i', \Lambda_i, \mathbb{C}_x, \mathbb{C}_x'\}$。

• 关键字密文的生成。

DO 根据每个文件的关键字索引向量 $D_i = \{d_{i,1}, d_{i,2}, \cdots, d_{i,m}\}$ 以及自己的索引私钥 $z$，为每个文件 $F_i$ 生成关键字密文 $I = \{I_i\}$：

$$I_i = g^{bz\prod_{\theta=1}^{m} H_1(\theta)^{d_{i,\theta}}}, i \in [1,k] \quad (7)$$

最后，DO 将密文 $\mathrm{CT} = \{C, I\}$ 上传到 CSP。

(e)陷门生成算法 Trapgen。

DU 执行该算法。$W'$ 为 DU 的查询关键字集。DU 首先向 DO 申请获得索引公钥 $e(g,g)^z$，再根据 $W'$ 生成查询向量 $Q = \{q_1, q_2, \cdots, q_m\}$，若关键字 $w_\theta$ 在 $W'$ 中，那么 $q_\theta$ 为 1，否则为 0。DU 随机选择 $v \in Z_p$，生成陷门 $T$。

• DU 生成与密钥相关的陷门组件 $T_1$：

$$T_1 = \{L_1^* = g^{tv}, \forall \mathrm{attr}(i) \in S, K_{\mathrm{attr}(i)}^* = h_{\mathrm{attr}(i)}^{tv}\} \quad (8)$$

• DU 生成与关键字相关的陷门组件 $T_2$：

$$T_2 = \{t_1 = g^{\frac{v}{b(\prod_{\theta=1}^{m} H_1(\theta))^{q_\theta}}}, t_2 = e(g,g)^{tz}\} \quad (9)$$

最后 DU 将完整的陷门 $T = \{T_1, T_2\}$ 发送给 CSP。

(f)搜索算法 Search。

• CSP 首先验证 $e(I,t_1) \geqslant t_2$ 是否成立：

$$e(I,t_1) = e(g,g)^{vz\prod_{\theta=1}^{m} H_1(\theta)^{d_{i,\theta}-q_\theta}} \quad (10)$$

如果 $Q \subseteq D_i$，那么 $d_{i,\theta} \geqslant q_\theta$，则 $\prod_{\theta=1}^{m} H_1(\theta)^{d_{i,\theta}-q_\theta}$ 为正整数，即 $e(I,t_1) \geqslant t_2$ 成立，说明搜索到包含查询关键字的文件，进行下一步操作。如果不成立，输出 $\perp$ 表示关键字搜索失败。

• CSP 检查 DU 的属性集 $S$ 是否满足或部分满足访问策略，如果不满足，则输出 $\perp$ 表示失败。如果 $S$ 满足策略 $P_i$，定义 $I \subset \{1,2,\cdots,l\}$ 为 $I = \{x : \rho(x) \in S\}$。然后，令 $\{\omega_x \in Z_p\}, x \in I$，是一组常数，使得 $\sum_{x \in I} \omega_x \cdot \lambda_x = s_i$。计算：

$$A_i = \prod_{x \in I} (e(\mathbb{C}_x, L_1^*) e(\mathbb{C}_x', K_{\mathrm{attr}(i)}^*))^{\omega_x} = e(g,g)^{atvs_i} \quad (11)$$

CSP 将位于 $i$ 层以及 $i$ 层以下的密文 $\mathrm{CT}' = (\Lambda_\sigma, C_i', A_i, \mathrm{CF}_i), \sigma \in [i, k-1]$ 发送给 DU。

(g)解密算法。

Decrypt(CT′,SK)：DU 根据 CT′ 和 SK 计算：

$$F_i = \frac{e(K, C_i')}{A_i^{1/v}} = e(g,g)^{\alpha s_i} \quad (12)$$

基于分层的特性，DU 也能够解密位于 $i$ 层以下的密文：

$$F_{i+1} = \frac{\Lambda_i}{F_i \cdot H(F_i)} = e(g,g)^{\alpha s_{i+1}} \quad (13)$$

