云存储中支持属性级用户撤销的可验证CP-ABE方案

李树全[1]，佘堃[1]，李若洁[2]
（1.电子科技大学信息与软件工程学院，四川 成都 610054；2.哥伦比亚大学经济学系，美国 纽约 10027）

摘要：针对云存储中撤销用户单一属性易影响其他共享用户的问题，基于密钥加密密钥二叉树最小覆盖集与用户绑定的路径元，构造了可验证的属性级用户撤销CP-ABE方案。撤销由数据服务管理器在密文侧完成，仅更新相关密文行及节点分量，剩余用户无需更新密钥，可抵抗撤销用户与未撤销用户的合谋；配对运算全部外包且结果可校验。在标准模型下基于判定性q-Parallel BDHE假设证明了方案的选择性消息机密性。实验表明，用户端解密开销恒定，与访问策略规模无关。

关键词：密文策略属性基加密；云存储安全；属性级用户撤销；可验证外包解密；标准模型
中图分类号：TP309
文献标志码：A
doi：10.11959/j.issn.1000

Verifiable CP-ABE scheme supporting attribute-level user revocation in cloud storage

LI Shuquan[1], SHE Kun[1], LI Ruojie[2]
1. School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
2. Department of Economics, Columbia University, New York 10027, USA

Abstract: To revoke a single attribute of a user without requiring key updates from the other users sharing that attribute, a verifiable ciphertext-policy attribute-based encryption (CP-ABE) scheme was constructed from the minimum cover set of a key encryption key (KEK) binary tree and user-bound path elements. A data service manager performed revocation on the ciphertext side by updating only the affected ciphertext rows and node components; the remaining users kept their keys unchanged, and collusion between revoked and non-revoked users was prevented. All bilinear pairings were outsourced, and the user verified the transformed result. Selective message confidentiality of the scheme was proved in the standard model under the decisional *q*-parallel bilinear Diffie-Hellman exponent (BDHE) assumption. Experimental results show that the user-side decryption cost is constant and independent of the access policy size.

Key words: ciphertext-policy attribute-based encryption, cloud storage security, attribute-level user revocation, verifiable outsourced decryption, standard model

0 引言

云存储作为云计算的延伸与发展，其核心特征是提供“存储即服务”，使得用户能够随时随地通过联网设备便捷地存取数据，因而获得日益广泛的应用。然而，由于用户数据存储于云端服务器，脱离了用户的直接控制，如何保护敏感数据的机密性和合法访问成为用户最为关注的问题。近年来，随着数据持续向云端迁移，云存储在提供弹性、可扩展性和成本优势的同时，仍面临数据机密性、完整性、访问控制及隐私保护等安全挑战[1-2]。

收稿日期：XXXX–XX–XX；修回日期：XXXX–XX–XX
通信作者：李树全，shuquanli@uestc.edu.cn
基金项目：国家自然科学基金资助项目（No.62273071）
Foundation Items：The National Natural Science Foundation of China (No.62273071)

在Shamir提出基于身份的加密概念[3]及Boneh等实现首个实用的基于身份加密方案[4]后，Sahai等人于2005年提出了属性基加密（attribute-based encryption, ABE）[5]方案。凭借其策略表达的灵活性，ABE在云存储系统中得到了广泛应用。然而，在实际应用中也面临关键挑战：云存储用户规模庞大，不同用户常共享相同属性。当需要撤销某一用户的特定属性时，如何在不影响其他拥有该属性的用户正常访问、不增加系统显著负担且不降低安全性的前提下，高效完成权限撤销，已成为一个亟待解决的重要问题。

1 相关工作

1.1 属性撤销

早期研究主要关注用户级撤销。Ostrovsky等[6]、Staddon等[7]、Liang等[8]、Attrapadung等[9]与Wang等[10]分别基于非单调访问结构、属性集约束、二叉树更新密钥、直接与间接混合撤销以及身份撤销列表实现了用户撤销，但存在效率较低、适用条件受限或用户私钥变长等问题；Xu等[11]（2019）在标准模型下实现了可撤销ABE，但密钥和密文更新规模随系统生命周期增长而扩大。Dai等[12]（2025）利用密钥加密密钥(key encryption key, KEK)树与部分密文更新实现恶意用户的追踪与撤销，其树叶节点对应用户，撤销对象为整个用户；Wang等[13]（2026）基于二叉树最小覆盖集按时间周期发布更新密钥实现间接撤销并支持黑盒追踪，但未撤销用户需结合当前周期的更新密钥生成有效解密密钥。可以发现，以上方案都只能实现系统级的用户撤销，无法仅撤销用户的某一属性；一旦用户被撤销，其全部属性对应的访问权限均失效。

在属性级用户撤销方面，Pirretti等[14]为属性设置有效期并周期性发布新版本，属性无法即时撤销；Ibraimi等[15]与Yu等[16]分别引入仲裁者和代理服务器实现即时撤销，但均要求第三方保持在线，后者还需更新全部未撤销用户的私钥；Yang等[17]的方案中属性授权机构需同时更新密文与用户密钥，计算负担较重。Hur等[18]（2011）基于KEK二叉树实现了细粒度属性撤销，但安全性仅在一般群模型下得到保证，缺乏标准模型下的形式化安全证明，且解密代价与属性数量线性相关。Yang等[19]（2013）实现了属性级撤销，但安全证明基于随机预言机模型，且非撤销用户需与授权机构交互更新密钥，不适合离线场景。

闫玺玺等[20]（2015）在数据外包环境下提出了一种支持即时属性撤销的CP-ABE方案，借助KEK二叉树与对称加密实现属性群密钥分发，但其属性群密钥以标量形式经密文头交付用户，可在用户间转发，且其安全模型要求询问属性集与挑战访问结构不相交，未覆盖属性撤销。马华等[21]（2015）基于LSSS与KEK二叉树实现了属性群成员变更，并将部分解密运算外包给解密服务器，但其属性群密钥同样以共享标量形式交付用户，且未给出形式化安全证明。Li等[22]（2018）以同一随机量将用户属性私钥与用户专属KEK绑定，以抵抗撤销用户与未撤销用户的合谋，但撤销时需为属性群全部剩余成员重新计算并分发KEK，且不支持外包解密。孙磊等[23]（2019）针对文献[21]等方案中属性私钥与属性群密钥相互独立所带来的合谋攻击隐患提出改进方案，并结合了外包解密，其密文头采用单元素形式，重加密时对全部密文行再随机化，且仅支持撤销。Yao等[24]（2026）在多授权机构ABE中同时支持属性撤销、白盒追踪、解密外包与策略更新，但撤销时需由授权机构为属性群全部剩余成员逐一计算并下发更新后的密钥分量。Li等[25]（2025）由数据拥有者针对待撤销访问的密文生成委托，云服务器据此将密文策略更新为原策略与新策略的合取，免除了未撤销用户的密钥更新，但每次撤销均需数据拥有者参与，且撤销通过收紧密文访问策略实现。

1.2 外包解密与可验证性

将代价高昂的配对运算外包给服务器可有效缓解CP-ABE的用户端解密负担，Green等[26]首次提出外包解密ABE方案；Lai等[27]形式化了可验证性的安全模型，指出必须将原始密文作为最终解密的辅助输入，并通过附加一套平行加密分量与校验值实现标准模型下的可验证性，但密文与解密开销近乎加倍。Qin等[28]给出将任意外包ABE转换为可验证外包ABE的通用方法：ABE层封装随机会话秘密，其抗碰撞哈希值作为验证信息，并经随机性提取器派生对称密钥，使额外密文开销降为一个短哈希值，机密性与可验证性均在标准模型下证明。仲红等[29]与明洋等[30]将随机秘密封装的思路扩展至

多授权机构场景，验证令牌随密文存储。王经纬等[31]（2023）在多授权场景下采用属性版本控制实现撤销并支持可验证外包解密，但撤销时需向全部未撤销用户发送更新密钥。Hou等[32]（2024）提出了基于区块链的可验证外包ABE方案，将数据用户生成转换密钥的计算开销由与属性数相关降低为常数级。

1.3　本文工作

针对云存储环境下数据机密性保护与细粒度访问控制需求，基于Waters[33]提出的CP-ABE方案，并借鉴Hur等[18]的属性撤销思路与Qin等[28]的可验证外包解密思路，本文提出了一种支持属性级用户撤销与可验证外包解密的CP-ABE方案（R-CP-ABE, revocable ciphertext-policy attribute-based encryption）。与现有方案相比，本文方案主要贡献如下：

1）实现了抗合谋且无需合法用户同步更新的属性级用户撤销。现有方案或不能抵抗撤销用户与未撤销用户的合谋[18,20-21]，或撤销时须向剩余成员重新下发密钥材料并维护每用户状态[22-24,31]。本文将成员资格绑定由密钥分发侧转移至密文头侧：用户持有与转换密钥随机量t/z绑定的静态路径元，密文头按KEK树最小覆盖集生成节点分量；撤销与授予通过更新存储密文实现，用户已有密钥材料无需更新，DSM除节点秘密与成员表外不维护每用户状态。

2）引入可验证外包解密机制。解密时全部双线性配对运算委托给DSM，用户端最终解密仅需一次G_T上的指数运算及常数次哈希运算，其计算量与参与重构的属性数及访问策略规模无关；同时沿用文献[28]的会话秘密封装与验证信息范式，使用户能够校验外包转换结果的正确性。

3）在标准模型下基于判定性q-Parallel BDHE假设证明了方案在选择性安全模型下的消息机密性，所用安全模型将撤销与授予过程纳入敌手的询问能力；并通过命题1论证了属性撤销的有效性与撤销用户同未撤销用户之间的抗合谋性，通过命题2论证了跨版本密文拼接攻击的抵抗性。

4）在128位安全级别的对称配对参数下，与文献[21,23]进行了加密、外包转换、用户解密及撤销开销的对比实验，并考察了属性群规模及成员在KEK树中的分布对撤销开销的影响。结果表明，本文用户端解密开销恒定、属性级撤销无需剩余合法用户参与。

2　预备知识

2.1　双线性群

双线性群是构造属性基加密方案的基础工具[33]。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$e: G\times G\to G_T$满足双线性、非退化性与可计算性。

2.2　判定性q-Parallel BDHE假设

本文采用的判定性q-Parallel BDHE假设由Waters[33]在其CP-ABE方案中正式提出，定义如下：选择一个阶为大素数p的群G，其生成元为g。随机选取$a,s,b_1,b_2,\cdots,b_q\in\mathbb{Z}_p$，给定敌手参数：

$$y=\Big(g,g^{s},g^{a},\cdots,g^{a^{q}},g^{a^{q+2}},\cdots,g^{a^{2q}},\ \forall 1\le j\le q: g^{sb_j},g^{a/b_j},\cdots,g^{a^{q}/b_j},g^{a^{q+2}/b_j},\cdots,g^{a^{2q}/b_j},\ \forall 1\le j,k\le q,k\ne j: g^{asb_k/b_j},\cdots,g^{a^{q}sb_k/b_j}\Big)$$

对于敌手而言，区分$e(g,g)^{a^{q+1}s}\in G_T$和G_T中的某个随机元素是困难的。定义算法$\mathcal{B}$的输出为$d\in\{0,1\}$，若输出概率满足

$$\left|\Pr\left[\mathcal{B}\left(y,Z=e(g,g)^{a^{q+1}s}\right)=0\right]-\Pr\left[\mathcal{B}\left(y,Z\leftarrow G_T\right)=0\right]\right|\ge\varepsilon$$

则称$\mathcal{B}$解决群G中判定性q-Parallel BDHE假设的优势为ε。如果不存在多项式时间算法$\mathcal{B}$能够以不可忽略的优势攻破判定性q-Parallel BDHE困难问题，则称判定性q-Parallel BDHE假设成立。

2.3　线性秘密共享方案

线性秘密共享方案(linear secret sharing scheme, LSSS)由Beimel[34]提出，本文沿用文献[33]中的标准定义：

设$\boldsymbol{M}$为$l\times n$的共享生成矩阵，映射$\rho: \{1,2,\cdots,l\}\to W$将其第$i$行$\boldsymbol{M}_i$标记为属性$\rho(i)$。共享秘密$s\in\mathbb{Z}_p$时，随机选取$y_2,\cdots,y_n\in\mathbb{Z}_p$，令列向量$\boldsymbol{v}=(s,y_2,\cdots,y_n)^{\mathrm{T}}$，则$\lambda_i=\boldsymbol{M}_i\boldsymbol{v}$为分配给属性$\rho(i)$的份额。

对属性集合S，令$I=\{i:\rho(i)\in S\}$。若存在常数

$\{w_i \in \mathbb{Z}_p\}_{i \in I}$ 使得 $\sum_{i \in I} w_i \boldsymbol{M}_i = (1,0,\cdots,0)$，则对任意合法份额有 $\sum_{i \in I} w_i \lambda_i = s$，此时称 S 满足访问结构 $(\boldsymbol{M},\rho)$，记为 $S \models (\boldsymbol{M},\rho)$，且 $\{w_i\}$ 可在关于 $\boldsymbol{M}$ 规模的多项式时间内求得；否则记为 $S \not\models (\boldsymbol{M},\rho)$，此时存在向量 $\boldsymbol{w}$，使得 $w_1 = -1$ 且对所有 $i \in I$ 有 $\boldsymbol{M}_i \boldsymbol{w}^{\mathrm{T}} = 0$。

2.4　密钥加密密钥树

密钥加密密钥树是指数据服务管理器（data service manager, DSM）基于用户集合 $\mathcal{U}$ 建立的完全二叉树[18]，用于表示和更新属性群的成员状态，从而实现属性级用户撤销，如图 1 所示。假设系统用户集合 $\mathcal{U} = \{u_1,u_2,\cdots,u_N\}$，属性集合 $W = \{att_1,att_2,\cdots,att_n\}$。设 $U_i \subseteq \mathcal{U}$ 是拥有属性 att_i 的用户集合，被称为属性群。例如，用户 u_1 拥有属性 $\{att_1,att_2\}$，用户 u_2 拥有属性 $\{att_1,att_2,att_3\}$，用户 u_3 拥有属性 $\{att_2,att_3\}$，则属性 att_1 对应的用户群 $U_1 = \{u_1,u_2\}$，属性 att_2 对应的用户群 $U_2 = \{u_1,u_2,u_3\}$，属性 att_3 对应的用户群 $U_3 = \{u_2,u_3\}$。

