可扩展且支持多种追踪机制的属性基加密方案

谢晴晴[1]，朱法铜[1]，冯霞[2]
（1.江苏大学计算机科学与通信工程学院，江苏 镇江 212013；2.海南大学网络空间安全学院（密码学院），海南 海口 570228）

摘　要：针对属性基加密方案中恶意用户追踪机制单一的问题，提出一种可扩展且支持多种追踪机制的属性基加密方案。首先，白盒追踪通过短签名实现，黑盒追踪通过将用户信息嵌入密钥和构造追踪密文来实现，提供了完备的恶意用户追踪能力。其次，设计了独立于属性空间的系统初始化机制，支持属性动态扩展而无须系统重置，有效满足了加密系统在大属性空间下的动态扩展需求。最后，安全性分析证明了所提方案具有选择明文攻击下密文的不可区分性、白盒可追踪性和黑盒可追踪性，性能分析证明了其可行性。

关键词：属性基加密；权限泄露；白盒追踪；黑盒追踪
中图分类号：TP309
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025134

Scalable attribute-based encryption scheme with multiple tracing mechanisms

XIE Qingqing[1], ZHU Fatong[1], FENG Xia[2]
1. School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China
2. School of Cyberspace Security (School of Cryptology), Hainan University, Haikou 570228, China

Abstract: To solve the problem of a single malicious user tracing mechanism in attribute-based encryption (ABE) schemes, a scalable ABE scheme with multiple tracing mechanisms was proposed. Firstly, white-box traceability was implemented using short signatures, and black-box traceability was achieved by embedding user information in secret key and constructing tracing ciphertexts, thereby providing a comprehensive capability for tracing malicious users. Secondly, a system initialization mechanism independent of the attribute universe was designed, which supports dynamic attribute expansion without system reset, effectively fulfilling the dynamic expansion requirements of encryption systems in large attribute universes. Finally, security analysis demonstrated that the proposed scheme exhibits ciphertext indistinguishability under chosen-plaintext attacks, white-box traceability, and black-box traceability. Performance analysis verified its feasibility.

Keywords: attribute-based encryption, permission leakage, white-box traceability, black-box traceability

0　引言

随着计算机技术和通信技术的发展，网络应用呈指数级增长态势，深刻改变着社会生产模式与人类生活方式。在此背景下，云存储以其弹性、可扩展、按需付费的特点，已然成为现代网络应用部署的核心基础设施。然而，云服务的可信度和安全性

收稿日期：2025–05–12；**修回日期：**2025–07–21
通信作者：谢晴晴，xieqq@ujs.edu.cn
基金项目：国家自然科学基金资助项目（No.62002139, No.62272203）；国家自然科学基金联合基金资助项目（No.U24A20238）；澳门特别行政区科技发展基金资助项目（No.0004/2023/ITP1）
Foundation Items: The National Natural Science Foundation of China (No.62002139, No.62272203), The Joint Funds of the National Natural Science Foundation of China (No.U24A20238), The Science and Technology Development Fund, Macao SAR (No.0004/2023/ITP1)

引发了用户对于私人敏感数据的安全性担忧。属性基加密（ABE, attribute-based encryption）机制[1-2]因其一对多加密和细粒度数据分享的特性，被认为是外包数据安全分享场景安全机制的最佳选择[3-4]。

区别于其他公钥加密机制[5]，ABE机制提供了更有表达力的细粒度访问控制能力。以密文策略属性基加密（CP-ABE, ciphertext-policy attribute-based encryption）作为云存储服务的底层加密算法为例[2]，数据拥有者指定数据的访问策略，并将绑定访问策略的密文存储在云服务器中，数据用户掌握的密钥代表着密文的访问权限。早期的ABE方案专注于加密方案的构建和访问策略表达能力的提升，数据用户一般被设定为可信或半可信的，并没有考虑到授权用户可能存在的恶意行为。在实际的应用场景中，数据用户很难做到完全可信，存在有意或无意泄露其解密权限的情况，通常这类用户被称为恶意用户。在经典ABE方案中，同一属性往往由多个授权用户共享，这导致不同用户共享对那些相同属性的解密能力。对于截获的密钥，很难根据密钥中的属性信息进行恶意用户追踪，这无形之中助推了用户泄露密钥的想法。针对这一安全隐患，研究者提出可追踪ABE方案[6-7]，以提供恶意用户身份的追踪机制，使恶意用户暴露在被发现、被问责的风险中，授权用户便不会毫无顾忌地滥用密钥。

近年来，研究者已经在可追踪ABE方案领域取得重要进展，但这些工作[6-11]仅支持单一追踪机制，即白盒（WB, white box）追踪或黑盒（BB, black box）追踪。值得警惕的是，恶意用户可以通过2种截然不同的途径非法传播其解密权限：直接暴露其掌握的密钥；售卖一个内置解密算法和密钥的黑盒。然而，黑盒追踪和白盒追踪是基于2种完全不同的方式构建的，白盒追踪通过将身份信息嵌入密钥组件，利用密钥组件直接定位用户身份；黑盒追踪则通过构建追踪密文，分析解密黑盒的输出结果，进而识别恶意用户身份。白盒追踪和黑盒追踪无法或不适合解决彼此的问题。白盒追踪由于无法获得黑盒中的密钥，无法处理黑盒泄露的情况。黑盒追踪虽然能够解决白盒追踪的问题，但是相较于白盒追踪开销过大。黑盒追踪和白盒追踪各有其局限性，仅依赖其中一种追踪机制无法应对复杂的密钥泄露场景。

大属性空间[12-13]和多机构授权[14-15]是研究者设计可追踪ABE方案时需要考虑的重要功能特性。一方面，系统初始化阶段必须预先定义与属性规模数量线性相关的系统参数。静态的属性空间配置和动态的属性支持需求之间存在固有矛盾：预定义的属性空间过小会导致系统频繁初始化，而预定义的属性空间过大时，过量参数预定义将造成资源浪费。针对该可扩展性的不足，Rouselakis等[16]提出大属性空间ABE方案，彻底消除属性相关参数的预定义需求，解决了上述ABE方案缺乏可扩展性的问题。另一方面，经典ABE方案依赖证书授权中心（CA, central authority）生成解密密钥，一旦CA出现单点故障，整个系统的安全性、可用性将面临巨大威胁。多授权机构ABE方案的核心在于消除CA的单点故障风险[17-20]，将CA的授权职能分散到多个授权机构。即便出现部分授权机构受损的情况，其他授权机构仍能正常运转。

为应对复杂多样的恶意用户追踪场景，弥补现有方案在追踪能力上的不足，本文提出一种全新的可扩展且支持多种追踪机制的属性基加密方案，其主要贡献有3个方面。

1) 本文方案兼容白盒追踪和黑盒追踪，完整覆盖2种典型密钥泄露场景，提升了加密系统可追踪性。

2) 本文采用L形判定性并行双线性Diffie-Hellman指数（L-DPBDHE, L-type decisional parallel bilinear Diffie-Hellman exponent）假设证明了方案满足选择明文攻击下密文的不可区分性和黑盒可追踪性，采用*l*-强Diffie-Hellman（*l*-SDH, *l*-strong Diffie-Hellman）假设证明了方案的白盒可追踪性。