最后获得相应的对称密钥：

$$\mathrm{ck}_i = \frac{C_i}{F_i} = \frac{\mathrm{ck}_i e(g,g)^{\alpha s_i}}{e(g,g)^{\alpha s_i}} \quad (14)$$

从而解密 $\mathrm{CF}_i$ 获得文件集 $F$。

### 4.3 安全性证明

定理 1：如果 q-parallel-BDHE 假设成立，那么攻击者不可能在多项式时间内找到一个不可忽略的概率以大小为 $l^* \times n^*$ 的挑战矩阵，选择性地攻破文中方案，其中 $l^*, n^* \leqslant q$，即提出的方案在随机预言模型下是选择性 CPA 安全的。

证明：假定在选择安全性的情况下有多项式时间敌手 A 可以有不可忽略的优势攻破本方案，那么敌手 A 可以构建出一个挑战者 B 以不可忽略的优势解决 q-parallel BDHE 的问题。具体过程如下：

(a)初始化。

挑战者 B 接受 q-parallel-BDHE 挑战向量 $y$ 和随机数 $T$。A 选择挑战访问策略 $(\boldsymbol{M}^*, \rho^*)$，$\boldsymbol{M}^*$ 有 $n^*$ 列，并将其发送给挑战者 B。

(b)系统建立。

B 随机选择 $\alpha' \in Z_p$，令 $\alpha = a^{q+1} + \alpha'$，使得 $e(g,g)^\alpha = e(g^a, g^{a^q}) e(g,g)^{\alpha'}$。模拟 B 编译参数 $h_1, \cdots, h_U$，对所有的 $\mathrm{attr}(i) \in U$，选择一个随机的 $\zeta_{\mathrm{attr}(i)} \in Z_p$。令 $X$ 表示索引 $x$ 的集合，即 $\rho^*(x) = \mathrm{attr}(i)$。B 设置 $h_{\mathrm{attr}(i)} = g^{\zeta_{\mathrm{attr}(i)}} \prod_{x \in X} g^{\frac{aM_{x,1}^*}{b_x}} \cdot g^{\frac{a^2 M_{x,2}^*}{b_x}} \cdots \cdot g^{\frac{a^{n^*} M_{x,n^*}^*}{b_x}}$。若 $X = \varnothing$，则 $h_{\mathrm{attr}(i)} = g^{\zeta_{\mathrm{attr}(i)}}$。

(c)阶段 1。

在此阶段，B 响应私钥查询。假设为 B 提供了一个针对集合 $S$ 的私钥查询，其中 $S$ 不满足 $\boldsymbol{M}^*$。B 选择随机数 $r \in Z_p$，找到一组向量 $\boldsymbol{\omega} = (\omega_1, \omega_2, \cdots, \omega_{n^*}) \in Z_p^{n^*}$ 使得 $\omega_1 = -1$。并且对所有的 $x$，其中 $\rho^*(x) \in S$，都有 $\boldsymbol{\omega} \cdot \boldsymbol{M}_x^* = 0$。根据 LSSS 的定义，这样的向量必然存在。

B 通过将 $L$ 设置为 $L = g^r \prod_{x=1,2,\cdots,n^*} (g^{a^{q+1-x}})^{\omega_x} = g^t$，将 $t$ 隐式地定义为 $t = r + \omega_1 a^q + \omega_2 a^{q-1} + \cdots + \omega_{n^*} a^{q-n^*+1}$。