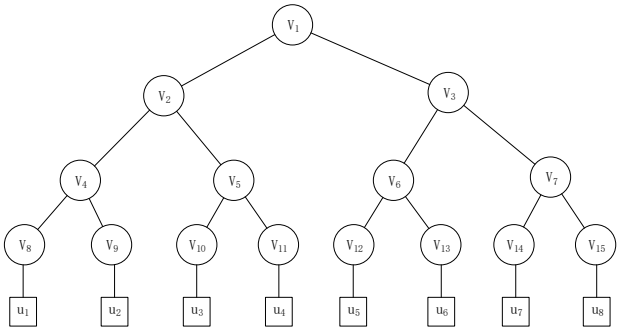


图 1　KEK 二叉树结构示意图

DSM 按如下过程构建密钥加密密钥树并计算相关节点集合：

1）DSM 根据从属性授权中心（attribute authority, AA）收到的属性群成员列表建立密钥加密密钥树，系统中的每个用户被分配到二叉树的一个叶节点，二叉树的每个节点记为 v_j。在本文方案中，DSM 为每个节点 v_j 与每个属性 x 生成一个节点秘密 $\theta_{x,j}$（见第 4 节）。

2）路径节点集 $\mathrm{Path}(u_t)$. 对于每一个用户 $u_t \in \mathcal{U}$，从叶节点到根节点上的所有节点被定义为用户 u_t 的路径节点，如 $\mathrm{Path}(u_7) = \{v_{14},v_7,v_3,v_1\}$。

3）最小覆盖集算法 $\mathrm{Mincs}(U_i)$. 对于拥有属性 att_i 的属性群 U_i，将所覆盖叶节点恰为 U_i 全部成员的最小树节点集合称为最小覆盖集 $\mathrm{Mincs}(U_i)$。如 $U_i = \{u_1,u_2,u_3,u_4,u_7,u_8\}$，因为节点 v_2 和 v_7 可以覆盖 U_i 中的所有成员，则 $\mathrm{Mincs}(U_i) = \{v_2,v_7\}$。

4）求 $\mathrm{Path}(u_k)$ 与 $\mathrm{Mincs}(U_i)$ 的交集：若用户拥有属性 att_i，即 $u_k \in U_i$，则交集有且只有一个节点；若 $u_k \notin U_i$，则交集为空。例如，在上述例子中 $\mathrm{Path}(u_7) \cap \mathrm{Mincs}(U_i) = \{v_7\}$。

3　系统模型与安全模型

本节首先描述了可撤销密文策略属性基加密方案 R-CP-ABE 的系统模型及其各组成部分的功能，然后对其进行形式化的描述，最后给出该方案的安全模型。

3.1　系统模型

本文提出的 R-CP-ABE 方案系统模型如图 2 所示。系统主要由属性授权中心 (AA)、云服务商 (cloud service provider, CSP)、数据拥有者 (data owner, DO) 和数据用户 (data user, DU) 4 个部分组成。

AA 主要负责系统公钥和主私钥的生成、用户属性授权与属性密钥签发，并维护用户属性授权状态；发生属性撤销或授予时，AA 更新相应属性群成员状态并通知 DSM，是系统所有实体中唯一被完全信任的一方。

CSP 负责提供数据外包服务，由数据服务器和数据服务管理器（DSM）组成。CSP 的存储服务保证所存密文各分量（含验证令牌）的完整性，DSM 负责维护属性群成员状态、KEK 树节点秘密及对应的密文头更新。当被 AA 告知需要更新密文时，DSM 对数据密文进行重加密操作。在解密数据时，DSM 会承担部分计算任务。同时本文假设 CSP 是诚实但好奇的 (honest-but-curious)，它既会诚实地按照指示正确地执行步骤，也会出于好奇在工作过程中窥探用户数据隐私，但 DSM 不与数据用户合谋。

关于上述假设的适用边界：DSM 持有节点秘密 *DSK* 与用户提交的转换密钥 *TK*，但不持有用户私钥 *SK*，定理 2 的证明中 *DSK* 由模拟器自行生成，故 DSM 单独或 *DSK* 泄露均不破坏机密性，DSM 与属性不满足访问策略的用户合谋亦可由定理 2 的

证明直接扩展覆盖；DSM与已被撤销的用户合谋时可按撤销前的覆盖集重新生成节点分量使撤销失效，这是由服务器执行撤销的方案[18,20-23]的共同边界，本文与同类文献一致。属性群成员变更的原子性由DSM以更新事务保证：更新期间对受影响密文加写锁并暂停处理相应的转换请求，全部更新完成后一次性替换存储中的密文，更新过程中的中间结果不对外提供。用户完全退出等价于撤销其全部属性，其叶节点不再分配给其他用户，以免已退出用户的路径元与新用户所在的覆盖节点匹配。

DO是拥有数据并希望将数据管理外包给CSP的客户端。DO负责定义访问策略，在外包前根据该策略加密数据，并将密文上传至CSP。

DU是系统中的消费者，可从CSP获取密文。若DU拥有满足DO所定义的加密数据访问策略的一组属性，并且在重构所需的属性上均为当前属性群成员，则该用户能够向DSM提交转换请求，接收转换结果后仅需执行常数次轻量运算即可恢复明文。

从系统部署的角度看，DSM承担的是外包计算与属性群状态维护两项职能，既可与数据服务器一样部署于云端，也可下沉至靠近终端的边缘节点，以缩短转换结果的回传时延。

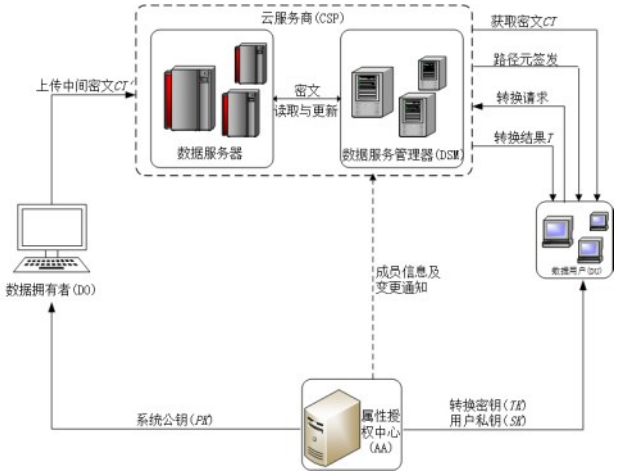


图2　R-CP-ABE方案系统模型图

3.2　系统形式化定义

首先给出本文相关符号及其含义，如表1所示。

R-CP-ABE方案由AA、DSM、DO、DU四类实体执行以下10个算法，分为8个阶段。

1) 系统初始化阶段

$\mathrm{AASetup}\left(1^{\lambda}\right)\rightarrow\left(\mathrm{PK},\mathrm{MSK}\right)$：AA运行该算法，输入安全参数$\lambda$，输出系统公钥PK和系统主私

表1　　**相关符号及其含义**

符号	含义
S	数据用户的属性集合
U_x	属性x对应的用户属性群
$(\boldsymbol{M},\rho)$	数据拥有者定义的LSSS访问结构
TK	转换密钥，$TK=\left(K,L,\{K_x\}_{x\in S}\right)$
SK	用户私钥SK = z，用于对外包转换结果进行最终去盲化
DSK	DSM主私钥，由各属性与KEK树节点对应的节点秘密组成
$\theta_{x,j}$	属性x在KEK树节点j上的节点秘密
Path(id)	用户id从叶节点到根节点的路径节点集合
$\mathrm{Mincs}\left(U_x\right)$	当前属性群U_x的最小覆盖节点集合
$P_{\mathrm{id},x,j}$	用户id关于属性x和路径节点j的路径元；上下文中用户确定时省略下标id
CT′	DO执行Encrypt后生成的中间密文
CT	DSM执行DSMEncrypt生成的最终密文
T	DSM外包解密后生成的转换结果
Hdr	与当前属性群状态对应的密文头
Token	密文校验值，$\mathrm{Token}=H\left(\mathrm{Tag}_0\ //\ C_M\right)$，其中$\mathrm{Tag}_0=H_0(R)$为会话秘密$R$的标签

钥 MSK。

2) 数据服务管理器初始化阶段

$\mathrm{DSMSetup}(N|W)\rightarrow \mathrm{DSK}$：DSM 运行该算法，输入用户规模 N 与属性集 W，建立 KEK 二叉树，为每个树节点 v_j 与每个属性 x 生成节点秘密 $\theta_{x,j}$，输出 DSM 主私钥 $\mathrm{DSK}=\{\theta_{x,j}\}$。此后 DSM 仅维护 DSK 与属性群成员表 $\{U_x\}$，不为单个密文保存任何秘密状态。

3) 密钥生成阶段

$\mathrm{AAKeyGen}(\mathrm{MSK},\mathrm{id},S)\rightarrow(\mathrm{TK}|\mathrm{SK})$：AA 运行该算法，输入系统主私钥 MSK、用户身份 id 及其属性集合 S，输出该用户的转换密钥 TK 和用户私钥 SK，TK 与 SK 通过同一随机量 z 绑定。

4) 路径元生成阶段

$\mathrm{DSMPathGen}(\mathrm{DSK},\mathrm{id},S,L)\rightarrow\{P_{\mathrm{id},x,j}\}$：DSM 运行该算法，输入 DSK、用户身份 id、属性集合 S 及该用户转换密钥中的分量 L，将 id 加入各属性群 U_x（$x\in S$），并输出用户 id 关于每个属性 $x\in S$、每个路径节点 $j\in\mathrm{Path}(\mathrm{id})$ 的路径元 $P_{\mathrm{id},x,j}$。

5) 数据加密阶段

$\mathrm{Encrypt}(\mathrm{PK},(\boldsymbol{M}|\rho),\mathrm{msg})\rightarrow CT'$：DO 运行该算法，输入系统公钥 PK、访问结构 $(\boldsymbol{M}|\rho)$ 和明文消息 msg，输出中间密文 CT'。

6) 数据重加密阶段

$\mathrm{DSMEncrypt}(\mathrm{PK},\mathrm{DSK},CT')\rightarrow(\mathrm{CT},\mathrm{Hdr})$：DSM 运行该算法，输入 PK、DSK 与中间密文 CT'，按各属性群当前的最小覆盖集 $\{\mathrm{Mincs}(U_x)\}$ 对密文进行重加密，输出密文 CT 和密文头 Hdr。CT 与 Hdr 共同构成访问受控的存储密文。

7) 数据解密阶段

$\mathrm{DSMDecrypt}(\mathrm{TK},\{P_{x,j}\},\mathrm{CT},\mathrm{Hdr})\rightarrow T$ 或 $\perp$：DSM 收到用户的转换请求后运行该算法，输入该用户提交的转换密钥 TK 与路径元 $\{P_{x,j}\}$、DSM 自身存储中的密文 CT 与密文头 Hdr。DSM 核验路径元与 TK 的一致性；若用户属性满足访问结构且在重构所需的属性上均为当前属性群成员，输出转换结果 T，否则输出 $\perp$。

$\mathrm{Decrypt}(\mathrm{CT},T,\mathrm{SK})\rightarrow$ msg 或 $\perp$：DU 运行该算法，输入自 CSP 取回的密文 CT、DSM 返回的转换结果 T 与用户私钥 SK，用 CT 中的校验值验证转换结果的正确性；校验通过则输出明文 msg，否则输出 $\perp$。

8) 属性群成员变更阶段

当用户 id 的属性集合由 $S_{\mathrm{id}}^{\mathrm{old}}$ 变为 $S_{\mathrm{id}}^{\mathrm{new}}$ 时，令 $S_{\mathrm{id}}^{-}=S_{\mathrm{id}}^{\mathrm{old}}\setminus S_{\mathrm{id}}^{\mathrm{new}}$，$S_{\mathrm{id}}^{+}=S_{\mathrm{id}}^{\mathrm{new}}\setminus S_{\mathrm{id}}^{\mathrm{old}}$，对每个 $x\in S_{\mathrm{id}}^{-}$，AA 撤销用户对属性 x 的授权并通知 DSM，DSM 将用户移出当前属性群 U_x，随后执行 DSMRevoke；对每个 $x\in S_{\mathrm{id}}^{+}$，AA 授予属性并补发 K_x，DSM 生成相应路径元、将用户加入 U_x，随后执行 DSMGrant。新用户注册是 $S_{\mathrm{id}}^{\mathrm{old}}=\varnothing$ 的特例。同一次变更中先处理 S_{id}^{-} 再处理 S_{id}^{+}，以防撤销前的旧行与授予后的新行拼接。每次成员变更连同对全部相关存储密文的更新视为一次原子操作，更新完成前 DSM 不处理新的加密与转换请求。用户本地缓存的旧密文与旧密文头不参与 DSM 的转换；用户在撤销前已具备解密资格时，对其缓存的旧版本的访问是合法的，本方案与文献[18,20-23]相同，不对已取得的历史版本提供追溯性保护；用户在任一版本下均不具备解密资格时，缓存的历史版本不能被组合利用，见 5.4 节命题 2。

$\mathrm{DSMRevoke}(\mathrm{PK},\mathrm{DSK},\mathrm{CT},\mathrm{Hdr},x)\rightarrow(\mathrm{CT}^{\mathrm{new}}|\mathrm{Hdr}^{\mathrm{new}})$：属性群 U_x 缩减时，DSM 对存储中每个访问策略含 x 的密文运行该算法，仅对访问结构中与属性 x 相关的行刷新行秘密、更新密文分量，并按新的 $\mathrm{Mincs}(U_x)$ 重新生成该行的节点分量。其余行的密文分量、其余属性的节点分量，以及所有用户的 TK、SK 与路径元均保持不变。更新后，被撤销用户对该密文不再享有由该属性带来的访问权。

$\mathrm{DSMGrant}(\mathrm{PK},\mathrm{DSK},\mathrm{CT},\mathrm{Hdr},x)\rightarrow(\mathrm{CT}^{\mathrm{new}}|\mathrm{Hdr}^{\mathrm{new}})$：属性群 U_x 扩大时，DSM 对存储中每个访问策略含 x 的密文运行该算法，对整个密文的 ABE 分量与封装分量再随机化，并对全部行刷新行秘密、按各行属性当前的最小覆盖集重新生成节点分量。会话秘密 R、C_M 与 Token 不变；除被授予属性 x 的用户追加相应的 K_x 与 $\{P_{\mathrm{id},x,j}\}$ 外，所有用户已有的 TK、SK 与路径元均保持不变。更新后，被授予用户即可访问该已有密文。

3.3 安全模型

本文在文献[33]选择性安全模型的基础上扩展撤销、授予与外包解密，刻画任一属性群状态下当

前密文的机密性；敌手可对包括挑战密文在内的任意密文发起撤销与授予更新询问，但不涉及多个历史版本材料的联合安全性，后者在5.4节命题2中讨论。DSM为诚实但好奇且不与用户合谋的实体，其重加密、路径元生成与密文更新分别通过挑战密文中的节点分量与密文头、私钥询问返回的路径元、更新询问返回的更新结果在游戏中刻画。R-CP-ABE在选择属性和明文攻击下的不可区分性（IND-sAtt-CPA）安全游戏定义如下：