3) 与最新的相关工作相比，本文方案考虑了大属性空间和多授权机构的功能特性。大属性空间设置使本文方案具有扩展性，在系统运行过程中能够动态地添加新属性而不需要重新初始化系统。支持多授权机构的设置能有效地解决密钥托管问题，降低对单一权威机构的信任风险。在兼容大属性空间和多授权机构功能的前提下，本文方案的性能开销与最新相关工作相比具有可比性。

1 相关工作

1.1 ABE

2005年，Sahai等[21]提出了模糊身份基加密

（FIBE, fuzzy identity-based encryption）方案。作为ABE的直接理论前驱，FIBE方案为后续ABE体系的构建提供了基础框架。Goyal等[1]首次提出了密钥策略属性基加密（KP-ABE, key-policy attribute-based encryption）方案，Bethencourt等[2]随后提出了密文策略属性基加密CP-ABE方案，自此ABE机制分化为CP-ABE和KP-ABE。文献[2]提出了基于访问树的CP-ABE方案，通过构建包含“与门”“或门”及门限逻辑的树形结构，实现了复杂访问策略的形式化描述。为突破传统方案仅支持单调逻辑的局限，Ostrovsky等[22]提出了一种支持非单调访问结构的属性基加密方案，突破传统策略仅支持正向逻辑的限制，可以更灵活地描述复杂的访问控制需求。随后，Waters等[23]开创性地将线性秘密共享方案融入ABE机制中，该方案借助矩阵实现了策略表达，为后续研究奠定了理论基础。

1.2　可追踪ABE

可追踪ABE方案通过密码学手段实现解密密钥溯源与恶意用户识别。根据追踪场景的差异，可追踪ABE方案分为白盒追踪与黑盒追踪[6-7]。

Liu等[6]提出了首个白盒可追踪的CP-ABE方案，为后续研究奠定了理论框架。文献[7]所提方案通过构建白盒追踪表来实现用户身份绑定，其具体做法如下。①系统初始化阶段：初始化并公布追踪表；②密钥生成阶段：为数据用户分配密钥组件c，并将c与用户身份标识GID的对应关系添加到追踪表；③追踪阶段：先进行密钥完整性检查，再根据密钥组件c到追踪表中查找恶意用户GID。虽然文献[6]所提方案实现了白盒可追踪，但其追踪表的存储开销与系统用户规模成正比。针对此缺陷，Ning等[8]借助Shamir门限方案来实现追踪，提出一种支持大属性空间的白盒可追踪CP-ABE方案，消除了追踪表的存储开销。更进一步地，Meng等[24-25]提出服务器辅助的可撤销白盒可追踪CP-ABE方案。

Liu等[7]提出了首个支持黑盒可追踪的CP-ABE方案，创造性地通过构造追踪密文来实现黑盒追踪。虽然该方案存在密文大小与用户规模成正比的不足，但该方案采用的“普通密文+追踪密文”的设置被后来的研究者所沿用。针对文献[7]所提方案的不足，Liu等[26]构造了一个黑盒可追踪CP-ABE方案以进一步降低密文存储开销。Ning等[27]构造了一种更短密文的黑盒可追踪CP-ABE方案，该方案的密文大小仅与参与加密的属性个数成正比。Qiao等[9]提出了一种可扩展的黑盒可追踪CP-ABE方案，该方案在实现普通密文和追踪密文不可区分的同时，其追踪效率与用户规模无关。为避免恶意用户无法通过丢弃部分密钥来逃避追踪，Fan等[28]将身份ID嵌入ABE方案，提出全新黑盒可追踪方案。

近年来，可追踪ABE方案的研究方向主要包括追踪方案的功能性和追踪机制的完备性。一方面，研究者将多授权机构、大属性空间等特性与可追踪机制相结合，设计了支持丰富功能的可追踪ABE方案[10-11,14,29]。另一方面，研究者尝试设计追踪机制更完备的ABE方案。基于Qiao等[9]提出的黑盒可追踪方案和Liu等[6]提出的白盒可追踪方案，He等[12]提出的方案可以实现白盒/黑盒可追踪机制的兼容。Nasiraee等[15]提出多授权机构的白盒/黑盒可追踪CP-ABE方案，该方案采用投票算法实现去中心化的系统架构，借助哈希函数实现匿名的白盒追踪，通过构造追踪密文实现了黑盒追踪。

表1为上述方案关于白盒可追踪、黑盒可追踪、多授权机构和大属性空间等方面的功能对比，其中√表示支持/具备该功能，×表示不支持/不具备该功能。

表1　　可追踪ABE方案的功能对比

方案	白盒可追踪	黑盒可追踪	多授权机构	大属性空间
文献[6,24-25]	√	×	×	×
文献[7,26-28]	×	√	×	×
文献[8]	√	×	×	√
文献[9]	×	√	×	√
文献[10-11]	√	×	√	√
文献[12-13]	√	√	×	√
文献[14,29]	√	×	√	×
文献[15]	√	√	√	×

2　预备知识

本节将介绍一些基础知识，包括双线性映射、本文涉及的困难问题假设以及线性秘密共享方案。

2.1　双线性映射

双线性映射又叫双线性配对，一般记作e:G ×

$G \to G_T$，其中 G 和 G_T 皆为素数阶循环群。其具备的主要性质如下。

1) 双线性：对于任意的元素 $x,y \in G$ 和任意的 $a,b \in Z_p$，都有 $e\left(x^a,y^b\right)=e\left(x,y\right)^{ab}$。

2) 对称性：对于任意的元素 $x,y \in G$，有 $e\left(x,y\right)=e\left(y,x\right)$。

3) 可计算：对于任意的元素 $x,y \in G$，$e\left(x,y\right)$ 是可被计算的。

2.2 困难问题假设

根据文献[30]，给出 l-SDH 假设的定义：给定一个元组 $(g,g^a,g^{a^2},\cdots,g^{a^l})$，其中 a 是从整数群 Z_p 上产生的随机元素，l-SDH 问题是输出一对元素 $\left(c,g^{\frac{1}{a+c}}\right)$。假设任意一个概率多项式时间（PPT, probabilistic polynomial-time）算法 $\mathcal{A}$ 解决 l-SDH 问题的优势为 ϵ，即 $|\Pr[\mathcal{A}(g,g^a,g^{a^2},\cdots,g^{a^l})=(c,g^{\frac{1}{a+c}})]|\geqslant\epsilon$。如果 ϵ 是可忽略的，那么意味着任意 PPT 算法仅能以可忽略优势解决 l-SDH 问题，则称 l-SDH 假设成立。