根据 $t$ 的定义，可以发现 $g^{at}$ 包含一个 $g^{-a^{q+1}}$ 项，当创建 $K$ 时，它将与 $g^a$ 中的未知项抵消。B 计算 $K$ 为 $K = g^{\alpha'} g^{ar} \prod_{x=2,\cdots,n^*} (g^{a^{q+2-x}})^{\omega_x}$。接下来必须计算 $K_{\mathrm{attr}(i)}$。首先考虑 $\mathrm{attr}(i) \in S$，但是其中不存在 $x$ 使得 $\rho^*(x) =$

attr($i$)。对于这些属性,可以简化为 $K_{\text{attr}(i)} = L^{\zeta_{\text{attr}(i)}}$。对于在访问结构中的属性,B 构造 $K_{\text{attr}(i)}$ 为: $K_{\text{attr}(i)} = L^{\zeta_{\text{attr}(i)}} \prod_{x \in X} \left( \prod_{j=1,\cdots,n^*} (g^{(\frac{a^j}{b_x})r}) \prod_{k=1,\cdots,n^*,k\neq j} (g^{\frac{a^{q+1+j-k}}{b_x}})^{\omega_k} \right)^{M^*_{x,j}}$。

(d)挑战。

敌手 A 将信息 $m_0$ 和 $m_1$ 发送给挑战者。B 随机选择一条信息 $m_\rho, \rho \in \{0,1\}$,创建 $C = m_\rho T \cdot e(g^s, g^{\alpha'})$, $C' = g^s$。B 随机选择 $y'_2, \cdots, y'_{n^*} \in Z_p$,通过向量 $v$ 共享秘密值 $v = (s, sa + y'_2, sa^2 + y'_3, \cdots, sa^{n^*-1} + y'_{n^*}) \in Z_p^{n^*}$。同时选择随机值 $r'_{\text{attr}(1)}, \cdots, r'_{\text{attr}(l)} \in Z_p$。对 $x = 1, \cdots, n^*$,将 $R_x$ 定义为所有 $k \neq x$ 的集合,使得 $\rho^*(x) = \rho^*(k)$。挑战密文组件被生成为 $D_x = g^{-r'_x} g^{-sb_x}$, $C_x = h^{r'_x}_{\rho^*(x)} \left( \prod_{j=2,\cdots,n^*} (g^a)^{M^*_{x,j} y'_j} \right) (g^{b_x \cdot s})^{-\zeta_{\rho^*(x)}} \cdot \left( \prod_{k \in R, j=1,\cdots,n^*} (g^{a^j \cdot s \cdot (\frac{b_x}{b_k})})^{M^*_{k,j}} \right)$。

(e)阶段 2。

与阶段 1 相同。

(f)猜测。

A 输出对随机数 $\rho$ 的猜想 $\rho'$,若 $\rho = \rho'$,则 B 输出 0 来猜测 $T = e(g,g)^{a^{n+1}s}$;否则输出 1,表示它认为 $T$ 是 $G_T$ 中的随机元素。当 $T$ 是群 $G_T$ 中的一个随机元素,有:

$$\left| \Pr[B(y, T = e(g,g)^{a^{n+1}s}) = 0] = \frac{1}{2} + \text{Adv}_A^{\text{CCA}}(1^\lambda) \right| 。$$

当 $T$ 是随机群元素时,信息 $m_\rho$ 对攻击者完全隐藏。有 $\Pr[B(y, T = R = 0] = \frac{1}{2}$。因此,B 具有不可忽略的优势来玩 q-parallel BDHE 游戏。

## 5　性能分析

文中方案与 Miao 的两个方案[5,8]进行了存储、通信开销和计算开销的比较。方案[8]实现了树形访问结构的基于属性的多关键字可搜索加密,方案[5]实现了树形访问结构的、文件分层-基于属性的多关键可搜索加密。在文献[5]中,密文信息的传递是由传输节点进行的。文献[5]中对传输节点的定义为:如果一个节点的孩子中至少有一个包含门限值,那么就称它为传输节点。

令 $L_G$, $L_{G_T}$ 分别表示 $G$、$G_T$ 中元素的长度, $|A_{le}|$, $|A_{nt}|$ 分别表示访问结构中属性的数量和传输节点的数量。令 $|S|$ 表示用户的属性数量, $U$ 表示系统中属性的个数。$m$ 表示关键字的个数。表 1 比较了 3 种方案公开参数大小、主密钥大小、用户属性密钥大小、索引密文大小以及陷门大小。从表中可以看出,文中方案的关键字索引密文长度小于方案[5]和方案[8]的。这是由于本方案的关键字索引密文与关键字的个数无关,是固定长度的,而另外两个方案都随着关键字数量的增加线性增长。