1) 初始化阶段. 敌手$\mathcal{A}$选择将要攻击的访问结构$(\boldsymbol{M}^*,\rho^*)$发送给挑战者$\mathcal{B}$。

2) 系统建立阶段. 挑战者$\mathcal{B}$运行AASetup与DSMSetup算法，并把产生的系统公钥PK发送给敌手$\mathcal{A}$。

3) 查询阶段1. 敌手$\mathcal{A}$适应性地提交以下两类询问。

□ 密钥询问KeyQuery(id,S)：敌手$\mathcal{A}$适应性地提交一系列密钥查询，其中S为用户的属性集合，id仅用于确定该用户在KEK二叉树中的叶节点及Path(id)，所有查询均须满足$S\not\models(\boldsymbol{M}^*,\rho^*)$，挑战者返回与(id,$S$)对应的TK、SK及路径元。

□ 更新询问UpdQuery(CT,Hdr,x,id,op)，op$\in$\{Revoke,Grant\}：挑战者将id移出或加入属性群U_x，对敌手指定的密文(CT,Hdr)运行DSMRevoke或DSMGrant，返回更新后的(CT^{new},Hdr^{new})。

4) 挑战阶段. 敌手$\mathcal{A}$提交长度相等的2个消息msg_0与msg_1给挑战者$\mathcal{B}$；然后$\mathcal{B}$随机选择$b\in\{0,1\}$，并在访问结构$(\boldsymbol{M}^*,\rho^*)$下加密$\mathrm{msg}_b$，产生中间密文$CT^{*\prime}$；再运行DSMEncrypt(PK,DSK,$CT^{*\prime}$)得到最终挑战密文CT*和头信息Hdr*，并将(CT*,Hdr*)发送给敌手$\mathcal{A}$。

5) 查询阶段2. 重复查询阶段1的两类询问，限制条件与查询阶段1相同。

6) 猜测阶段. 敌手$\mathcal{A}$输出一个值b'作为对b的猜测。如果$b'=b$，则称敌手$\mathcal{A}$赢得了该游戏。敌手$\mathcal{A}$在该游戏中的优势定义为

$$\mathrm{Adv}_{\mathcal{A}}=\left|\Pr[b'=b]-1/2\right|$$

定义 1. R-CP-ABE方案是安全的，如果在多项式时间内没有敌手能够以不可忽略的优势来攻破以上安全模型。

4 R-CP-ABE方案构造

本文要求ρ为单射，即每个属性在访问矩阵中至多出现一行；与文献[33]相同，可通过为每个属性设置k份公开参数放宽为至多k次使用。

1) 系统初始化阶段. 由属性授权中心AA执行初始化，建立系统。

AASetup$(1^\lambda)\to$(PK,MSK)：初始化算法选择两个其阶为素数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$e:G\times G\to G_T$。属性授权中心AA随机生成$\alpha,a\in\mathbb{Z}_p$和$h_1,h_2,\cdots,h_{|W|}\in G$，选取抗第二原像哈希函数$H_0:G_T\to\{0,1\}^{128}$与抗碰撞哈希函数$H:\{0,1\}^*\to\{0,1\}^{256}$；从两两独立哈希函数族$\mathcal{H}$中随机选取$h:G_T\to\{0,1\}^{128}$，用于从会话秘密中提取对称密钥；选取密钥长度为$\mathrm{l}_{SE}=128$位、满足一次性语义安全(one-time semantic security)的对称加密方案SE=(SE.Enc,SE.Dec)；设置主私钥MSK$=g^\alpha$，系统公钥PK$=\left(g,e(g,g)^\alpha,g^a,h_1,\cdots,h_{|W|},H_0,H,h,\mathrm{SE}\right)$。

2) 数据服务管理器初始化阶段. 由DSM执行。

DSMSetup$(N|W)\to$DSK：以系统用户集$\mathcal{U}$（$|\mathcal{U}|=N$）为叶节点建立KEK二叉树，记其全部节点的下标集合为Nodes，并以下标j指代节点v_j。对每个$j\in$Nodes与每个属性$x\in W$，随机选$\theta_{x,j}\in\mathbb{Z}_p^*$，令

$$\mathrm{DSK}=\left\{\theta_{x,j}\right\}_{x\in W,j\in\mathrm{Nodes}}$$

$\theta_{x,j}$为节点v_j关于属性x的节点秘密，由DSM保密。用户id的路径节点集记为Path(id)，属性群U_x的最小覆盖集记为Mincs(U_x)，定义同2.4节。

3) 密钥生成阶段. 用户向AA申请属性私钥。

AAKeyGen(MSK,id,S)$\to$(TK|SK)：AA验证用户id的属性集S后，随机选$t\in\mathbb{Z}_p$，$z\in\mathbb{Z}_p^*$，计算

$$\mathrm{TK}=\left(K=g^{\alpha/z}g^{at/z},\ L=g^{t/z},\ \left\{K_x=h_x^{t/z}\right\}_{x\in S}\right)$$

$$\mathrm{SK}=z$$

AA经安全信道将(TK|SK)发送给用户，将(id,S,L)发送给DSM。

4) 路径元生成阶段. 由DSM执行。

DSMPathGen(DSK,id,S,L)$\to\left\{P_{\mathrm{id},x,j}\right\}$：对每个$x\in S$、$j\in$Path(id)计算

$$P_{\mathrm{id},x,j} = L^{1/\theta_{x,j}} = g^{t/(z\theta_{x,j})}$$

经安全信道发送给用户，DSM 不保存。用户注册时对其全部属性执行一次；已有用户新增属性 x' 时，AA 以该用户密钥生成时的随机量 t/z 计算并补发 $K_{x'} = h_{x'}^{t/z}$，DSM 由 L 计算并补发 $\left\{P_{id,x'j}\right\}_{j\in Path(id)}$，二者绑定同一随机量 t/z，追加到用户已有的 TK 与路径元集合上。路径元与密文无关、与属性群的后续变动无关，长期不变。

5) 数据加密阶段. 由数据拥有者 DO 执行。

$\mathrm{Encrypt}\left(\mathrm{PK},(\boldsymbol{M}|\rho),\mathrm{msg}\right)\to CT'$：$\boldsymbol{M}$ 为 $l\times n$ 矩阵。随机选 $\boldsymbol{v}=\left(s|y_2|\cdots|y_n\right)^{\mathrm{T}}\in\mathbb{Z}_p^n$，计算 $\lambda_i=\boldsymbol{M}_i\boldsymbol{v}$；随机选 $r_1,\cdots,r_l\in\mathbb{Z}_p$，计算

$$C'=g^s, C_i=g^{a\lambda_i}h_{\rho(i)}^{-r_i}, D_i=g^{r_i}, i=1,\cdots,l$$

随机选取会话秘密 $R\in G_T$，计算 $C=R\cdot e(g|g)^{\alpha s}$，计算

$$\mathrm{Tag}_0=H_0(R), K_E=h(R)$$

$$C_M=\mathrm{SE.Enc}_{K_E}(\mathrm{msg})$$

$$\mathrm{Token}=H\left(\mathrm{Tag}_0\ /\!/\ C_M\right)$$

其中 // 表示字节串拼接；由于 Tag_0 长度固定为 $\mathrm{l}_{H_0}=128$ 位，$\mathrm{Tag}_0\ /\!/\ C_M$ 可唯一切分为 Tag_0 与 C_M，即不同的 (Tag_0,C_M) 拼接结果必不相同，故结果校验的可靠性可归约到 H 的抗碰撞性。输出中间密文 $CT'=\left(C,C_M,C',\left\{C_i|D_i\right\}_{1\le i\le l},\mathrm{Token},(\boldsymbol{M}|\rho)\right)$，上传至 CSP。

6) 数据重加密阶段. 由 DSM 执行，将属性级访问控制嵌入密文。

$\mathrm{DSMEncrypt}(\mathrm{PK},\mathrm{DSK},CT')\to(\mathrm{CT},\mathrm{Hdr})$：对每一行 i，记 $x=\rho(i)$，随机选行秘密 $k_i\in\mathbb{Z}_p$，计算

$$C_i'=C_i\cdot\left(g^a\right)^{k_i}$$

对每个覆盖节点 $j\in\mathrm{Mincs}(U_x)$，随机选 $r_{i,j}\in\mathbb{Z}_p$，计算节点分量

$$C_{i,j}=g^{-ak_i}\cdot g^{-r_{i,j}/\theta_{x,j}}, D_{i,j}=g^{r_{i,j}}$$

输出

$$\mathrm{CT}=\left(C,C_M,C',\left\{C_i'|D_i\right\}_{1\le i\le l},\mathrm{Token},(\boldsymbol{M}|\rho)\right)$$

$$\mathrm{Hdr}=\left\{\left(i|j|C_{i,j}|D_{i,j}\right):1\le i\le l,j\in\mathrm{Mincs}\left(U_{\rho(i)}\right)\right\}$$

DSM 将 (CT,Hdr) 上传到云存储服务器存储。节点分量采用两元素形式以便在标准假设下证明安全性；若压缩为单个群元素 $g^{k_i\theta_{x,j}}$ 可节省存储，但其安全性目前仅能在一般群模型下论证。

7) 数据解密阶段. 数据用户先从 CSP 取回密文 CT。

□ DSM 外包解密.

$\mathrm{DSMDecrypt}\left(\mathrm{TK},\left\{P_{x,j}\right\},\mathrm{CT},\mathrm{Hdr}\right)\to T$ 或 $\perp$：用户提交转换请求、TK 与路径元 $\left\{P_{x,j}\right\}_{x\in S,j\in J}$，其中 J 为其路径节点索引集。DSM 首先对每个提交的路径元核验

$$P_{x,j}^{\theta_{x,j}}\stackrel{?}{=}L$$

如果不相等，则说明路径元与 TK 不属于同一用户，返回 $\perp$。令 $I=\left\{i:\rho(i)\in S\right\}$。对每个 $i\in I$，若 $J\cap\mathrm{Mincs}\left(U_{\rho(i)}\right)\neq\varnothing$，取其中的节点 j_i（由最小覆盖集性质恰有一个）；令 $I^{\mathrm{cur}}=\left\{i\in I:j_i\text{存在}\right\}$，即用户持有且当前为属性群成员的属性所对应的行。若不存在常数集合 $\left\{w_i\in\mathbb{Z}_p\right\}_{i\in I^{cur}}$ 使得 $\sum_{i\in I^{cur}}w_i\mathbf{M}_i=(1,0,\cdots,0)$，返回 $\perp$；否则固定一组 $\left(I^{cur},\left\{w_i\right\}\right)$，计算

$$T'=\prod_{i\in I^{\mathrm{cur}}}\left[e\left(C_i'C_{i,j_i},L\right)\cdot e\left(D_i,K_{\rho(i)}\right)\cdot e\left(D_{i,j_i},P_{\rho(i)j_i}\right)\right]^{w_i}$$

$$T''=e\left(C'|K\right),$$

$$T=\frac{T''}{T'}=e(g,g)^{\alpha s/z}$$

其中 $C_i'C_{i,j_i}$ 为 G 中的乘法，每个参与重构的行只需 3 次配对运算。DSM 将转换结果 T 返回给用户。与文献[18,20-21]不同，本方案的密文头不交付可被任何持有者直接使用的属性群密钥，而是交付节点分量 $\left(C_{i,j},D_{i,j}\right)$；用户由此获得的消去项 $e\left(g,g\right)^{-ak_it/z}$ 绑定其自身转换密钥的随机量 $\frac{t}{z}$，对其他用户无效。路径元在注册时一次签发，撤销时不更新，授予时仅追加新属性的一组，属性群成员变动通过更新存储密文实现；路径元携带节点索引并可由 DSM 用 $\theta_{x,j}$ 校验其与 TK 的一致性，因此转换请求无需附带用户身份。

□ 用户解密.

Decrypt$(\mathrm{CT}, T, \mathrm{SK}) \to$ msg 或 $\perp$：用户从 CSP 取回的 CT 中读取 C、C_M、Token，以 $\mathrm{SK} = z$ 执行一次 G_T 上指数运算恢复会话秘密 $R = C \cdot (T^z)^{-1}$，计算 $\mathrm{Tag}_0 = H_0(R)$，校验

$$H(\mathrm{Tag}_0 \,/\!/\, C_M) \stackrel{?}{=} \mathrm{Token}$$

不相等则返回 $\perp$；否则计算 $K_E = h(R)$，输出 $\mathrm{msg} = \mathrm{SE.Dec}_{K_E}(C_M)$。校验的作用在于：若 DSM 返回错误的 T 使用户恢复出 $R' \neq R$，则通过校验需要 $H_0(R') = H_0(R)$，即找到 H_0 对随机目标 R 的第二原像；若 C_M 被篡改，则需要 H 的碰撞，该验证范式及其可验证性定义和证明与文献[28]相同，优势不超过 H_0 第二原像优势与 H 碰撞优势之和。由于 R、C_M 与 *Token* 在撤销与授予中保持不变，DSM 重放旧版本的转换结果或替换密文头，要么得到同一 R，要么同样需通过上述校验；DSM 返回 $\perp$ 的选择性失败属于可用性问题，DSM 与用户合谋已由 3.1 节的信任假设排除，二者不在可验证性定义范围内。

8) 属性群成员变更阶段. AA 按 3.2 节第 8 阶段的规则更新 $\{U_x\}$ 并通知 DSM，DSM 对所有涉及相应属性的已存储密文执行更新，两个算法都只用到 PK、DSK 与现存的 $(\mathrm{CT}, \mathrm{Hdr})$。

□ 撤销. DSMRevoke$(\mathrm{PK}, \mathrm{DSK}, \mathrm{CT}, \mathrm{Hdr}, x) \to (\mathrm{CT}^{\mathrm{new}} | \mathrm{Hdr}^{\mathrm{new}})$：对每个满足 $\rho(i) = x$ 的行 i：

(a) 任取 Hdr 中行 i 现存的一个节点分量 $(C_{i,j} | D_{i,j})$，计算

$$g^{-ak_i} = C_{i,j} \cdot D_{i,j}^{1/\theta_{x,j}}$$

随机选行秘密增量 $\Delta_i \in \mathbb{Z}_p$，令 $C_i' \leftarrow C_i' \cdot (g^a)^{\Delta_i}$，即行秘密由 k_i 更新为 $k_i + \Delta_i$，无需 DSM 知道 k_i。