根据文献[23]，这里给出 L-DPBDHE 假设的定义。执行群生成算法 $\mathcal{G}(\cdot)$，得到素数阶循环群的描述信息 $\mathrm{GD}=\left(G,G_T,p,e\right)$，随机挑选 $L+2$ 个随机整数 $s,a,b_1,b_2,\cdots,b_L \in Z_p$。敌手获得群描述信息 GD 和元组 Tu。

$$\mathrm{Tu}=\left\{g,g^s,\left\{g^{a^j},g^{b_k},g^{sb_k},g^{a^jb_k},g^{\frac{a^j}{b_k^2}}\right\}_{\forall(j,k)\in[L,L]},\left\{g^{a^j\frac{b_k}{b_l^2}}\right\}_{\forall(j,k,l)\in[2L,L,L],k\neq l},\left\{g^{\frac{a^j}{b_k}}\right\}_{\forall(j,k)\in[2L,L],j\neq L+1},\left\{g^{s\frac{b_k}{b_l}},g^{sa^j\frac{b_k}{b_l^2}}\right\}_{\forall(j,k,l)\in[L,L,L],k\neq l}\right\}$$

L-DPBDHE 问题是区分 $e\left(g,g\right)^{sa^{L+1}}$ 和随机元素 R。PPT 算法 $\mathcal{A}$ 解决 L-DPBDHE 问题的优势为 ϵ，即 $|\Pr[\mathcal{A}(\mathrm{Tu},e\left(g,g\right)^{sa^{L+1}})=0]-\Pr[\mathcal{A}(e\left(g,g\right)^{sa^{L+1}},R)=0]|\geqslant\epsilon$。如果 ϵ 是可忽略的，则说明任意 PPT 算法仅能以可忽略的优势解决 L-DPBDHE 问题，则称 L-DPBDHE 假设成立。

2.3 线性秘密共享方案

线性秘密共享方案（LSSS, linear secret sharing scheme）是一种借助矩阵实现秘密分发和访问策略表达的密码学技术，其思想是将秘密值分割成多个份额，并将这些份额分配给多个参与者。只有满足访问控制条件的参与者集合才能恢复原始秘密值，而不满足条件的集合则无法获得任何关于秘密的信息。线性秘密共享方案由秘密分享和秘密恢复 2 个部分组成。

1) 秘密分享：假设 $\boldsymbol{M}$ 为一个 l 行 n 列的矩阵，存在一个映射函数 ρ 将矩阵 $\boldsymbol{M}$ 的某一行映射到属性集合 S 中的某个属性。秘密分享时，对于用于分享的秘密信息 s，构建一个向量 $\boldsymbol{v}=\left(s,v_2,\cdots,v_n\right)$，其中 $\left\{v_i\right\}_{i\in[2,n]}$ 是随机元素，则属性 $\rho(i)$ 对应的分享值 $\lambda_i=\boldsymbol{M}_i\boldsymbol{v}$。

2) 秘密恢复：对于任何授权集合 $L \in \left(M,\rho\right)$ 和 $I=\left\{i:i\in[l]\wedge\rho(i)\in L\right\}$，存在定值集合 $\left\{w_i\right\}_{i\in I}$ 满足 $\sum_{i\in I}w_i\boldsymbol{M}_i=(1,0,\cdots,0)$，则秘密值 $s=\sum_{i\in I}w_i\lambda_i$。

3 系统定义

本节首先介绍本文方案的系统模型，其次介绍本文方案的框架，最后介绍本文方案的安全模型。

3.1 系统模型

本文方案包括 7 个参与方，各参与方职责如下。

1) 可信机构（TA, trusted authority）：TA 负责初始化系统并公布公共参数。同时，TA 协助属性授权机构（AA, attribute authority）完成初始化。

2) 属性授权机构：完成初始化之后，AA 负责为合法用户颁发解密密钥。

3) 云服务器（CS, cloud server）：CS 负责存储和分享密文。

4) 数据拥有者（DO, data owner）：DO 是机密数据的所有者和贡献者，负责基于访问策略加密数据，并将密文上传至 CS。本文方案假设 DO 是诚实且可信的。

5) 数据用户（DU, data user）：DU 是机密数据的使用者。本文方案中，DU 存在故意泄露自己的解密权限（解密密钥或解密黑盒）以牟利的可能。

6) 未授权用户（UAU, un-authorized user）：UAU 从授权用户处得到解密密钥或解密黑盒，非

法访问机密数据。

7) 追踪者（Tracer）：当检测到密钥滥用事件时，追踪者执行追踪来定位恶意用户的身份信息。

图 1 为本文方案的系统模型。首先，TA 完成初始化并公布公共参数，继而各 AA 从 TA 处获得主密钥并完成自身的初始化；其次，DU 从 AA 处获取密钥；接着，DO 指定机密数据的访问策略，加密并将密文交付给 CS 进行分享；然后，DU 下载并解密密文，获得明文数据；最后，当检测到密钥泄露时，追踪者执行白盒追踪或黑盒追踪，定位恶意用户的身份。

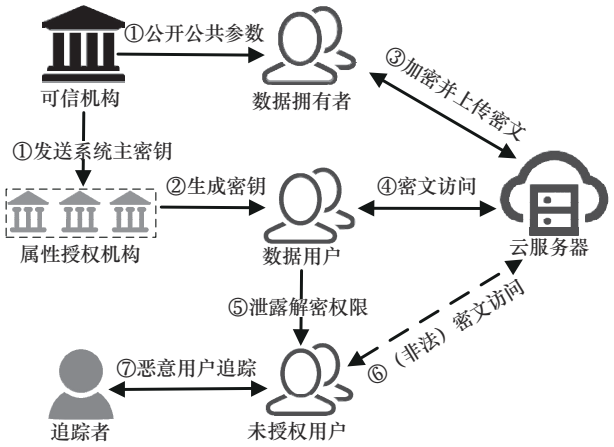


图1　本文方案的系统模型

3.2　方案框架

本文方案包含如下 8 个算法。

1) $\mathrm{Setup}_{\mathrm{TA}}(1^{\lambda}) \rightarrow \{\mathrm{PP},\mathrm{MSK}\}$：该算法由 TA 执行，以安全参数 1^{λ} 为输入，输出系统的公共参数 PP 和主密钥 MSK。此外，该算法还初始化白盒追踪表 T_{W} 和黑盒追踪表 T_{B}。

2) $\mathrm{Setup}_{\mathrm{AA}_j}(\mathrm{PP},\mathrm{MSK}) \rightarrow \left\{\mathrm{APK}_j,\mathrm{ASK}_j\right\}$：该算法由 AA 执行，以 PP 和 MSK 作为输入，输出属性授权机构的密钥对 $\left\{\mathrm{APK}_j,\mathrm{ASK}_j\right\}$。