表 1　与不同方案的存储、通信开销的比较

| 指标 | Miao 的方案[8] | Miao 的方案[5] | FH-LABMKSE |
|---|---|---|---|
| 公开参数大小 | $4L_G$ | $5L_G + L_{G_T}$ | $(3+U)L_G + L_{G_T}$ |
| 主密钥大小 | $5L_G$ | $5L_G$ | $2L_G$ |
| 属性密钥大小 | $(2\|S\|+1)L_G$ | $(2\|S\|+2)L_G$ | $(\|S\|+2)L_G + L_{G_T}$ |
| 关键字密文大小 | $(3+m+2\|A_{le}\|)L_G$ | $(2+m+2\|A_{le}\|)L_G$ | $L_{G_T}$ |
| 陷门大小 | $(2\|S\|+3)L_G$ | $(2\|S\|+4)L_G$ | $(\|S\|+2)L_G + L_{G_T}$ |

表 2　与不同方案的计算开销的比较

| 指标 | Miao 的方案[8] | Miao 的方案[5] | FH-LABMKSE |
|---|---|---|---|
| 密钥生成时间 | $(2\|S\|+2)E_G$ | $(2\|S\|+3)E_G$ | $(4+\|S\|)E_G$ |
| 加密时间 | $(3k+km+2k\|A_{le}\|)E_G$ | $(2k+2km+2\|A_{le}\|)E_G + (k+2j\|A_{nt}\|)E_{G_T}$ | $(2k+3\|A_{le}\|)E_G + (3k-2)E_{G_T}$ |
| 陷门生成时间 | $(2\|S\|+3)E_G$ | $(2\|S\|+4)E_G$ | $(\|S\|+2)E_G + E_{G_T}$ |
| 搜索时间 | $(2\|A_{le}\|+3)e + 2E_{G_T}$ | $(2\|A_{le}\|+3)e + 2E_{G_T}$ | $(2\|A_{le}\|+1)e + E_{G_T}$ |

对计算开销的比较,为了方便,本节主要考虑了耗时的指数运算和双线对运算,令 $E_G$、$E_{G_T}$ 表示群 $G$、$G_T$ 的指数运算时间; $e$ 表示双线性配对计算时间。表 2 给出了在 $k$ 个文件、$m$ 个关键字的条件下三种方案的 Kengen 算法、Encrypt 算法、Trapgen 算法和 Search 算法的计算开销。$j$ 表示每个传输节点的子节点数。从表 2 中可以看出,文中方案的加密时间随 $k$ 线性增长,另外两个方案均随 $km$ 线性增加。同时,因为文中取消了传输节点,因此加密计算开销明显低于其余两个方案。

## 6　结束语

文中实现了一种文件分层的多关键字可搜索方案,在云环境中实现了安全灵活的文件分层的细粒度

访问控制和关键字搜索。方案通过将秘密值分配的方法,将下一层的秘密值直接嵌入上一层的密文中,提高了分层文件的解密效率;其次通过对多关键字构造索引向量的方式,实现了固定关键字密文长度的多关键字可搜索,最后采用先关键字搜索,再解密的方式进一步提高了关键字的搜索效率。该方案不仅提高了解密效率,而且节省了通信和存储开销。

**参考文献:**


[1] SONG D X, WAGNER D, PERRIG A. Practical techniques for searches on encrypted data[C]//Proceedings of 2000 IEEE symposium on security and privacy. Berkeley, CA: IEEE, 2000: 44-55.

[2] BONEH D, CRESCENZO G D, OSTROVSKY R, et al. Public key encryption with keyword search[C]//Advances in cryptology — EUROCRYPT 2004. Interlaken, Switzerland: Springer, 2004: 506-522.