(b) 删除 Hdr 中行 i 的全部旧节点分量，按新的 $\mathrm{Mincs}(U_x)$ 生成

$$C_{i,j'} = g^{-ak_i} \cdot g^{-a\Delta_i} \cdot g^{-r_{i,j'}/\theta_{x,j'}}$$

$$D_{i,j'} = g^{r_{i,j'}}, j' \in \mathrm{Mincs}(U_x)$$

其中 $r_{i,j'} \in \mathbb{Z}_p$ 为新随机数；$\mathrm{Mincs}(U_x) = \varnothing$ 时不生成节点分量，该行不再对任何用户可用。其余行的分量、C、C'、D_i、C_M、Token、其余属性的节点分量，以及所有用户的 TK、SK、路径元均保持不变，输出 $(\mathrm{CT}^{\mathrm{new}} | \mathrm{Hdr}^{\mathrm{new}})$。行秘密必须在每次撤销时刷新：旧节点分量的消去项为 $-ak_i$，而更新后的 C_i' 含 $k_i + \Delta_i$，故被撤销用户保留的旧节点分量对更新后的密文无用。

□ 授予. DSMGrant$(\mathrm{PK}, \mathrm{DSK}, \mathrm{CT}, \mathrm{Hdr}, x) \to (\mathrm{CT}^{\mathrm{new}} | \mathrm{Hdr}^{\mathrm{new}})$：本文采用独立会话秘密 R 的密钥封装机制(key encapsulation mechanism, KEM)构造。由于 R 与 ABE 加密指数 s 相互独立，DSMGrant 可利用公开参数重新随机化 ABE 密文分量而不改变 R，因而 C_M 与 Token 无需重新生成。

(a) 全密文再随机化. 随机选 $\boldsymbol{v}'' = (s'' | y_2'' | \cdots | y_n'')^{\mathrm{T}} \in \mathbb{Z}_p^n$ 与 $r_1'', \cdots, r_l'' \in \mathbb{Z}_p$，计算 $\lambda_i'' = \boldsymbol{M}_i \boldsymbol{v}''$，令

$$C \leftarrow C \cdot (e(g|g)^a)^{s''}, C' \leftarrow C' \cdot g^{s''},$$

$$C_i' \leftarrow C_i' \cdot g^{a\lambda_i''} h_{\rho(i)}^{-r_i''},$$

$$D_i \leftarrow D_i \cdot g^{r_i''}, i = 1, \cdots, l$$

(b) 对每一行 $i = 1, \cdots, l$，随机选 $\Delta_i \in \mathbb{Z}_p$，令 $C_i' \leftarrow C_i' \cdot (g^a)^{\Delta_i}$；对现存节点分量非空的行，按 DSMRevoke 的 (a)(b) 由任一旧节点分量恢复 g^{-ak_i}，删除该行全部旧节点分量，按当前的 $\mathit{Mincs}(U_{\rho(i)})$ 以新随机数重新生成

$$C_{i,j'} = g^{-a(k_i + \Delta_i)} \cdot g^{-r_{i,j'}/\theta_{\rho(i),j'}},$$

$$D_{i,j'} = g^{r_{i,j'}}, \quad j' \in \mathit{Mincs}(U_{\rho(i)})$$

R、C_M、Token 不变；除被授予属性 x 的用户追加 K_x 与 $\{P_{\mathrm{id},x,j}\}$ 外，所有用户已有的 TK、SK、路径元不变，输出 $(\mathrm{CT}^{\mathrm{new}} | \mathrm{Hdr}^{\mathrm{new}})$。整体再随机化并刷新全部行秘密后，新版本各行的份额与行秘密均与授予前版本相互独立，不同版本的行无法在同一次重构中组合使用（见 5.4 节命题 2）；因此，同时被撤销属性 y、被授予属性 x 的用户，不能将授予前版本中由 y 对应行得到的配对值与授予后版本中由 x 对应行得到的配对值组合，从而获得其在任一时刻都不具备的访问权。

5 正确性与安全性分析

5.1 方案正确性分析

定理 1（R-CP-ABE 方案的正确性） 对于任意安全参数 λ，任意访问结构 $(\boldsymbol{M},\rho)$，任意明文消息 msg，若用户属性集合 $S\models(\boldsymbol{M},\rho)$，且用户在重构所需的属性上均为当前属性群成员，则：

$$\text{Decrypt}(\text{CT},T,\text{SK})=\text{msg}$$

其中 $T=\text{DSMDecrypt}\left(\text{TK},\left\{P_{x,j}\right\},\text{CT},\text{Hdr}\right)$。

证明. 由第 4 节数据解密阶段的推导可知，DSM 外包解密阶段利用 $\text{TK},\left\{P_{x,j}\right\},\text{CT},\text{Hdr}$ 进行解密计算。

根据 KEK 最小覆盖集的性质，对每个用于重构的属性 $x=\rho(i)$，均存在唯一节点

$$j_i\in\text{Path}(\text{id})\cap\text{Mincs}(U_x)$$

对第 i 行，将普通分量与节点分量合并配对后，行秘密 k_i 被完全消去：

$$\begin{aligned}&e\left(C_i'C_{i,j_i},L\right)e\left(D_i,K_x\right)e\left(D_{i,j_i},P_{x,j_i}\right)\\&=e(g,g)^{a\lambda_i t/z}\end{aligned}$$

因此，DSM 利用 I^{cur} 对应的重构系数 $\{w_i\}$ 计算

$$\begin{aligned}T'&=\prod_{i\in I^{\text{cur}}}\left[e\left(C_i'C_{i,j_i},L\right)e\left(D_i,K_{\rho(i)}\right)e\left(D_{i,j_i},P_{\rho(i)j_i}\right)\right]^{w_i}\\&=e(g,g)^{ast/z}\end{aligned}$$

另一方面，转换密钥中的

$$K=g^{\alpha/z}g^{at/z}$$

与 $C'=g^s$ 进行配对可得

$$\begin{aligned}T''&=e(C',K)=e\left(g^s,g^{\alpha/z}g^{at/z}\right)\\&=e(g,g)^{\alpha s/z}e(g,g)^{ast/z}\end{aligned}$$

于是由 T' 与 T''，DSM 输出

$$T=\frac{T''}{T'}=e(g,g)^{\alpha s/z}$$

用户由 T 利用私钥 $\text{SK}=z$ 计算

$$T^z=e(g,g)^{\alpha s}$$

加密阶段的封装分量为

$$C=R\cdot e(g,g)^{\alpha s}$$

故用户恢复

$$R'=C\left(T^z\right)^{-1}=R$$

计算 $\text{Tag}_0=H_0(R)$，并校验

$$H\left(\text{Tag}_0\ /\!/\ C_M\right)\overset{?}{=}\text{Token}$$

如果 Token 校验通过，计算 $K_E=h(R)$，$\text{msg}=\text{SE.Dec}_{K_E}(C_M)$，输出 msg，正确性得证。□

5.2 方案安全性分析

定理 2 若判定性 q-Parallel BDHE 假设成立，则没有多项式时间的敌手能够攻破 R-CP-ABE 方案，其中挑战矩阵为 $\boldsymbol{M}^*\in\mathbb{Z}_p^{l^*\times n^*}$，$l^*,n^*\leqslant q$。

证明. 假设存在一个多项式时间的敌手 $\mathcal{A}$ 能够在上述安全模型中以不可忽略优势 $\varepsilon=\text{Adv}_{\mathcal{A}}$ 攻破 R-CP-ABE 方案，那么可以构造一个模拟器 $\mathcal{B}$ 能够以不可忽略的优势攻破判定性 q-Parallel BDHE 假设。

初始化阶段. 模拟器 $\mathcal{B}$ 以判定性 q-Parallel BDHE 假设中的 y,Z 为输入，敌手 $\mathcal{A}$ 提交挑战访问结构 $(\boldsymbol{M}^*,\rho^*)$ 给 $\mathcal{B}$。

系统建立阶段. $\mathcal{B}$ 随机选择 $\alpha'\in\mathbb{Z}_p$ 并计算 $e(g,g)^{\alpha}=e(g,g)^{\alpha'}e\left(g^a,g^{a^q}\right)$，即有 $\alpha=\alpha'+a^{q+1}$。设置群元素 $h_1,h_2,\cdots,h_{|W|}$，对于每个 $x(1\leqslant x\leqslant|W|)$，随机选择参数 $z_x\in\mathbb{Z}_p$，若 $\rho^*(i)=x$，即属性 x 对应挑战矩阵中的第 i 行，则设置 $h_x=g^{z_x}\prod_{j=1}^{n^*}\left(g^{a^j/b_i}\right)^{M_{i,j}^*}$，若属性 x 未出现在挑战访问结构中，则设置 $h_x=g^{z_x}$，由于 z_x 是随机的，因此参数 h_x 是随机分布的。模拟器 $\mathcal{B}$ 按照 DSMSetup 算法建立 KEK 二叉树，对于每个属性 x 以及每个 KEK 树节点 j，随机选择 $\theta_{x,j}\in\mathbb{Z}_p^*$，令 $\text{DSK}=\{\theta_{x,j}\}_{x,j}$，这些节点秘密由 DSM 独立产生，与 q-Parallel BDHE 挑战实例无关，因此模拟器可以按照真实算法直接生成。模拟器 $\mathcal{B}$ 按 AASetup 的方式选取 H_0、H、h 与 SE，将系统公钥 $\text{PK}=\left(g,e(g,g)^{\alpha},g^a,h_1,h_2,\cdots,h_{|W|},H_0,H,h,\text{SE}\right)$ 发送给敌手 $\mathcal{A}$，自己保存 α'、DSK、各随机参数和已编程的属性公开参数，用于回答密钥询问。

查询阶段 1. 在该阶段模拟器 $\mathcal{B}$ 回答敌手关于 (id,S) 的私钥问询，其中 S 不满足访问矩阵 $\boldsymbol{M}^*$，id 只用于确定该用户在 KEK 二叉树中的叶节点及 Path(id)。模拟器首先随机选择 $r\in\mathbb{Z}_p,z\in\mathbb{Z}_p^*$，并找到一个向量 $\boldsymbol{w}=\left(w_1,w_2,\cdots,w_{n^*}\right)\in\mathbb{Z}_p^{n^*}$，其中 $w_1=$

-1，并且对于所有满足$\rho^*(i)\in S$的i，有$\boldsymbol{M}_i^*\boldsymbol{w}^{\mathrm{T}}=0$。由LSSS的定义知这样一个向量能在多项式时间内计算得到。模拟器定义变量t为

$$t = r + w_1 a^q + w_2 a^{q-1} + \cdots + w_{n^*} a^{q-n^*+1}$$

计算$L=\left[g^r\prod_{j=1}^{n^*}\left(g^{a^{q+1-j}}\right)^{w_j}\right]^{1/z}$，通过$t$的定义及$w_1=-1$，$g^{at}$中包含了$g^{-a^{q+1}}$项，而$g^{-a^{q+1}}$在假设中并没有给出，但在生成密钥$K$时，由于隐含设置了$\alpha=\alpha'+a^{q+1}$，因此，$g^{-a^{q+1}}$能通过与$g^{\alpha}=g^{\alpha'}g^{a^{q+1}}$相乘而被消去：

$$\begin{aligned}K &= \left[g^{\alpha'}\left(g^a\right)^r\prod_{j=2}^{n^*}\left(g^{a^{q+2-j}}\right)^{w_j}\right]^{1/z}\\ &= g^{\alpha'/z}\left(g^a\right)^{r/z}\prod_{j=2}^{n^*}\left(g^{a^{q+2-j}}\right)^{w_j/z}\end{aligned}$$

对于$\forall x\in S$按下面方法来计算K_x：如果不存在i满足$\rho^*(i)=x$，则令$K_x=L^{z_x}=\left(g^{t/z}\right)^{z_x}=\left(g^{z_x}\right)^{t/z}=h_x^{t/z}$；对于$x\in S$且$x$在访问结构中出现的情形，必须保证$g^{a^{q+1}/b_i}$形式的项都能够被模拟。因为有$\boldsymbol{M}_i^*\boldsymbol{w}^{\mathrm{T}}=0$，所以$g^{a^{q+1}/b_i}$形式的这些项都能够消去。模拟器创建

$$\begin{aligned}K_x &= L^{z_x}\prod_{j=1}^{n^*}\left(g^{a^j/b_i}\right)^{M_{ij}^* r/z}\\ &\cdot\prod_{j=1}^{n^*}\prod_{\substack{k=1\\k\neq j}}^{n^*}\left(g^{a^{q+1+j-k}/b_i}\right)^{M_{ij}^* w_k/z}\end{aligned}$$

将$h_x^{t/z}$按t的定义展开，指数中$\frac{a^{q+1}}{b_i}$项的系数为$\frac{\sum_{j=1}^{n^*}M_{ij}^*w_j}{z}=\frac{\mathbf{M}_i^*\mathbf{w}^T}{z}=0$，其余项的幂次$q+1+j-k$（$j\neq k$）均在实例给出的范围内，故上式等于$h_x^{t/z}$。模拟器$\mathcal{B}$同时运行路径元生成算法DSMPathGen，利用id确定Path(id)。由于所有节点秘密$\theta_{x,j}$都由模拟器自己产生，因此可以直接计算$P_{\mathrm{id},x,j}=L^{1/\theta_{x,j}}=g^{t/\left(z\theta_{x,j}\right)}$，其中$x\in S$，$j\in\mathrm{Path}(\mathrm{id})$。

最终模拟器向敌手返回$\mathrm{TK}=\left(K,L,\{K_x\}_{x\in S}\right)$,$\mathrm{SK}=z$以及路径元$\{P_{\mathrm{id},x,j}\}_{x\in S,j\in\mathrm{Path}(\mathrm{id})}$；上述结果与真实AAKeyGen和DSMPathGen的输出具有相同分布。

对更新询问UpdQuery(CT,Hdr,x,id,op)，模拟器先按op将id移出或加入属性群U_x，并维护属性群状态$\{U_x\}$供挑战阶段使用。若(CT,Hdr)非挑战密文，模拟器持有全部公开参数与DSK，按真实算法执行DSMRevoke或DSMGrant。若为挑战密文（仅在查询阶段2出现）：DSMRevoke只需$g^{-ak_i^*}$、新的Δ_i、$r_{i,j}$与$\theta_{x,j}$，其中k_i^*与$\theta_{x,j}$由模拟器在挑战阶段与系统建立阶段自行产生；DSMGrant的步骤(a)只用到公开参数$e(g,g)^{\alpha}$、g、g^a、$\{h_x\}$与模拟器自选的$\mathbf{v}''$、$\{r_i''\}$，步骤(b)与DSMRevoke相同。两种更新均不涉及q-Parallel BDHE实例中的未知元素，模拟器可按真实算法执行，且所用随机量与真实算法同分布。