3) $\mathrm{KeyGen}(\mathrm{S}_{\mathrm{GID}},\mathrm{GID},\mathrm{PP},\left\{\mathrm{ASK}_j\right\}) \rightarrow \mathrm{SK}_{\mathrm{GID}}$：该算法由负责用户属性集 S_{GID} 中属性的 AA 执行，以用户属性集 S_{GID}、用户身份标识 GID、公共参数 PP 和属性机构私钥 ASK_j 为输入，为该用户生成密钥 $\mathrm{SK}_{\mathrm{GID}}$。该算法还将添加 GID 的追踪信息到追踪表 T_{W} 和 T_{B}。

4) $\mathrm{Encrypt}\left(m,\Psi,\mathrm{PP},\left\{\mathrm{APK}_j\right\}\right) \rightarrow \mathrm{CT}$：该算法由 DO 执行数据加密算法，以明文 m、访问策略 Ψ、PP 以及属性机构公钥 $\left\{\mathrm{APK}_j\right\}$ 为输入，该算法基于 Ψ 加密 m 得到密文 CT。

5) $\mathrm{Decrypt}(\mathrm{CT},\mathrm{SK}_{\mathrm{GID}}) \rightarrow m$ 或 $\perp$：该算法由 DU 执行，以密文 CT 和密钥 $\mathrm{SK}_{\mathrm{GID}}$ 为输入，当且仅当 $\mathrm{SK}_{\mathrm{GID}}$ 绑定的用户属性集满足 CT 绑定的策略时，该算法输出明文 m，否则输出 $\perp$ 表示失败。

6) $\mathrm{WTrace}(\mathrm{PP},\mathrm{SK}) \rightarrow$ GID 或 $\perp$：该算法由 Tracer 执行，以 PP 和被检测到的泄露密钥 SK 作为输入。首先，该算法对密钥 SK 进行密钥完整性检查。如果通过检查，该算法输出泄露密钥用户 GID，否则输出终止符号 $\perp$ 表示追踪失败。

7)$\mathrm{BKEncrypt}\left(\Psi,\mathrm{PP},\left\{\mathrm{APK}_j\right\}\right) \rightarrow \{\mathrm{TCT},\zeta\}$：该算法由 Tracer 执行，以访问策略 Ψ、公共参数 PP 和属性机构公钥 $\left\{\mathrm{APK}_j\right\}$ 为输入，输出追踪密文 TCT 和追踪密文参数 ζ。

8)$\mathrm{BTrace}(\mathrm{PP},\mathrm{TCT},\zeta) \rightarrow$ GID 或 $\perp$：该算法在 Tracer 和解密 BB 之间交互执行，以 PP、TCT 和 ζ 作为输入，通过分析解密黑盒的输出，最终输出恶意用户的身份标识 GID。

上述各算法在方案中的执行流程以及数据交互情况如图 2 和图 3 所示。

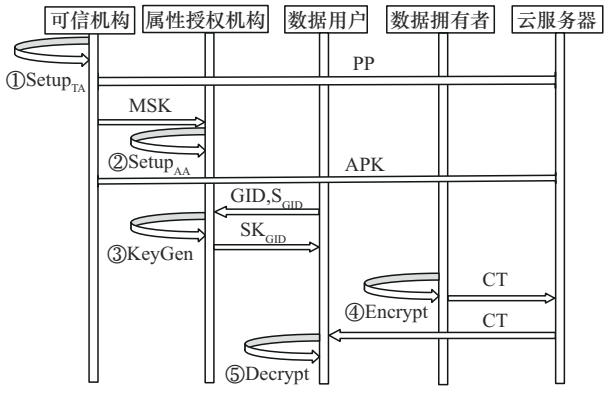


图2　方案框架（初始化、密钥生成、加密以及解密阶段）

图 2 展示了加密系统初始化、密钥生成、加密以及解密阶段的交互情况。图 3 为执行恶意用户追踪时，Tracer 与 UAU 之间的交互。当 UAU 使用解密 WB 非法访问机密数据时，Tracer 从 WB 处获得密钥并执行白盒追踪（WTrace）算法定位恶意用户身份；当 UAU 使用解密 BB 非法访问数据时，Tracer 执行黑盒追踪（BTrace）机制来识别恶意用户。

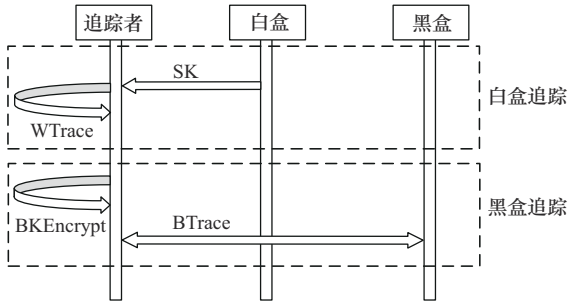


图3 方案框架(追踪阶段)

另外，TA 执行 Setup_{TA} 算法和 AA 执行 $\text{Setup}_{\text{AA}_j}$ 来完成系统的初始化，2 个初始化过程均未涉及对于属性的处理，而动态属性支持和属性相关计算是执行加密算法 Encrypt 和密钥生成算法 KeyGen 过程中完成的，因此本文方案实现了对大属性空间特性的支持。

3.3 安全模型

本节给出本文方案的安全模型，以及需满足的安全要求，包括选择明文攻击下的密文不可区分性(IND-CPA)、白盒可追踪性和黑盒可追踪性。

1)选择明文攻击下的密文不可区分性由挑战者 $\mathcal{C}$ 和一个 PPT 敌手 $\mathcal{A}$ 之间的密文不可区分游戏定义。

系统初始化：敌手 $\mathcal{A}$ 发送给挑战者 $\mathcal{C}$ 一个挑战访问策略 Ψ^*。此后，$\mathcal{A}$ 用于查询密钥的属性集合均不能满足 Ψ^*。

设置：$\mathcal{C}$ 执行算法 Setup_{TA}，生成系统的公共参数 PP 和主密钥 MSK。随后，$\mathcal{C}$ 执行 Setup_{AA} 算法为属性授权机构 AA_j 生成密钥对 $\{\text{APK}_j,\text{ASK}_j\}$。$\mathcal{C}$ 将 PP 和 APK_j 发送给 $\mathcal{A}$。

阶段 1：$\mathcal{A}$ 适应性地挑选一系列属性集合 S^* 和身份标识 GID^* 并发送给 $\mathcal{C}$，$\mathcal{C}$ 回复 $\mathcal{A}$ 对应的 SK^*。

挑战：$\mathcal{A}$ 随机挑选明文消息 m_0 和 m_1 并发送给挑战者 $\mathcal{C}$，其中 $|m_0|=|m_1|$。$\mathcal{C}$ 首先随机选择其中一条消息 m_v，然后 $\mathcal{C}$ 执行算法 Encrypt 加密 m_v 得到密文 CT_{m_v}。最终，CT_{m_v} 被返回给 $\mathcal{A}$。

阶段 2：与阶段 1 相似，$\mathcal{A}$ 适应性地挑选一系列属性集合 S^* 和身份标识 GID^* 并发送给 $\mathcal{C}$，$\mathcal{C}$ 回复 $\mathcal{A}$ 对应的密钥 SK^*。