[3] 陈燕俐,杨华山. 可支持属性撤销的基于 CP-ABE 可搜索加密方案[J]. 重庆邮电大学学报:自然科学版, 2016, 28(4): 545-554.

[4] LI J, SHI Y, ZHANG Y. Searchable ciphertext-policy attribute-based encryption with revocation in cloud storage[J]. International Journal of Communication System, 2017, 30(1): 1-13.

[5] MIAO Y, MA J, LIU X, et al. Attribute-based keyword search over hierarchical data in cloud computing[J]. IEEE Transactions on Service Computing, 2020, 13(6): 985-998.

[6] DAN B, WATERS B. Conjunctive, subset, and range queries on encrypted data[C]//Theory of cryptography. Amsterdam, The Netherlands: Springer, 2007: 535-554.

[7] 曹来成,王纬婷,康一帆,等. 属性盲化的模糊可搜索加密云存储方案[J]. 北京理工大学学报, 2019, 39(7): 706-713.

[8] MIAO Y, MA J, LIU X, et al. Practical attribute based multi-keyword search scheme in mobile crowdsourcing[J]. IEEE Internet of Things Journal, 2018, 5(4): 3008-3018.

[9] 朱智强,苏 航,孙 磊,等. 云存储中基于属性的关键词搜索加密方案研究[J]. 网络与信息安全学报, 2017, 3(11): 1-11.

[10] SAHAI A, WATERS B. Fuzzy identity-based encryption[C]//Advances in cryptology - EUROCRYPT 2005. Aarhus, Denmark: Springer, 2005: 457-473.

[11] KAUSHIK K, VARADHARAJAN V, NALLUSAMY R. Multi-user attribute based searchable encryption[C]//Proceeding of international conference on mobile data management. Milan: IEEE, 2013: 200-205.

[12] XHAFA F, WANG J, CHEN X, et al. An efficient PHR service system supporting fuzzy keyword search and fine-grained access control[J]. Soft Computing, 2014, 18(9): 1795-1802.

[13] WANG S, ZHOU J, LIU J K, et al. An efficient file hierarchy attribute-based encryption scheme in cloud computing[J]. IEEE Transactions on Information Forensics and Security, 2016, 11(6): 1265-1277.

[14] WANG G, LIU Q, WU J, et al. Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers[J]. Computers and Security, 2011, 30(5): 320-331.

[15] WAN Z, LIU J, DENG R H. HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing[J]. IEEE Transactions on Information Forensics and Securit, 2012, 7(2): 743-754.

[16] DENG H, WU Q, QIN B, et al. Ciphertext-policy hierarchical attribute-based encryption with short ciphertexts[J]. Information Science, 2014, 275: 370-384.

[17] FU J S, WANG N. A practical attribute-based document collection hierarchical encryption scheme in cloud computing[J]. IEEE Access, 2019, 7: 36218-36232.

[18] WANG C, LI W, LI Y, et al. A ciphertext-policy attribute-based encryption scheme supporting keyword search function[C]//Cyberspace safety and security. Zhangjiajie, China: Springer, 2013: 377-386.

[19] GOWDA B K, SUMATHI R. Hierarchy attribute-based encryption with timing enabled privacy preserving keyword search mechanism for e-health clouds[C]//2th IEEE international conference on recent trends in electronics information & communication technology. Bangalore: IEEE, 2017: 425-429.

[20] WATERS B. Ciphertext-policy attribute-based encryption: an expressive, ecient, and provably secure realization[C]//Public key cryptography - PKC 2011. Taormina: Springer, 2011: 53-70.

[21] BEIMEL A. Secure schemes for secret sharing and key distribution[D]. Haifa, Israel: Israel Institute of Technology, 1996.

[22] BALU A, KUPPUSAMY K. An expressive and provably secure ciphertext-policy attribute based encryption[J]. Information Science, 2014, 276: 354-362.