挑战阶段. 敌手提交2个消息$\mathrm{msg}_0,\mathrm{msg}_1$给模拟器$\mathcal{B}$；模拟器随机选择$b\in\{0,1\}$，令会话秘密$R^*\in G_T$，根据$q$-Parallel BDHE实例中的$g^s$，并计算$C'^*=g^s$；$\mathcal{B}$随机选择$y_2',y_3',\cdots,y_{n^*}'$，模拟器隐式定义共享向量

$$\boldsymbol{v}^*=\left(s,sa+y_2',sa^2+y_3',\cdots,sa^{n^*-1}+y_{n^*}'\right)$$

来共享秘密。对于挑战矩阵$\boldsymbol{M}^*$的第i行，定义其秘密份额为$\lambda_i^*=\boldsymbol{M}_i^*\boldsymbol{v}^*$。此外，对于每一个挑战矩阵行$i=1,\cdots,l^*$，模拟器随机选择$r_i'\in\mathbb{Z}_p$，并隐式定义$r_i^*=sb_i+r_i'$。由于$q$-Parallel BDHE实例给出了$g^{sb_i}$，模拟器能够计算

$$D_i^*=g^{sb_i}g^{r_i'}=g^{r_i^*}$$

记$x_i=\rho^*(i)$，根据系统建立阶段对挑战属性参数的设置，$h_{x_i}=g^{z_{x_i}}\prod_{j=1}^{n^*}\left(g^{a^j/b_i}\right)^{M_{ij}^*}$，模拟器将$\lambda_i^*=\mathbf{M}_i^*\mathbf{v}^*$与$r_i^*=sb_i+r_i'$代入$C_i^*=g^{a\lambda_i^*}h_{x_i}^{-r_i^*}$，其中$g^{\sum_j M_{ij}^*a^j s}$与$h_{x_i}$编程式中的$\prod_j\left(g^{a^j/b_i}\right)^{-M_{ij}^*sb_i}$相消，可得

$$C_i^*=h_{x_i}^{-r_i'}\left(g^{sb_i}\right)^{-z_{x_i}}\prod_{j=2}^{n^*}\left(g^a\right)^{M_{ij}^*y_j'}$$

因此$\left(C_i^*,D_i^*\right)$与真实Encrypt算法产生的$\left(g^{a\lambda_i}h_{\rho(i)}^{-r_i},g^{r_i}\right)$具有完全相同的分布。根据系统建立阶段的隐式设置$\alpha=\alpha'+a^{q+1}$，设$q$-Parallel BDHE实例中的目标元素为$Z$，模拟器计算$C^*=R^*\cdot e\left(g^s,g^{\alpha'}\right)\cdot Z$，$\mathrm{Tag}_0^*=H_0\left(R^*\right)$，$K_E^*=h\left(R^*\right)$，$C_M^*=$

$\mathrm{SE.Enc}_{K_E^*}(\mathrm{msg}_b)$，以及 $\mathrm{Token}^* = H(\mathrm{Tag}_0^* \,//\, C_M^*)$。因此，模拟器得到中间挑战密文

$$CT_{\mathrm{int}}^* = \left(C^*, C_M^*, C^*, \{C_i^*, D_i^*\}_{i=1}^{l^*}, \mathrm{Token}^*, (\boldsymbol{M}^*, \rho^*)\right)$$

模拟器以 CT_{int}^* 作为输入，按照当前真实方案继续模拟 DSM 的 DSMEncrypt 算法。对于每一行 $i = 1, \cdots, l^*$，模拟器独立随机选择 $k_i^* \in \mathbb{Z}_p$，根据 DSMEncrypt 算法，将中间密文中的行分量 C_i^* 更新为 $\hat{C}_i^* = C_i^* (g^a)^{k_i^*}$，即 $\hat{C}_i^* = g^{a(\lambda_i^* + k_i^*)} h_{\rho^*(i)}^{-r_i^*}$。令 $x_i = \rho^*(i)$，根据挑战时刻固定的属性群状态 $U_{x_i}^*$，模拟器计算其最小覆盖集 $\mathrm{Mincs}(U_{x_i}^*)$，对于每一个 $j \in \mathrm{Mincs}(U_{x_i}^*)$，模拟器独立随机选择 $r_{i,j}^* \in \mathbb{Z}_p$，利用 DSMSetup 阶段产生并保存的节点秘密 $\theta_{x_i j}$，计算 $C_{i,j}^* = (g^a)^{-k_i^*} g^{-r_{i,j}^* / \theta_{x_i,j}}$，以及 $D_{i,j}^* = g^{r_{i,j}^*}$。因此挑战头信息为：

$$\mathrm{Hdr}^* = \left\{ \left(i, j, C_{i,j}^*, D_{i,j}^*\right) : 1 \leqslant i \leqslant l^*, j \in \mathrm{Mincs}\left(U_{\rho^*(i)}^*\right) \right\}$$

需要指出，以上节点分量生成过程中涉及的 $k_i^*, r_{i,j}^*, \theta_{x_i,j}$ 全部由模拟器自己随机产生，而 g^a 属于系统公钥，因此 DSMEncrypt 的整个过程不需要知道 g^a，也不需要额外调用 q-Parallel BDHE 实例中的任何未知元素。换言之，一旦模拟器成功构造了中间密文 CT_{int}^*，就可以完全按照 DSMEncrypt 算法继续生成最终密文：

$$CT^* = \left(C^*, C_M^*, C^*, \{\hat{C}_i^*, D_i^*\}_{i=1}^{l^*}, \mathrm{Token}^*, (\boldsymbol{M}^*, \rho^*)\right)$$

模拟器 $\mathcal{B}$ 将密文 CT^* 和头信息 Hdr^* 返回给敌手 $\mathcal{A}$。

查询阶段 2. 和阶段 1 类似，敌手 $\mathcal{A}$ 继续提交密钥询问与更新询问，模拟器按阶段 1 的方式应答，其中对挑战密文的更新询问按阶段 1 所述处理；限制条件与阶段 1 相同。

猜测阶段. 敌手 $\mathcal{A}$ 最终输出一个对 b 的猜测 b'；如果 $b' = b$，模拟器输出 0 来猜测 $Z = e(g,g)^{a^{q+1}s}$，否则输出 1 表示 Z 是群 G_T 中的随机元素。设敌手 $\mathcal{A}$ 的优势为 $\mathrm{Adv}_{\mathcal{A}} = \left| \Pr[b' = b] - 1/2 \right| = \varepsilon$，不失一般性设 $\Pr[b' = b] = 1/2 + \varepsilon$。当 $Z = e(g,g)^{a^{q+1}s}$ 时，$\mathcal{B}$ 对 $\mathcal{A}$ 给出的是真实游戏的完美模拟，故

$$\Pr\left[\mathcal{B}\left(y, Z = e(g,g)^{a^{q+1}s}\right) | 0\right] = \Pr[b' = b] = 1/2 + \varepsilon$$

当 Z 为 G_T 中的随机元素时，$C^* = R^* \cdot e(g^s, g^{\alpha'}) \cdot Z$ 在 G_T 中均匀分布且与 R^* 独立。此时挑战密文中与 R^* 相关的只有 $\mathrm{Tag}_0^* = H_0(R^*)$；$\mathrm{Token}^*$ 由公开的 Tag_0^* 与 C_M^* 计算得到，不再泄露 R^* 的信息。由于 R^* 在 G_T 中均匀选取，给定 Tag_0^* 后其平均最小熵至少为 $\log p - \mathrm{l}_{H_0}$，又 h 取自两两独立哈希函数族 $\mathcal{H}$，当 $\log p - \mathrm{l}_{H_0} \geqslant \mathrm{l}_{\mathrm{SE}} + 2\log(1/\delta)$ 时，由剩余哈希引理，$K_E^* = h(R^*)$ 与均匀分布的统计距离不超过 δ。在本文参数下 $512 - 128 \geqslant 128 + 256$，取 $\delta = 2^{-128}$。再由 SE 的一次性语义安全性，C_M^* 不泄露关于 b 的不可忽略信息，因此

$$\Pr\left[\mathcal{B}(y, Z \leftarrow G_T) = 0\right] = 1/2 + \mathrm{negl}(\lambda)$$

由 2.2 节中判定性 q-Parallel BDHE 假设的优势定义及上述两个概率，可得

$$\mathrm{Adv}_{\mathcal{B}} = \left| \left(\frac{1}{2} + \varepsilon\right) - \left(\frac{1}{2} + \mathrm{negl}(\lambda)\right) \right| = \varepsilon - \mathrm{negl}(\lambda)$$

由于敌手 $\mathcal{A}$ 的优势 ε 是不可忽略的，因此模拟器 $\mathcal{B}$ 的优势也是不可忽略的，即可以构造模拟器 $\mathcal{B}$ 以不可忽略的优势攻破判定性 q-Parallel BDHE 困难问题。因此没有多项式时间的敌手能够攻破 R-CP-ABE 方案。□

5.3　撤销有效性与抗合谋性

命题 1（属性撤销有效性与抗合谋性）设用户 u 的属性集合满足 $S_u \vDash (\mathbf{M}, \rho)$ 且 $S_u \setminus \{x\} \nvDash (\mathbf{M}, \rho)$，DSM 执行 DSMRevoke 撤销 u 的属性 x 后：

1）u 不能解密更新后的密文；

2）即使仍拥有属性 x 的合法用户 v 向 u 提供其关于 x 的密钥材料，u 也不能将其与自身其他属性密钥组合完成解密。

证明.　下面分别证明上述两个性质：

1）撤销属性 x 后，原可解密用户不能再解密当前密文。

设访问矩阵第 i 行对应属性 $\rho(i) = x$，在撤销前，由于 $u \in U_x$，根据最小覆盖集的性质，存在唯一节点 $j \in \mathrm{Path}(u) \cap \mathrm{Mincs}(U_x)$，因此用户可以利用其路径元和当前节点分量消去第 i 行中引入的秘密 k_i，该行能够正常参与 LSSS 重构。

属性 x 被撤销后，有 $u \notin U_x^{\mathrm{new}}$，DSM 根据新的属性群重新计算 $\mathrm{Mincs}(U_x^{\mathrm{new}})$，于是由最小覆盖集性质，有 $\mathrm{Path}(u) \cap \mathrm{Mincs}(U_x^{\mathrm{new}}) = \varnothing$；与此同时，

DSMRevoke随机选择$\Delta_i \in \mathbb{Z}_p$，并将行秘密更新为$k_i^{\text{new}} = k_i + \Delta_i$，从而$C_i^{'\text{new}} = g^{a(\lambda_i + k_i + \Delta_i)} h_x^{-r_i}$，然后根据新的$\text{Mincs}\left(U_x^{\text{new}}\right)$重新生成节点分量，由于$\text{Path}(u) \cap \text{Mincs}\left(U_x^{\text{new}}\right) = \varnothing$，用户$u$原来持有的路径元不存在与当前密文头相匹配的节点。因此用户无法得到用于消去k_i^{new}的合法节点项，属性x对应的第i行不能为其提供有效的LSSS份额。

即使用户保留撤销前该行的旧节点分量，该旧节点分量也只能消去原来的k_i。令$\tau_u = t_u / z_u$，利用更新后的当前行与用户原有密钥可得

$$e\left(C_i^{'\text{new}}, L_u\right) e\left(D_i, K_{u,x}\right) = e(g,g)^{a(\lambda_i + k_i + \Delta_i)\tau_u}$$

而撤销前旧节点分量产生的消去项为

$$e\left(C_{i,j}^{\text{old}}, L_u\right) e\left(D_{i,j}^{\text{old}}, P_{u,x,j}\right) = e(g,g)^{-ak_i\tau_u}。$$

将两式相乘得到

$$e(g,g)^{a(\lambda_i + k_i + \Delta_i)\tau_u} e(g,g)^{-ak_i\tau_u} = e(g,g)^{a\lambda_i\tau_u} \cdot e(g,g)^{a\Delta_i\tau_u}$$

因此，与正确的LSSS行份额$e(g,g)^{a\lambda_i\tau_u}$相比，仍然留下残余因子$\Gamma_i^{\text{old}} = e(g,g)^{a\Delta_i\tau_u}$。由于$\Delta_i$独立均匀随机选取，在非退化参数条件下，有$\Pr\left[\Gamma_i^{\text{old}} = 1_{G_T}\right] \leqslant 1/p$；因此除可忽略概率外，该残余因子不等于群单位元，撤销前的旧节点分量不能与更新后的当前密文行组合得到正确的LSSS行份额。又根据$S_u \setminus \{x\} \models (\boldsymbol{M}, \rho)$，所以失去属性$x$对应的有效行后，用户$u$的剩余属性不足以完成LSSS重构，因此用户不能再解密更新后的当前密文。

2）另一用户提供属性x的密钥仍不能完成解密。

设另一合法用户v当前仍具有属性x，为简化表示，令$\tau_u = t_u / z_u, \tau_v = t_v / z_v$，因此用户$u$的相关转换密钥分量为$L_u = g^{\tau_u}$，而用户$v$关于属性$x$的密钥材料为$K_{v,x} = h_x^{\tau_v}$，以及对某一当前合法覆盖节点$j$的路径元$P_{v,x,j} = g^{\tau_v / \theta_{x,j}}$。假设用户$v$将属性$x$相关的$K_{v,x}$、$P_{v,x,j}$提供给已被撤销属性$x$的用户$u$，用户$u$在其他属性上使用的转换密钥仍然绑定于自己的$L_u = g^{\tau_u}$；若用户$u$将$v$的$K_{v,x}$与自己的$L_u$用于属性$x$对应的原始密文行，则有

$$e\left(C_i^{'\text{new}}, L_u\right) e\left(D_i, K_{v,x}\right) = e(g,g)^{a\left(\lambda_i + k_i^{\text{new}}\right)\tau_u} \cdot e\left(g, h_x\right)^{r_i(\tau_v - \tau_u)};$$

另一方面，如果再借用用户v的路径元，则节点部分为

$$e\left(C_{i,j}^{\text{new}}, L_u\right) e\left(D_{i,j}^{\text{new}}, P_{v,x,j}\right) = e(g,g)^{-ak_i^{\text{new}}\tau_u} \cdot e(g,g)^{r_{i,j}(\tau_v - \tau_u)/\theta_{x,j}},$$

将两式相乘以后，虽然k_i^{new}可以消去，但仍然存在由$\tau_v - \tau_u$产生的残余因子

$$\Gamma_{u,v} = e\left(g, h_x\right)^{r_i(\tau_v - \tau_u)} e(g,g)^{r_{i,j}(\tau_v - \tau_u)/\theta_{x,j}}$$

记$h_x = g^{\xi_x}$，则$\Gamma_{u,v} = e(g,g)^{(\tau_v - \tau_u)\left(\xi_x r_i + r_{i,j}/\theta_{x,j}\right)}$