猜测：$\mathcal{A}$ 输出对于 v 猜测，记为 v'。若 $v=v'$，则 $\mathcal{A}$ 获得游戏的胜利。定义 $\mathcal{A}$ 赢得游戏的优势为 $\text{Adv}_{\mathcal{A}}=\left|\Pr[v=v']-\frac{1}{2}\right|$。

定义 1 一个 CP-ABE 方案是选择明文攻击下密文不可区分的，如果任意 PPT 敌手仅能以可忽略优势赢得上述游戏。

2)白盒可追踪性由挑战者 $\mathcal{C}$ 和一个 PPT 敌手 $\mathcal{A}$ 之间的白盒可追踪游戏定义。

系统初始化：首先，$\mathcal{C}$ 执行 Setup_{TA} 算法，生成 PP 和 MSK。接着，$\mathcal{C}$ 执行 Setup_{AA} 算法为 AA_j 生成密钥 $\{\text{APK}_j,\text{ASK}_j\}$。最后，$\mathcal{C}$ 将 PP 和 APK_j 发送给 $\mathcal{A}$。

查询阶段：敌手 $\mathcal{A}$ 适应性地发起关于 $(\text{GID}_1,S_{\text{GID}_1}),(\text{GID}_2,S_{\text{GID}_2}),\cdots,(\text{GID}_q,S_{\text{GID}_q})$ 的解密密钥请求，其中 q 为查询解密密钥的次数上限。

解密密钥伪造阶段：$\mathcal{A}$ 提交给 $\mathcal{C}$ 其伪造的解密密钥 SK^*，挑战者 $\mathcal{C}$ 执行 WTrace 算法，若满足 $\text{WTrace}(\text{PP},\text{SK}^*)\notin\{\text{GID}_1,\text{GID}_2,\cdots,\text{GID}_q\}$ 且 $\text{WTrace}(\text{PP},\text{SK}^*)\neq\perp$，则 $\mathcal{A}$ 赢得上述游戏。

定义挑战者在上述游戏中的优势为 $\Pr[\text{WTrace}(\text{PP},\text{SK}^*)\notin\{\perp,\text{GID}_1,\text{GID}_2,\text{L},\text{GID}_q\}]$。

定义 2 一个 CP-ABE 方案是白盒可追踪的，若任意 PPT 敌手仅能以可忽略优势赢得上述游戏。

3)黑盒可追踪性由挑战者 $\mathcal{C}$ 和一个 PPT 敌手 $\mathcal{A}$ 之间的盒可追踪游戏定义。

系统初始化：$\mathcal{A}$ 发送挑战访问策略 Ψ^* 给挑战者 $\mathcal{C}$。此后，用于查询密钥的属性集均不满足 Ψ^*。

设置：$\mathcal{C}$ 执行 Setup_{TA} 算法，生成加密系统的公共参数 PP 和主密钥 MSK。$\mathcal{C}$ 执行 Setup_{AA} 算法为属性授权机构生成密钥对 $\{\text{APK}_j,\text{ASK}_j\}$。$\mathcal{C}$ 将 PP 和 APK_j 发送给敌手 $\mathcal{A}$。

阶段 1：$\mathcal{A}$ 适应性地挑选属性集合 S^* 和身份标识 GID^* 并发送给 $\mathcal{C}$，$\mathcal{C}$ 回复 $\mathcal{A}$ 对应的密钥 SK^*。

挑战：$\mathcal{A}$ 随机挑选明文 m 并发送给 $\mathcal{C}$。$\mathcal{C}$ 随机挑选 $v\in\{0,1\}$，若 $v=0$，执行加密算法 Encrypt 加密 m，将输出的密文 CT 发送给敌手，否则执行算法 BKEncrypt，将追踪密文 TCT 发送给 $\mathcal{A}$。

阶段 2：与阶段 1 相似，$\mathcal{A}$ 适应性地向 $\mathcal{C}$ 发送属性集合并查询对应的 SK。

猜测：敌手 $\mathcal{A}$ 输出对于 v 猜测 v'。若 $v=v'$，则获得游戏的胜利。定义敌手 $\mathcal{A}$ 获胜的优势为

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

定义 3　一个 CP-ABE 方案是黑盒可追踪的，若任意 PPT 敌手仅能以可忽略优势赢得上述游戏。

4　具体构造

本节将介绍 3.2 节中定义的算法细节。

1)$\mathrm{Setup}_{\mathrm{TA}}(1^{\lambda})\rightarrow\{\mathrm{PP},\mathrm{MSK}\}$：该算法以安全参数 1^{λ} 为输入，输出公共参数 PP 和主密钥 MSK。

首先，该算法运行群生成算法 $\mathcal{G}\left(1^{\lambda}\right)$ 得到素数循环群 G、G_T 和双线性映射 $e{:}G\times G\rightarrow G_T$，其中群 G、G_T 的生成元为 g，阶为 p。

其次，该算法随机选择 3 个元素 $h\in G$ 和 $\alpha,\beta\in Z_p$，并计算 $e(g,h)^{\alpha},g^{\alpha},g^{\beta},h^{\beta}$。

接着，该算法选择 3 个抗碰撞的哈希函数 $H_0{:}\{0,1\}^{*}\rightarrow Z_p,H_1{:}Z_p\rightarrow G,H{:}G_T\rightarrow\{0,1\}^{\lambda}$，并定义映射函数 $f_1{:}U_a\rightarrow U_A,f_2{:}[l]\rightarrow U_A$，其中 f_1 将属性空间 U_a 映射到属性机构空间 U_A，f_2 将访问策略矩阵的一行映射到属性机构空间 U_A。

最后，该算法输出加密系统的公共参数 $\mathrm{PP}=\{e,g,h,e(g,h)^{\alpha},g^{\alpha},g^{\beta},h^{\beta},f_1,f_2,H_0,H_1,H\}$ 和系统主密钥 $\mathrm{MSK}=\{\beta\}$。同时，该算法初始化白盒追踪表 T_{W} 和黑盒追踪表 T_{B}。

2)$\mathrm{Setup}_{\mathrm{AA}_j}(\mathrm{PP},\mathrm{MSK})\rightarrow\{\mathrm{APK}_j,\mathrm{ASK}_j\}$：该算法以公共参数 PP 和系统主密钥 MSK 作为输入，输出属性授权机构 AA_j 的公私钥对 $\{\mathrm{APK}_j,\mathrm{ASK}_j\}$。该算法随机选择元素 $\gamma_j\in Z_p$，并计算 AA_j 的公私钥对 $\mathrm{APK}_j=\{g^{\gamma_j}\},\mathrm{ASK}_j=\{\gamma_j,\beta\}$，其中 β 为主密钥。