$\Gamma_{u,v} = 1_{G_T}$当且仅当$\tau_v = \tau_u$或$\xi_x r_i + \dfrac{r_{i,j}}{\theta_{x,j}} = 0$。$\tau_u$、$\tau_v$由不同用户的$t$、$z$独立产生，$\Pr\left[\tau_v = \tau_u\right] \leqslant \dfrac{1}{p}$；$r_{i,j}$独立于$\xi_x$、$r_i$、$\theta_{x,j}$均匀选取，$\Pr\left[\xi_x r_i + r_{i,j}/\theta_{x,j} = 0\right] \leqslant \dfrac{1}{p}$，故$\Pr\left[\Gamma_{u,v} = 1_{G_T}\right] \leqslant \dfrac{2}{p}$，$\Gamma_{u,v} = 1_{G_T}$仅以可忽略概率发生，因此该合谋攻击不能得到正确的属性行份额。从实际DSMDecrypt算法来看，这种拼接还会被路径元一致性检查发现。对于用户v的路径元，有

$$P_{v,x,j}^{\theta_{x,j}} = g^{\tau_v} = L_v$$

而用户u提交的主转换分量为$L_u = g^{\tau_u}$，由于不同用户的随机量独立产生，除可忽略概率外有$L_v \neq L_u$，因此$P_{v,x,j}^{\theta_{x,j}} \neq L_u$，DSM会拒绝该路径元。需要指出，用户持有转换密钥、私钥与路径元后可在本地自行完成全部配对运算，因此上述结论并不依赖DSM的一致性检查，而是由残余因子$\Gamma_{u,v}$不可消去以及下述不同用户的行份额无法合并所保证。此外，即使敌手不将用户v的属性密钥材料与L_u混合，而是对属性x对应的整个密文行均采用用户v的$L_v, K_{v,x}, P_{v,x,j}$，该行得到的有效份额也绑定于τ_v，即为$e(g,g)^{a\lambda_i\tau_v}$，而用户$u$的其他属性行仍绑定于$\tau_u$。由于不同用户的随机量彼此独立，来自不同用户的行份额不能共同重构为同一随机量τ下的$e(g,g)^{as\tau}$，因此也不能与某一单一用户的转换密钥和私钥SK配合完成最终解密。

由此可知，属性x的密钥材料不能从用户v的

转换密钥中抽出后，与用户 u 的其他属性密钥自由组合。其根本原因在于，一个合法用户的全部属性密钥及路径元都统一绑定于同一个随机量 t/z。不同用户的随机量相互独立。因此，即使其他合法用户将自己关于属性 x 的密钥材料提供给被撤销用户，也不能使该用户重新获得利用属性 x 解密当前密文的能力。

综上，本文方案实现了属性级用户撤销，并能够抵抗上述用户间的属性密钥合谋攻击。□

5.4 抗跨版本密文拼接攻击

记密文 (CT,Hdr) 自生成起经各次 DSMRevoke 与 DSMGrant 产生的全部版本为 $\left\{\left(CT^{(v)},Hdr^{(v)}\right)\right\}_{v=0}^{N}$，其中版本 0 为 DSMEncrypt 的输出；每次 DSMGrant 开始一个新的时期，时期内各版本由 DSMRevoke 产生。此处的版本仅指密文更新后的历史快照，用于安全分析，方案本身不维护版本信息。对用户 u 与版本 v，记版本 v 下属性群 U_x 的最小覆盖集为 $Mincs^{(v)}\left(U_x\right)$，并令

$$I_u^{(v)}=\left\{i:\rho(i)\in S_u,\ Path(u)\cap Mincs^{(v)}\left(U_{\rho(i)}\right)\neq\varnothing\right\}$$

即版本 v 下用户 u 既持有属性又为相应属性群成员的行集合。

命题 2（跨版本密文抗拼接性） 设用户集合 $\mathcal{C}$ 中每个用户 u 在每个版本 $v\in\{0,\cdots,N\}$ 下均满足 $\left\{\rho(i):i\in I_u^{(v)}\right\}\models(\mathbf{M},\rho)$。则在一般双线性群模型下，持有 $\mathcal{C}$ 中全部用户的 TK、SK、路径元以及全部版本 $\left\{\left(CT^{(v)},Hdr^{(v)}\right)\right\}_{v=0}^{N}$ 的敌手，恢复会话秘密 R 的概率不超过 $O\left(q^2/p\right)$，其中 q 为敌手进行的群运算与配对运算总数。

证明． 令 $R=e(g,g)^{\omega}$，$h_x=g^{\xi_x}$，$\tau_u=\dfrac{t_u}{z_u}$，$\bar{\theta}_{x,j}=\dfrac{1}{\theta_{x,j}}$，将 ω、α、a、ξ_x、τ_u、$\bar{\theta}_{x,j}$ 及各版本的 $s^{(v)}$、$y_c^{(v)}$、$k_i^{(v)}$、$r_i^{(v)}$、$r_{i,j}^{(v)}$ 视为形式变量（z_u 为敌手已知标量），则敌手持有的每个 G 中元素的指数都是这些变量的多项式，G_T 中元素 $C^{(v)}$ 与 $e(g,g)^{\alpha}$ 的指数分别为 $\omega+\alpha s^{(v)}$ 与 α。由 DSMRevoke 与 DSMGrant 的定义，同一时期内各版本共享同一组 s、y_c，且只发生撤销；$k_i^{(v')}$ 与 $k_i^{(v'')}$ 为同一变量当且仅当两版本之间行 i 未被改动，此时该行的全部分量在两版本中相同，且

$$Mincs^{(v')}\left(U_{\rho(i)}\right)=Mincs^{(v'')}\left(U_{\rho(i)}\right)\quad(1)$$

恢复 R 即计算指数为 ω 的元素。在一般双线性群模型下，敌手可得的 G_T 元素的指数是 $C^{(v)}$、$e(g,g)^{\alpha}$ 的指数与 G 中元素指数两两乘积的 $\mathbb{Z}_p$-线性组合；只需证明不存在等于单项式 ω 的组合，再由 Schwartz–Zippel 引理即得 $O\left(q^2/p\right)$ 的概率界。假设存在这样的组合。

ω 只出现在 $C^{(v)}$ 的指数中，故组合中必以非零系数使用了某个 $C^{(v)}$，其附带的单项式 $\alpha s^{(v)}$ 须由其他项消去。G 中含 α 的元素只有 K_u（指数 $\dfrac{\alpha}{z_u}+a\tau_u$），$e(g,g)^{\alpha}$ 的指数不含 $s^{(v)}$，故只能将 K_u 与 $C^{(v)}$（指数 $s^{(v)}$）配对，即组合中必有某用户 u 以非零系数使用了 $e\left(K_u,C^{(v)}\right)$，并须消去由此引入的 $a\tau_u s^{(v)}$。不同用户的项含不同的 τ_u，互不相消，以下固定 u。含 τ_u 且能产生 $a\tau_u s^{(\cdot)}$ 的配对只有 $e\left(L_u,C_i'^{(v')}\right)$，指数为

$$a\tau_u\lambda_i^{(v')}+a\tau_u k_i^{(v')}-\xi_{\rho(i)}r_i^{(v')}\tau_u\quad(2)$$

其中 $\lambda_i^{(v')}=M_{i,1}s^{(v')}+\sum_{c\geqslant 2}M_{i,c}\,y_c^{(v')}$。设 $c_{i,v'}$ 为组合中 $e\left(L_u,C_i'^{(v')}\right)$ 的净系数（行 i 未被改动的各版本中 C_i' 为同一元素，按同一元素计）。对每个 $c_{i,v'}\neq 0$ 的行 i，式(2)引入的两个附加单项式各给出一个条件：$\xi_{\rho(i)}r_i^{(v')}\tau_u$ 只能由 $e\left(D_i^{(v')},K_{u,\rho(i)}\right)$ 消去，故 $\rho(i)\in S_u$；$a\tau_u k_i^{(v')}$ 只能由 $e\left(C_{i,j}^{(v'')},L_u\right)$ 消去且须 $k_i^{(v'')}=k_i^{(v')}$，它引入的 $r_{i,j}^{(v'')}\tau_u\bar{\theta}_{\rho(i),j}$ 又只能由 $e\left(D_{i,j}^{(v'')},P_{u,\rho(i),j}\right)$ 消去，故 $j\in Path(u)$，而 $j\in Mincs^{(v'')}\left(U_{\rho(i)}\right)=Mincs^{(v')}\left(U_{\rho(i)}\right)$（由式 (1)），两个条件合起来即 $i\in I_u^{(v')}$。

最后，记版本 v 所在的时期为 E，并记 $\mathbf{e}_1=(1,0,\cdots,0)\in\mathbb{Z}_p^n$，其他时期的行含不同的 s、y_c 变量，对 $a\tau_u s^{(v)}$ 的消去没有贡献，因此只需考察时期 E 内的版本拼接。时期 E 内各版本的 y_c 为同一变量，故式 (2)中 $a\tau_u y_c$（$c\geqslant 2$）的系数须按时期汇总为零：

$$\sum_{v'\in E}\sum_i c_{i,v'}M_{i,c}=0;$$

而要消去 $a\tau_u s^{(v)}$，须

表2　本文与相关方案功能特征对比

对比方案	访问结构	安全假设	安全模型	属性撤销			属性授予	外包解密	结果校验	抗撤销用户合谋
				用户材料更新	密文更新范围	每用户状态				
文献[20]	访问树	DBDH	标准模型	否	全密文	无	更新密文	×	×	×
文献[21]	LSSS	—	无形式化证明	否	封装层与相关行	无	更新密文	√	×	×
文献[22]	访问树	CDH	随机预言机模型	是	全密文	有	重发用户 KEK	×	×	√
文献[23]	LSSS	CDH	标准模型	是	全密文	有	不支持	√	×	√
文献[24]	LSSS	q-1	随机预言机模型	是	局部	有	直接签发密钥	√	×	√
文献[25]	LSSS	—	一般双线性群模型	否	全密文	无	不支持	×	√	√
文献[31]	LSSS	q-DPBDHE2	随机预言机模型	是	局部	有	直接签发密钥	√	√	√
文献[32]	LSSS	q-1	标准模型	—	—	—	不支持	√	√	—
本文	LSSS	q-Parallel BDHE	标准模型	否	局部	无	更新密文	√	√	√

$\sum_{v'\in E}\sum_i c_{i,v'}M_{i,1}\neq 0$。于是 $\sum_{v'\in E}\sum_{i\in I_u^{(v')}} c_{i,v'}\mathbf{M}_i$ 为 $\mathbf{e}_1$ 的非零倍数。时期 E 内只发生撤销，覆盖集只缩小不扩大，故 $I_u^{(v')}$ 随 v' 单调递减，上式涉及的行均属于该时期首个版本的 $I_u^{(E,0)}$，即 $\mathbf{e}_1$ 属于 $\left\{\mathbf{M}_i: i\in I_u^{(E,0)}\right\}$ 的张成空间，由 LSSS 的性质 $\left\{\rho(i): i\in I_u^{(E,0)}\right\}\vDash(\mathbf{M},\rho)$，与假设矛盾。□

命题2的敌手持有密文的全部历史版本，因此涵盖用户本地缓存旧密文与旧密文头的情形。本方案与文献[18,20-23]相同，不保证已获授权用户在撤销后失去对既有密文及其已获取明文的访问能力。命题1讨论属性撤销后当前版本中相应覆盖资格的失效及用户经转换重新获得 R 的能力的变化；命题2讨论从未在任何版本具备解密资格的用户在持有全部历史密文及合法取得的密钥材料时对跨版本密文拼接攻击的抵抗性。

6　性能分析与实验

6.1　理论分析

6.1.1　功能比较

表2从安全性、撤销机制与功能三方面对本文与相关方案进行比较。文献[20-24,31]与本文为属性级用户撤销；文献[25]以收紧访问策略并重加密密文的方式撤销对单个文件的访问权，撤销粒度与机制均不同；文献[32]为可验证外包方案，不支持撤销。安全模型方面，文献[24]为静态安全，文献[32]为选择性安全，文献[25]为自适应多密文安全；文献[25]的机密性由归约至其基础方案获得，该基础方案在一般双线性群模型下证明，未归约到具体困难性假设，故表2中其安全假设一列记为“—”。表中“用户材料更新”指撤销时系统是否需要为合法用户重新生成并重发长期密钥材料，“每用户状态”指撤销执行方是否需要维护用户特定的撤销相关材料。

可以看出，文献[20-21]撤销时无需更新用户材料，但属性群密钥以标量形式交付用户，不能抵抗撤销用户与未撤销用户的合谋；文献[22-23]以用户专属KEK抵抗该类合谋，但撤销时需为合法用户更新KEK并维护每用户状态；文献[31]同样需要向合法用户分发更新密钥。本文以与用户转换密钥绑定的路径元实现撤销，在表2所列方案中本文同时实现了撤销时不更新用户材料、不维护每用户状态并抵抗撤销用户与未撤销用户的合谋；此外，本文支持属性授予、外包解密及对外包结果的校验。文献[24,31]为多授权方案，其计算量按单一授权机构折算后列入表3比较。

6.1.2　性能比较

在进行性能比较前先给出各符号所代表的含义，$|W|$ 是系统属性集的大小，$|G|,|G_T|,|\mathbb{Z}_p|$ 分别代表 $G,G_T,\mathbb{Z}_p$ 中元素的长度，$|S|$ 是系统中用户拥有的

属性数，$|I|$ 是参与重构的属性行数，l 是与密文相关的属性数，$\tilde{l}$ 为文献[25]撤销时新增访问结构的行数，N 是系统中用户总数，$\log N$ 为KEK树的

表3 本文与相关方案计算量比较

对比方案	加密代价	解密代价	撤销代价	
			密文侧	用户相关
文献[20]	$(l+2)T_{\exp}$	$(\lvert I\rvert+1)T_{\text{pair}}+\lvert I\rvert T_{\exp}$	$(l+3)T_{\exp}$	0
文献[21]	$(2l+2)T_{\exp}$	$2T_{\text{pair}}+T_{\exp}$	$6T_{\exp}$	$\lvert U_x\rvert T_{\exp}$
文献[22]	$(2l+2)T_{\exp}+lT_H$	$(3\lvert I\rvert+1)T_{\text{pair}}+\lvert I\rvert T_{\exp}$	$\left(2l+4+\lvert\text{Mincs}(U_x)\rvert\right)T_{\exp}$	$\left(2\lvert U_x\rvert+1\right)T_{\exp}$
文献[23]	$(2l+2)T_{\exp}$	$T_{\exp}$	$\left(l+4+\lvert\text{Mincs}(U_x)\rvert\right)T_{\exp}$	$\left(2\lvert U_x\rvert+1\right)T_{\exp}$
文献[24]	$(5l+1)T_{\exp}$	$(2l+2\lvert S\rvert+1)T_{\exp}$	$T_{\exp}$	$\lvert U_x\rvert T_{\exp}$
文献[25]	$(2l+5)T_{\exp}+(l+3)T_H$	$3T_{pair}+2T_{\exp}+2T_H$	$\left(2(l+\tilde{l})+2\right)T_{\exp}$ $+(l+\tilde{l}+1)T_H$	0
文献[31]	$(6l+1)T_{\exp}+(l+1)T_H$	$T_{\exp}+T_H$	T_{pair}	$2\lvert U_x\rvert T_{\exp}$
文献[32]	$(3l+5)T_{\exp}+T_H$	$2T_{\exp}+T_H$	—	—
本文	$(3l+2)T_{\exp}+3T_H$	$T_{\exp}+3T_H$	$\left(2+2\lvert\text{Mincs}(U_x)\rvert\right)T_{\exp}$	0

加密代价仅统计数据拥有者端开销，解密代价仅统计用户端开销；撤销代价的密文侧为每个受影响密文的开销，用户相关为每次撤销中与用户材料相关的开销。

高度，每个用户的路径节点数为$\log N+1$，$\lvert\text{Mincs}(U_x)\rvert$为属性$x$的最小覆盖集大小，$\lvert U_x\rvert$为属性群的成员数，$T_{\exp}$、$T_{\text{pair}}$和$T_H$分别代表指数运算时间、配对运算时间和哈希运算时间，其中文献[22,25,31]的T_H为哈希到群G的运算，其开销与群上指数运算相当。比较时取ρ为单射，撤销中需要的旧秘密均按服务器从密文头恢复计数。

在存储方面，本文的公钥规模为$(\lvert W\rvert+2)\lvert G\rvert+\lvert G_T\rvert$，小于包含逐属性DSM公钥的文献[22-23]。本文的额外开销来自用户绑定路径元与两元素节点分量：用户需为每个属性保存$\log N+1$个路径元，本地密钥材料为$\left(\lvert S\rvert+2+\lvert S\rvert(\log N+1)\right)\lvert G\rvert+\lvert\mathbb{Z}_p\rvert$，当$\lvert S\rvert=10$、$N=2^{20}$时多出210个群元素；密文头为$2\sum_i\lvert\text{Mincs}(U_{\rho(i)})\rvert\lvert G\rvert$，是文献[22-23]的2倍。作为交换，本文DSM不保存任何用户特定的密码材料，而文献[21]需保存用户转换密钥，文献[22-23]需为撤销维护每个用户的专属KEK。

由表3可知，本文用户解密只需1次指数与3次哈希运算，为常数级，仅比文献[23]多3次哈希以获得结果校验能力；数据拥有者加密比文献[21,23]多出l次指数与3次哈希，来自每行随机量D_i与会话秘密的封装。服务器端的重加密与外包转换开销本文较高，均由DSM承担，具体见6.2节实验。文献[24]虽支持外包解密，但其转换密钥由用户针对每个密文生成，需对密文的$2l$个分量与自身$2\lvert S\rvert$个密钥分量逐一盲化，用户端开销随策略规模与属性数增长；文献[32]将转换密钥生成完全前移至密钥生成中心，用户端解密与校验为常数级，与本文同量级，但该方案不支持撤销。

撤销方面，表中两列单位不同：用户相关开销每次撤销发生一次，密文侧开销需乘以受影响的密文数。文献[22-23]每次撤销需为属性群全部剩余成员重新生成并分发KEK，文献[21]的剩余成员需在后续解密前自行更新转换密钥分量，而本文既不向合法用户分发任何密钥材料，也不需要其本地更新，用户相关开销为0；文献[20]该项亦为0，但其群密钥可在用户间转发，不抗撤销用户合谋。文献[31]以属性版本更新实现撤销，密文侧仅需1次配对，开销低于本文，但须为属性群内全部剩余成员分别生成并分发更新密钥；文献[24]的更新密钥为标量形式，亦须由授权机构逐一为剩余成员计算并下发更新后的密钥分量，两者的用户相关开销分别为$2\lvert U_x\rvert T_{\exp}$与$\lvert U_x\rvert T_{\exp}$。文献[25]撤销时不更新任何用户密钥，用户相关开销亦为0，但须由数据拥有者在线生成委托、云端对整条密文重加密，且访

问矩阵随每次撤销单调增长，密文侧开销随策略规模累积；本文密文侧开销与策略规模 l 无关，当 $\left|\mathrm{Mincs}\left(U_x\right)\right| < l+2$ 时低于需全密文再随机化的文献[23]；覆盖集规模取决于成员在KEK树叶节点上的分布方式，见6.2节实验。授予时本文需对存量密文整体再随机化，密文侧开销高于对比方案，但已有成员的材料保持不变。

为量化存储与通信开销，取6.2节实验参数 $|G|=258\text{B}$、$\left|G_T\right|=513\text{B}$、$\left|\mathbb{Z}_p\right|=64\text{B}$，对称密钥 $\left|C_k\right|=16\text{B}$。本文与文献[21,23]的属性密钥均为 $(|S|+2)|G|+\left|\mathbb{Z}_p\right|$，差异在与撤销相关的用户密钥：文献[21]为 $\log N+1$ 个对称路径密钥，文献[23]为每属性两个群元素及节点索引，本文为每属性 $\log N+1$ 个群元素的路径元，三者的用户端存储开销分别为 $(|S|+2)|G|+\left|\mathbb{Z}_p\right|+(\log N+1)\left|C_k\right|$、$(3|S|+2)|G|+(|S|+1)\left|\mathbb{Z}_p\right|$ 与 $(|S|+2)|G|+\left|\mathbb{Z}_p\right|+|S|(\log N+1)|G|$。当 $N=2^{20}$、$|S|=10$ 时，上述开销依次为3.5 KB、9.0 KB与57.3 KB。本文路径元随 $|S|$ 与 $\log N$ 线性增长，其中属性数的影响远大于KEK树高：$|S|$ 加倍时开销加倍，而用户规模由 2^{20} 增至 2^{24} 时仅增加约19%。这是本文使撤销与授予均无需合法用户更新密钥、DSM不维护每用户状态所付出的一次性存储代价。本文转换请求提交 TK 与全部路径元，约57.3 KB，与用户端存储开销同量级，转换结果为一个 G_T 元素；密文头每个覆盖节点占两个群元素，每行大小为 $2\left|\mathrm{Mincs}\left(U_x\right)\right|\cdot|G|$；连续分布下 $|Mincs|=4$ 时为2.1 KB；随机分布下，6.2节实验中覆盖集均值为109～396，对应每行56～204 KB。

综上，本文以用户侧静态路径材料、密文头与服务器端计算为代价，使撤销过程无需触达任何合法用户、用户解密保持常数级；撤销的密文侧计算量仅在覆盖集较小时低于文献[23]。本文适用于用户数量大、终端间歇在线、难以保证合法用户及时同步更新，且属性群成员在KEK树中占据相邻叶节点（如按注册批次或部门分配叶节点）的场景。

6.2　实验仿真

实验平台为macOS（Apple M1 Pro，16 GB 内存），实验程序采用 Python 3.10.9 编写，基于 Charm-Crypto Framework 0.62 和 PBC 1.0.0 实现。由于本文方案及对比方案均基于对称双线性配对 $e: G\times G\to G_T$ 描述，实验选取 PBC 的 Type-A 超奇异曲线 $y^2=x^3+x$，由PBC参数生成函数产生，本文将该组 Type-A 参数记为 A512_2048：群阶 p 为512位素数，基域 F_q 的 q 为2048位素数，嵌入次数为2，目标群位于4096位的 F_{q^2} 中。群阶取512位是为了满足5.2节中提取函数 h 的熵条件；此时曲线侧抵抗Pollard rho攻击的强度为256位，有限域侧按经典数域筛法估计高于128位[35-36]，整体达到128位安全级别。

基础运算时间实测为：G 上指数运算 $T_{\exp}^{G}=9.78\text{ms}$，$G_T$ 上指数运算 $T_{\exp}^{G_T}=1.29\text{ms}$，配对运算 $T_{\text{pair}}=13.36\text{ms}$，$G$ 上乘法 $T_{\text{mul}}=0.011\text{ms}$，均为100次测量的平均值；$G$ 与 G_T 元素的序列化长度分别为258B和513B。各实验中的理论值均由同次运行测得的基础运算时间按表3的计数公式推算。哈希函数 H_0、H 以 SHA-256 实现，提取函数 h 以 HMAC-SHA256 截取128位近似，对称加密不计入计时；各方案在计时前均完成一次正确性自检。

加密与解密实验每个数据点重复20次；撤销实验对每个属性群规模随机生成20组成员分布，用户相关开销重复3次。图中数据为均值，误差棒为95%置信区间。对比方案为文献[21]（马华等，2015）与文献[23]（孙磊等，2019），二者与本文同为单授权、LSSS访问结构并支持外包解密。

访问策略取 l 个属性的与门，l 取10至50，每个属性的最小覆盖集规模取4，每个数据点重复20次，数据拥有者加密开销如图3(a)所示。三种方案的开销均随 l 线性增长，每增加一个属性，文献[21]、文献[23]和本文方案分别增加约20.2ms、19.9ms和29.9ms。文献[21]与文献[23]的数据拥有者加密结构相同，二者曲线基本重合；本文约为其1.5倍，多出的 l 次指数运算来自每行随机量 $D_i=g^{r_i}$，用于生成校验信息的3次哈希运算开销可忽略。

在DSM端，文献[21]的原始加密流程中不存在与本文DSMEncrypt直接对应的上传阶段处理，其成员变更后的密文更新开销见撤销实验。$l=10$ 时，文献[23]与本文的重加密开销分别约为487.7ms和890.1ms，$l=50$ 时分别约为2495.1ms和4477.6ms。本文约为文献[23]的1.8倍，来自每个

覆盖节点的两元素节点分量，该部分由DSM承担。

访问策略采用$|I|$-of-$|I|$门限LSSS，用户属性集满足全部策略行，$|I|$取10至50；文献[21]按已发生一次属性群成员变更后的密文状态实现。每个数据

点重复20次，用户解密开销如图3(b)所示。三种方案的用户解密开销均与$|I|$无关。文献[23]和本文的用户端仅需一次G_T上指数运算，平均开销分别为1.30ms和1.32ms，本文额外的哈希运算用于外包结果校验，引入的开销不超过0.02ms；文献[21]的用户端还需完成两次双线性配对，平均开销约为28.7ms，约为本文的21倍。

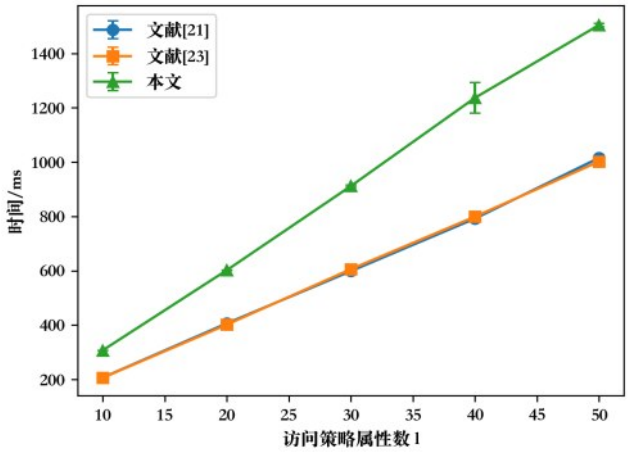
外包转换由服务器承担。$|I|=10$时，文献[21]、文献[23]与本文的转换开销分别约为273.5ms、421.4ms和520.0ms，$|I|=50$时分别约为1390.3ms、2074.1ms和2551.0ms。本文每行与文献[23]同为3次配对运算，多出的开销来自路径元核验所需的1次G上指数运算。

KEK树取$N=2^{10}$个叶节点，$|U_x|$表示撤销后属性群的剩余成员数，取值为128至1000，每个取值重复3次，结果如图4所示。文献[21]的剩余成员需依据新的密文头各自更新一个转换密钥分量，文献[23]需由DSM更新属性群密钥并为每个剩余成员重新计算用户专属KEK，二者的开销均随$|U_x|$线性增长，每增加一名成员分别增加约9.7ms和19.7ms；当$|U_x|=1000$时，分别为9.7s和19.7s。本文方案撤销时既不向合法用户分发任何密钥材料，也不要求其在本地更新，用户相关开销为0，与属性群规模无关。

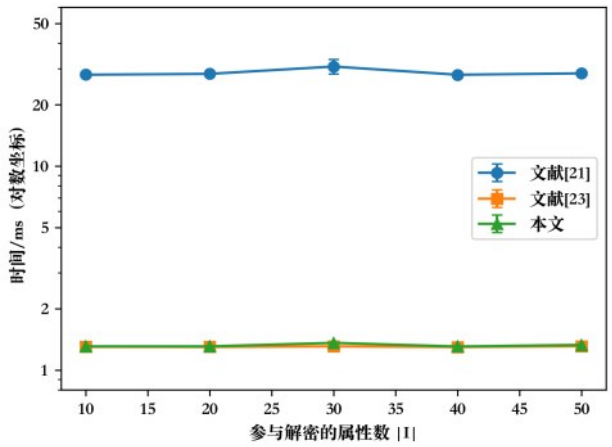
文献[21]的属性群密钥以共享标量形式交付用户，不抵抗撤销用户与未撤销用户的合谋，其撤销所完成的安全目标与本文不同，且其撤销中与用户相关的更新由各剩余成员在后续解密前分别完成。因此，图4保留文献[21]，用以比较撤销过程是否需要剩余用户参与；密文侧撤销开销与多密文总撤销开销仅比较撤销安全目标相同的文献[23]与本文方案。作为参考，文献[21]单个密文的更新开销约为51ms。

实验取$l=20$，由于两种方案的密文侧开销均取决于最小覆盖集规模$c=\left|\mathrm{Mincs}\left(U_x\right)\right|$，为考察属性群成员在KEK树中分布方式对最小覆盖集大小的影响，本文设置两种成员分布情形：随机叶节点分布，即从KEK树的N个叶节点中随机选取$|U_x|+1$个节点作为撤销前的属性群成员；连续叶节点分布，即撤销前的属性群成员占据编号连续的$|U_x|+1$个叶节点，起始位置随机选取且不跨越叶节点编号边界。两种情形下被撤销用户均从撤销前的成员中随机选取，每个$|U_x|$取20组成员分布的平均值，结果如图5所示。

图5(a)(b)为每个受影响密文的更新开销，本文为$(2+2c)T_{\exp}$，文献[23]为$(l+4+c)T_{\exp}$。随机分布下c的平均值为109～396，本文开销为文献[23]的1.7～1.9倍，$|U_x|=640$时分别为7.9 s和4.1 s；连续分布下，被撤销用户在成员区间中留下空