3)$\mathrm{KeyGen}(S_{\mathrm{GID}},\mathrm{GID},\mathrm{PP},\{\mathrm{ASK}_j\})\rightarrow\mathrm{SK}_{\mathrm{GID}}$：该算法以用户属性集 S_{GID}、公共参数 PP、用户身份 GID 和属性授权机构私钥 ASK_j 为输入，S_{GID} 对应的属性授权机构协作为 GID 颁发解密密钥 $\mathrm{SK}_{\mathrm{GID}}$。

首先，为用户 GID 随机挑选 3 个元素 $u,\delta,c\in Z_p$，并随机选择一个字符串 $r_Q\in\{0,1\}^{*}$，借助哈希函数计算 $Q=H_1(\mathrm{GID}\|r_Q)$。

其次，计算密钥组件 $\mathrm{SK}_0=c,\mathrm{SK}_1=h^{u+(\beta+c)\delta}$，$\mathrm{SK}_4=g^{\frac{(\alpha+u)}{(\beta+c)}}\cdot g^{\delta}$。

接着，对于属性集合 S_{GID} 中的每一个属性 $\mathrm{att}_i\in S_{\mathrm{GID}}$，其对应的属性授权机构 $\mathrm{AA}_j(j=f_1(\mathrm{att}_i))$ 使用其私钥 ASK_j 计算密钥组件 $\mathrm{SK}_{i,2}$ 和 $\mathrm{SK}_{i,3}$，如式(1)所示。

$$\mathrm{SK}_{i,2}=Q^{H_0(\mathrm{att}_i)}h^{-(\beta+c)\delta\cdot\gamma_j},\mathrm{SK}_{i,3}=h^{\frac{\gamma_j u}{H_0(\mathrm{att}_i)}}Q \tag{1}$$

最后，输出用户 GID 的解密密钥 $\mathrm{SK}_{\mathrm{GID}}=\left\{S_{\mathrm{GID}},\mathrm{SK}_0,\mathrm{SK}_1,\{\mathrm{SK}_{i,2},\mathrm{SK}_{i,3}\}_{\mathrm{att}_i\in S_{\mathrm{GID}}},\mathrm{SK}_4\right\}$。

与此同时，该算法计算白盒追踪索引 $\mathrm{IW}=c$ 和黑盒追踪索引 $\mathrm{IB}=H\left(e(g,h)^{u+(\beta+c)\delta}\right)$，将(IW,GID)与(IB,GID)分别记录到追踪表 T_{W} 和 T_{B} 中。

4)$\mathrm{Encrypt}\left(m,\Psi,\mathrm{PP},\{\mathrm{APK}_j\}\right)\rightarrow\mathrm{CT}$：该算法由 DO 执行数据加密算法，以明文 m、访问策略 Ψ、公共参数 PP 和属性机构公钥 $\{\mathrm{APK}_j\}$ 为输入，基于策略 Ψ 加密 m，得到密文 CT。

首先，随机选择元素 $s\in Z_p$，并计算密文组件 $C_0=me(g,h)^{s\cdot\alpha},C_4=h^{s},C_5=h^{\beta s}$。

其次，构造随机向量 $\boldsymbol{v}=(s,v_2,v_3,\cdots,v_l)^{\mathrm{T}}$ 和 $\boldsymbol{w}=(0,w_2,w_3,\cdots,w_l)^{\mathrm{T}}$，对于矩阵 $\boldsymbol{M}$ 的每行 $\boldsymbol{M}_i$，计算 $\lambda_i=\boldsymbol{M}_i\boldsymbol{v}$ 和 $\omega_i=\boldsymbol{M}_i\boldsymbol{w}$。

接着，为矩阵 $\boldsymbol{M}$ 的第 i 行 $\boldsymbol{M}_i$ 随机选择元素 $r_i\in Z_p,t\in G$，并使用其对应的属性机构公钥 $g^{\gamma_{f_2(i)}}$ 计算密文组件 $C_{i,1},C_{i,2},C_{i,3}$，如式(2)所示。

$$C_{i,1}=\left(g^{\gamma_{f_2(i)}}\right)^{r_i}\cdot g^{\lambda_i},C_{i,2}=g^{r_i}t^{\omega_i},C_{i,3}=g^{H_0(\mathrm{att}_i)r_i} \tag{2}$$

最终，输出密文 CT，如式(3)所示。

$$\mathrm{CT}=\left\{C_0,\{C_{i,1},C_{i,2},C_{i,3}\}_{i\in[1,l]},C_4,C_5,\{M,\rho\}\right\} \tag{3}$$

5)$\mathrm{Decrypt}(\mathrm{CT},\mathrm{SK}_{\mathrm{GID}})\rightarrow m$ 或 $\perp$：该算法以密文 CT 和解密密钥 $\mathrm{SK}_{\mathrm{GID}}$ 作为输入，输出明文 m 或输出终止符 $\perp$ 表示解密失败。

首先，检查密钥 $\mathrm{SK}_{\mathrm{GID}}$ 的属性集是否满足 CT 绑定的访问控制策略。若不满足，算法输出 $\perp$ 表示解密失败，否则计算常数集合 $\{o_i\}_{i\in[1,l]}$，使 $\sum_{i\in[1,l]}o_i\boldsymbol{M}_i=[1,0,\cdots,0]$ 成立。

其次，分别计算 R_0 和 R_1，如式(4)和式(5)所示。

$$R_0=\prod_{i=1}^{l}\left(\frac{e\left(C_{i,1},\mathrm{SK}_1\right)\cdot e\left(C_{i,2},\mathrm{SK}_{i,2}\right)}{e\left(C_{i,3},\mathrm{SK}_{i,3}\right)}\right)^{o_i}=\prod_{i=1}^{l}\left(e\left(g,h\right)^{u\lambda_i}\cdot e\left(h,g\right)^{(\beta+c)\delta\lambda_i}\right)^{o_i}=e\left(g,h\right)^{u\sum_{i=1}^{l}\lambda_i o_i}\cdot e\left(h,g\right)^{(\beta+c)\delta\sum_{i=1}^{l}\lambda_i o_i}=e\left(g,h\right)^{s\cdot\left(u+(\beta+c)\delta\right)} \tag{4}$$

$$R_1=e\left(C_5C_4^{\mathrm{SK}_0},\mathrm{SK}_4\right)=e\left(h^{\beta s}h^{sc},g^{\frac{\alpha+u}{\beta+c}}\cdot g^{\delta}\right)=e\left(g,h\right)^{s\left(u+\alpha+(\beta+c)\delta\right)} \tag{5}$$

最后，通过式(6)计算得到数据明文m。

$$m=\frac{C_0\cdot R_0}{R_1}=\frac{m\cdot e\left(g,h\right)^{s\cdot\alpha}\cdot e\left(g,h\right)^{s\cdot\left(u+(\beta+c)\delta\right)}}{e\left(g,h\right)^{s\left(u+\alpha+(\beta+c)\delta\right)}} \tag{6}$$