(a)　数据拥有者加密开销随l的变化



(b)　用户解密开销随$|I|$的变化

图3　加密与解密开销

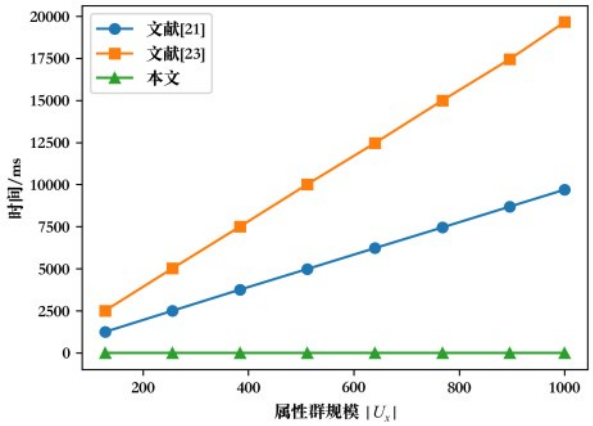
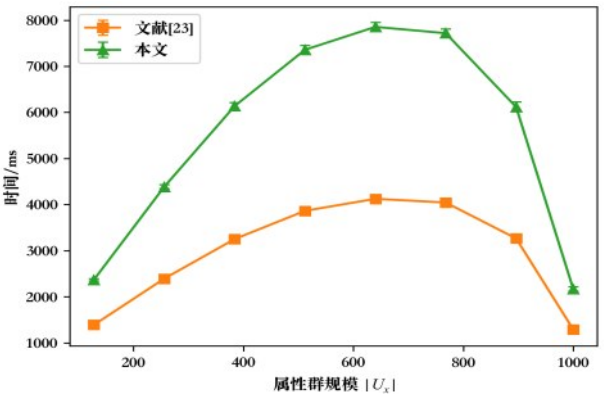


图4　撤销的用户相关开销随$|U_x|$的变化

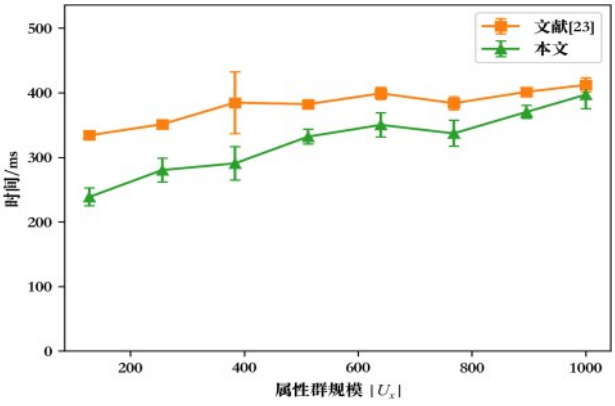
位，c 的平均值为 11～19，本文开销为 239～397ms，为文献[23]的 71%～96%，且随$|U_x|$增大而逐渐接近文献[23]。二者的差异来自本文每个覆盖节点需生成两个群元素，仅当$c < l + 2$时本文的单密文开销更低。

图 5(c)(d)为一次撤销的总开销，即受影响密文数与单密文开销之积加上图 4 所示的用户相关开销，由图 4-图 5 的实测值合成。连续分布下，本文在任意受影响密文数下均低于文献[23]；随机分布下，本文仅当受影响密文数少于交叉点时更低，交叉点随$|U_x|$在 2.5 至 22.4 之间变化，$|U_x| = 512$时约为 2.9，$|U_x| = 1000$时约为 22.4。

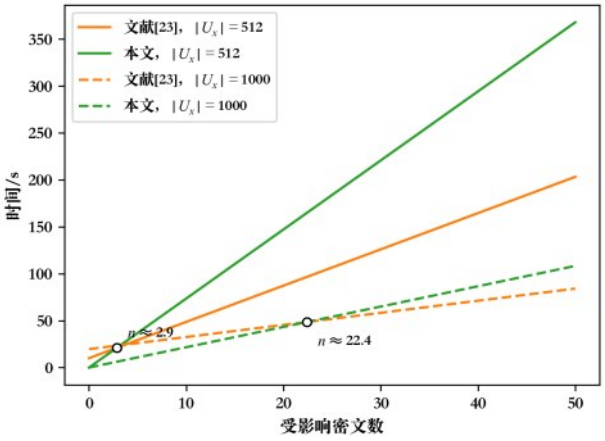
图 4 与图 5 表明，本文撤销的主要优势在于不需要剩余合法用户参与，且与成员分布无关；密文侧计算量的优势则依赖于覆盖集规模，在成员随机分布且受影响密文较多时，本文的密文侧开销高于文献[23]，这是以长期静态路径元替代逐用户密钥更新所付出的代价。叶节点由 DSM 在用户注册时分配，按部门或注册批次为同一属性群的用户分配相邻叶节点即可接近连续分布；若部署中无法控制叶节点分配，密文侧开销将高于文献[23]，但撤销与授予不需要合法用户参与这一优势与成员分布



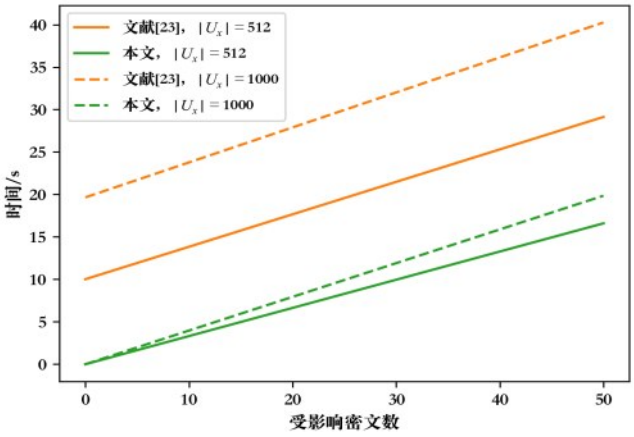
(a)　随机叶节点分布下的单密文撤销开销



(b)　连续叶节点分布下的单密文撤销开销



(c)　随机叶节点分布下的总撤销开销



(d)　连续叶节点分布下的总撤销开销

图5　不同叶节点分布下的撤销开销

无关。

本文实验在单机上完成，所报告的开销为各算法的计算时间，未计入网络传输时延，也未考虑多用户并发请求对DSM密文更新与外包转换的影响；DSM单次转换的收发数据量可由6.1节给出的请求与结果规模推算。多机环境下的端到端时延与DSM并发吞吐评估是进一步的工作。

7 结束语

针对云存储中属性由多个用户共享、撤销单一属性易影响其他用户的问题，本文提出一种支持属性级用户撤销的可验证CP-ABE方案。该方案利用KEK二叉树最小覆盖集与用户绑定的路径元，由DSM在密文侧完成属性撤销，剩余合法用户无需更新密钥，并能抵抗撤销用户与未撤销用户的合谋；同时将配对运算外包并支持结果校验。方案在标准模型下基于判定性q-Parallel BDHE假设证明了安全性，实验表明其用户端解密开销恒定。未来工作将研究多授权中心场景下的属性撤销，降低密文头与路径元开销，并探索自适应安全的实现。

参考文献：

[1] Zhang Y, Deng R H, Xu S, et al. Attribute-based encryption for cloud computing access control: a survey[J]. ACM Computing Surveys, 2020, 53(4): 83.

[2] Ali S, Talpur D B, Abro A, et al. Security and privacy in multi-cloud and hybrid cloud environments: challenges, strategies, and future directions[J]. Computers & Security, 2025, 157: 104599.

[3] Shamir A. Identity-based cryptosystems and signature schemes[C]//Proceedings of Advances in Cryptology (CRYPTO 1984). Berlin: Springer, 1985: 47-53.

[4] Boneh D, Franklin M. Identity-based encryption from the Weil pairing [C]//Proceedings of Advances in Cryptology (CRYPTO 2001). Berlin: Springer, 2001: 213-229.

[5] Sahai A, Waters B. Fuzzy identity-based encryption[C]//Proceedings of Advances in Cryptology (EUROCRYPT 2005). Berlin: Springer, 2005: 457-473.

[6] Ostrovsky R, Sahai A, Waters B. Attribute-based encryption with non-monotonic access structures[C]//Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS). New York: ACM Press, 2007: 195-203.

[7] Staddon J, Golle P, Gagné M, et al. A content-driven access control system[C]//Proceedings of the 7th Symposium on Identity and Trust on the Internet. New York: ACM Press, 2008: 26-35.

[8] Liang X, Lu R, Lin X, et al. Ciphertext policy attribute based encryption with efficient revocation[R]. Waterloo: University of Waterloo, 2011.

[9] Attrapadung N, Imai H. Attribute-based encryption supporting direct/indirect revocation modes[C]//Proceedings of IMA International Conference on Cryptography and Coding. Berlin: Springer, 2009: 278-300.

[10] Wang W, Wang Z, Li B, et al. IR-CP-ABE: identity revocable ciphertext-policy attribute-based encryption for flexible secure group-based communication[EB/OL]. IACR Cryptology ePrint Archive, 2017: 1100.

[11] Xu S, Yang G, Mu Y. Revocable attribute-based encryption with decryption key exposure resistance and ciphertext delegation[J]. Information Sciences, 2019, 479: 116-134.

[12] Dai Y, Xue L, Yang B, et al. A traceable and revocable decentralized attribute-based encryption scheme with fully hidden access policy for cloud-based smart healthcare[J]. Computer Standards & Interfaces, 2025, 92: 103936.

[13] Wang W, Zhong N, Xiao J, et al. Efficient revocable attribute-based encryption with black-box traceability[J]. International Journal of Information Security, 2026, 25: 81.

[14] Pirretti M, Traynor P, McDaniel P, et al. Secure attribute-based systems [C]//Proceedings of the 13th ACM Conference on Computer and Communications Security. New York: ACM Press, 2006: 99-112.

[15] Ibraimi L, Petkovic M, Nikova S, et al. Mediated ciphertext-policy attribute-based encryption and its application[C]//Proceedings of the 10th International Workshop on Information Security Applications (WISA 2009). Berlin: Springer, 2009: 309-323.

[16] Yu S, Wang C, Ren K, et al. Attribute based data sharing with attribute revocation[C]//Proceedings of the ACM Symposium on Information, Computer and Communications Security (ASIACCS 2010). New York: ACM Press, 2010: 261-270.

[17] Yang K, Jia X. Security for Cloud Storage Systems[M]. New York: Springer, 2014.

[18] 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.

[19] Yang K, Jia X, Ren K. Attribute-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. New York: ACM Press, 2013: 523-528.

[20] 闫玺玺, 汤永利. 数据外包环境下一种支持撤销的属性基加密方案[J]. 通信学报, 2015, 36(10): 92-100.
Yan X X, Tang Y L. Attribute-based encryption scheme with efficient revocation in data outsourcing systems[J]. Journal on Communications, 2015, 36(10): 92-100.

[21] 马华, 白翠翠, 李宾, 等. 支持属性撤销和解密外包的属性基加密方案[J]. 西安电子科技大学学报(自然科学版), 2015, 42(6): 6-10.
Ma H, Bai C C, Li B, et al. Attribute-based encryption scheme supporting attribute revocation and decryption outsourcing[J]. Journal of Xidian University (Natural Science), 2015, 42(6): 6-10.

[22] Li J, Yao W, Han J, et al. User collusion avoidance CP-ABE with efficient attribute revocation for cloud storage[J]. IEEE Systems Journal, 2018, 12(2): 1767-1777.

[23] 孙磊, 赵志远, 王建华, 等. 云存储环境下支持属性撤销的属性基加密方案[J]. 通信学报, 2019, 40(5): 47-56.
Sun L, Zhao Z Y, Wang J H, et al. Attribute-based encryption scheme supporting attribute revocation in cloud storage environment[J]. Journal on Communications, 2019, 40(5): 47-56.

[24] Yao Y Q, Zhang Y J, Liu Z Y, et al. Traceable and revocable multi-authority ABE supporting decryption outsourcing and policy update

for cloud data access control[J]. Frontiers of Computer Science, 2026, 20(4): 2004805.

[25] Li Y, Zhu L, Deng Y, et al. Fast revocable attribute-based encryption with data integrity for Internet of Things[J]. Journal of Systems Architecture, 2025, 168: 103551.

[26] Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts[C]//Proceedings of the 20th USENIX Security Symposium. Berkeley: USENIX Association, 2011.

[27] Lai J, Deng R H, Guan C, et al. Attribute-based encryption with verifiable outsourced decryption[J]. IEEE Transactions on Information Forensics and Security, 2013, 8(8): 1343-1354.

[28] Qin B, Deng R H, Liu S, et al. Attribute-based encryption with efficient verifiable outsourced decryption[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(7): 1384-1393.

[29] 仲红, 崔杰, 朱文龙, 等. 高效且可验证的多授权机构属性基加密方案[J]. 软件学报, 2018, 29(7): 2006-2017.
Zhong H, Cui J, Zhu W L, et al. Efficient and verifiable multi-authority attribute-based encryption scheme[J]. Journal of Software, 2018, 29 (7): 2006-2017.

[30] 明洋, 何宝康. 支持属性撤销的可验证外包的多授权属性加密方案[J]. 计算机应用, 2019, 39(12): 3556-3562.
Ming Y, He B K. Multi-authority attribute-based encryption scheme with verifiable outsourcing and attribute revocation[J]. Journal of Computer Applications, 2019, 39(12): 3556-3562.

[31] 王经纬, 吴静雯, 殷新春. 抗共谋攻击的多授权电子健康记录共享方案[J]. 电子学报, 2023, 51(5): 1179-1186.
Wang J W, Wu J W, Yin X C. Collusion-resistant multi-authority electronic health records sharing scheme[J]. Acta Electronica Sinica, 2023, 51(5): 1179-1186.

[32] Hou Z, Ning J, Huang X, et al. Blockchain-based efficient verifiable outsourced attribute-based encryption in cloud[J]. Computer Standards & Interfaces, 2024, 90: 103854.

[33] Waters B. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization[C]//Proceedings of the 14th International Conference on Practice and Theory in Public Key Cryptography (PKC 2011). Berlin: Springer, 2011: 53-70.

[34] Beimel A. Secure schemes for secret sharing and key distribution[D]. Haifa: Technion-Israel Institute of Technology, 1996.

[35] Barbulescu R, Duquesne S. Updating key size estimations for pairings [J]. Journal of Cryptology, 2019, 32(4): 1298-1336.

[36] Barker E. Recommendation for key management: Part 1 – General: NIST SP 800-57 Part 1 Rev. 5[S]. Gaithersburg, MD: National Institute of Standards and Technology, 2020.

李树全（1971– ），男，四川眉山人，电子科技大学高级工程师、硕士生导师，主要研究方向为信息安全与密码学。

佘堃（1967– ），男，四川成都人，博士，电子科技大学教授、博士生导师，主要研究方向为云计算与信息安全。