6)WTrace(PP,SK)→GID或⊥：算法首先对密钥$\mathrm{SK}=\left\{\mathrm{S}_{\mathrm{GID}},\mathrm{SK}_0,\mathrm{SK}_1,\left\{\mathrm{SK}_{i,2},\mathrm{SK}_{i,3}\right\}_{\mathrm{att}_i\in\mathrm{S}_{\mathrm{GID}}},\mathrm{SK}_4\right\}$进行完整性检查，具体包括如下3项。

①$\mathrm{SK}_0\in Z_p,\mathrm{SK}_1,\mathrm{SK}_{i,2},\mathrm{SK}_{i,3},\mathrm{SK}_4\in G$。

②等式$e\left(h^{\beta}h^{\mathrm{SK}_0},\mathrm{SK}_4\right)=e\left(g,h\right)^{\alpha}\cdot e\left(\mathrm{SK}_1,g\right)$成立。

③$\exists\mathrm{att}(i)\in S$，使式(7)成立。

$$e\left(\frac{\left(\mathrm{SK}_{i,3}\right)^{H_0\left(\mathrm{att}_i\right)}}{\mathrm{SK}_{i,2}},g\right)=e\left(\frac{\left(h^{\frac{\gamma_j\cdot u}{H_0\left(\mathrm{att}_i\right)}}\cdot Q\right)^{H_0\left(\mathrm{att}_i\right)}}{Q^{H_0\left(\mathrm{att}_i\right)}\cdot h^{-(\beta+c)\delta\cdot\gamma_j}},g\right)=e\left(\frac{h^{\gamma_j\cdot u}\cdot Q^{H_0\left(\mathrm{att}_i\right)}}{Q^{H_0\left(\mathrm{att}_i\right)}\cdot h^{-(\beta+c)\delta\cdot\gamma_j}},g\right)=e\left(h^{\gamma_j\cdot\left((\beta+c)\delta+u\right)},g\right)=e\left(h^{(\beta+c)\delta+u},g^{\gamma_j}\right)=e\left(\mathrm{SK}_1,g^{\gamma_{f_1(\mathrm{att}_i)}}\right) \tag{7}$$

如果未通过上述3条检查条件，则算法输出⊥表示白盒追踪失败。否则认为SK通过了密钥完整性检查，该算法再根据白盒追踪索引$\mathrm{IW}=\mathrm{SK}_0$到白盒追踪表$T_{\mathrm{W}}$中查找。若查找成功，则输出对应的全局身份标识GID，否则输出⊥表示失败。

7)BKEncrypt$\left(\Psi,\mathrm{PP},\left\{\mathrm{APK}_j\right\}\right)\rightarrow\{\mathrm{TCT},\zeta\}$：该算法以访问策略$\Psi$、公共参数PP和属性机构公钥$\left\{\mathrm{APK}_j\right\}$为输入，输出追踪密文TCT和追踪密文参数$\zeta$，解密黑盒关联的属性集需满足策略$\Psi$。TCT和普通密文CT在形式上是一致的，其加密过程相似。唯一的不同是BKEncrypt算法随机挑选元素$\hat{s}$、$\bar{s}$并将其嵌入追踪密文，而加密算法则只使用了一个秘密值s。BKEncrypt算法的加密过程如下。

首先，随机选择元素$\hat{s},\bar{s}\in Z_p$和$m\in G_T$，并计算$\overline{C}_0=m\cdot e\left(g,h\right)^{\bar{s}\cdot\alpha}$和$\overline{C}_4=h^{\bar{s}},\overline{C}_5=h^{\beta\cdot\bar{s}}$。

其次，构造2个随机向量$\hat{\boldsymbol{v}}=\left(\hat{s},\hat{v}_2,\hat{v}_3,\cdots,\hat{v}_l\right)$和$\hat{\boldsymbol{w}}=\left(0,\hat{w}_2,\hat{w}_3,\cdots,\hat{w}_l\right)$，对于矩阵$\boldsymbol{M}$的第$i$行$M_i$，计算$\hat{\lambda}_i=M_i\hat{\boldsymbol{v}}$和$\hat{\omega}_i=M_i\hat{\boldsymbol{w}}$。

接着，为访问策略矩阵$\boldsymbol{M}$的第i行M_i随机选择整数元素$\hat{r}_{\mathrm{i}}\in Z_{\mathrm{p}}$，并分别计算追踪密文组件$\hat{C}_{\mathrm{i},1}=g^{\gamma_{f_2(i)}\cdot\hat{r}_i+\hat{\lambda}_i},\hat{C}_{i,2}=g^{\hat{r}_i}t^{\hat{\omega}_i},\hat{C}_{i,3}=g^{H_0(\rho(i))\hat{r}_i}$。

最后，输出密文参数$\zeta=\{\hat{s},\bar{s},m\}$和追踪密文$\mathrm{TCT}=\left\{\overline{C}_0,\left\{\hat{C}_{i,1},\hat{C}_{i,2},\hat{C}_{i,3}\right\}_{i\in[l]},\overline{C}_4,\overline{C}_5,\{M,\rho\}\right\}$。注意TCT和CT在形式上是一致的，解密黑盒无法区分追踪密文和普通密文，只有满足这一特性黑盒追踪才能正常执行，否则解密黑盒可以返回随机结果以规避黑盒追踪。

8)BTrace(PP,TCT,ζ)→GID或⊥：该算法以公共参数PP、追踪密文TCT和追踪密文参数ζ为输入，追踪成功则输出恶意用户GID，否则输出⊥表示失败。该算法涉及追踪者和解密黑盒的交互。

首先，追踪者将追踪密文TCT输入解密黑盒中，解密黑盒执行解密算法Decrypt(TCT,SK)并输出解密结果m'。具体地，解密黑盒内部计算$\widehat{R_0},\overline{R}_1$如式(8)和式(9)所示，并进一步计算解密结果$m'$如式(10)所示。

$$\widehat{R_0}=\prod_{i=1}^{l}\left(\frac{e\left(\hat{C}_{i,1},\mathrm{SK}_1\right)\cdot e\left(\hat{C}_{i,2},\mathrm{SK}_{i,2}\right)}{e\left(\hat{C}_{i,3},\mathrm{SK}_{i,3}\right)}\right)^{o_i}=\prod_{i=1}^{l}\left(e\left(g,h\right)^{u\hat{\lambda}_i}\cdot e\left(h,g\right)^{(\beta+c)\delta\hat{\lambda}_i}\right)^{o_i}=e\left(g,h\right)^{u\sum_{i=1}^{l}\hat{\lambda}_i o_i}\cdot e\left(h,g\right)^{(\beta+c)\delta\sum_{i=1}^{l}\hat{\lambda}_i o_i}=e\left(g,h\right)^{\hat{s}\cdot\left(u+(\beta+c)\delta\right)} \tag{8}$$

$$\bar{R}_1 = e\left(\bar{C}_5 \bar{C}_4^{\mathrm{SK}_0}, \mathrm{SK}_4\right) = e\left(h^{\beta\bar{s}} h^{\bar{s}c}, g^{\frac{\alpha+u}{\beta+c}} \cdot g^{\delta}\right) = e(g,h)^{\bar{s}\left(u+\alpha+(\beta+c)\delta\right)} \tag{9}$$

$$m' = \frac{\bar{C}_0 \cdot \widehat{R_0}}{\bar{R}_1} = m \cdot e(g,h)^{(\hat{s}-\bar{s})\cdot\left(u+(\beta+c)\delta\right)} \tag{10}$$

注意，$\widehat{R_0}$、$\bar{R}_1$、m'的计算过程与式(4)~式(6)相似，但由于追踪密文TCT中嵌入了2个不同的元素$\hat{s}$、$\bar{s}$且$\hat{s}-\bar{s}$不为0，故m'不是明文数据，而是包含了用户身份信息的元素，解密黑盒对此一无所知。

然后，算法计算黑盒追踪索引IB，如式(11)所示，并根据IB查询黑盒追踪表T_{B}。如果T_{B}中存在IB，则输出对应GID，否则输出⊥表示失败。

$$\mathrm{IB} = H\left(\left(\frac{m'}{m}\right)^{\frac{1}{(\hat{s}-\bar{s})}}\right) = H\left(e(g,h)^{u+(\beta+c)\delta}\right) \tag{11}$$

5　安全性分析

本节证明本文方案是选择明文攻击下的密文不可区分的、白盒可追踪的和黑盒可追踪的。

5.1　选择明文攻击的密文不可区分性证明

定理1　若所有PPT敌手均不能以不可忽略优势赢得3.3节定义的密文不可区分游戏，则本文方案具备选择明文攻击下密文不可区分性。

证明　挑战者$\mathcal{C}$选择双线性群描述信息$\mathrm{GD} = \{G, G_T, e, p, g\}$。$\mathcal{C}$构造L-DPBDHE元组Tu，如式(12)所示。敌手$\mathcal{A}$发送挑战访问策略$\Psi^* = \{\boldsymbol{M}^*, \rho^*\}$给$\mathcal{C}$。此后，$\mathcal{A}$向$\mathcal{C}$查询解密密钥所用的属性集合均不满足访问策略$\Psi^*$。

$$\mathrm{Tu} = \left\{ g, g^s, \left\{ g^{a^j}, g^{b_k}, g^{sb_k}, g^{a^j b_k}, g^{\frac{a^j}{b_k^2}} \right\}_{\forall(j,k)\in[L,L]}, \left\{ g^{a^j \frac{b_k}{b_l^2}} \right\}_{\forall(j,k,l)\in[2L,L,L],k\neq l}, \left\{ g^{\frac{a^j}{b_k}} \right\}_{\forall(j,k)\in[2L,L],j\neq L+1}, \left\{ g^{s\frac{b_k}{b_l}}, g^{sa^j\frac{b_k}{b_l^2}} \right\}_{\forall(j,k,l)\in[L,L,L],k\neq l} \right\} \tag{12}$$

设置：挑战者$\mathcal{C}$随机选择元素$\tilde{\alpha}, \tilde{\beta} \in Z_p$，并令$\alpha = \tilde{\alpha} + a, \beta = \tilde{\beta} + b, h = g^{a^L}$，其中$a$和$L$属于Tu。然后，挑战者$\mathcal{C}$计算加密系统的公共参数$\mathrm{PP} = \{e, g, e(g,h)^{\alpha} = e(g,g)^{(\tilde{\alpha}+a)\cdot a^L}, g^{\alpha} = g^{\tilde{\alpha}+a}, g^{\beta} = g^{\tilde{\beta}+b}, h = g^{a^L}, h^{\beta} = g^{a^L(\tilde{\beta}+b)}, f_2, H_0, H_1\}$和系统主密钥$\mathrm{MSK} = \{\beta = \tilde{\beta} + b\}$。接着，$\mathcal{C}$运行$\mathrm{Setup}_{\mathrm{AA}_j}$算法得到属性授权机构的公私钥对$\{\mathrm{APK}_j, \mathrm{ASK}_j\}$。最后，$\mathcal{C}$将PP和$\mathrm{APK}_j$发送给$\mathcal{A}$。

阶段1：$\mathcal{A}$适应性地挑选一系列属性集合S^*和身份标识GID^*并发送给$\mathcal{C}$，$\mathcal{C}$回复$\mathcal{A}$对应的SK^*。

挑战：敌手$\mathcal{A}$挑选2个等长明文数据m_0, m_1并发送给挑战者$\mathcal{C}$。首先，$\mathcal{C}$随机挑选一个比特$v \in \{0,1\}$，然后$\mathcal{C}$计算m_v密文CT_{m_v}，具体步骤如下。

1)计算数据密文CT的组件，包括$C_0 = m_v \cdot e(g,h)^{s\alpha} = m_v \cdot Z \cdot e(g,g)^{s\tilde{\alpha}\cdot a^L}, C_4 = h^s = g^{s\cdot a^L}, C_5 = g^{s\cdot a^L(\tilde{\beta}+b)}$。

2)随机选择整数元素$v_2, \cdots, v_n, w_2, \cdots, w_n$，并设置向量$\boldsymbol{v} = \left(s, sa + v_2, sa^2 + v_3, \cdots, sa^{n-1} + v_n\right)^{\mathrm{T}}$和向量$\boldsymbol{w} = \left(0, w_2, w_3, \cdots, w_{n-1}, w_n\right)^{\mathrm{T}}$，并计算$\lambda_x = \sum_{i\in[n]} M_{x,i}^* sa^{i-1} + \sum_{i\in[2,n]} M_{x,i}^* sv_i = \sum_{i\in[n]} M_{x,i}^* sa^{i-1} + \lambda_x^*, \omega_x = \sum_{i\in[n]} M_{x,i}^* w_i = \omega_x^*$。

3)访问策略中的每一属性$\rho(i)$，令$r_i = b_i, t = g^a, \gamma_{f_2(i)} = s \cdot \tilde{\gamma}_{f_2(i)}, H_0(\rho(i)) = \frac{a^i}{b_i}$，计算$C_{i,1} = g^{sb_i \cdot \tilde{\gamma}_{f_2(i)} + \lambda_i}, C_{i,2} = g^{b_i}(g^a)^{\omega_i}, C_{i,3} = g^{a^i}$。

最终，CT_{m_v}被发送给敌手$\mathcal{A}$。

阶段2：与阶段1相似，敌手$\mathcal{A}$多次适应性地向$\mathcal{C}$查询解密密钥SK。

猜测：敌手$\mathcal{A}$输出对于v的猜测v'。若$\mathcal{A}$输出0，则在$\mathcal{A}$的视角下$Z = e(g,g)^{s\cdot a^{L+1}}$，否则输出1表明$Z$是群$G_T$上的随机元素。有2种情况需要考虑。

若$Z = e(g,g)^{s\cdot a^{L+1}}$，从$\mathcal{A}$的视角看，挑战密文是一个合法的密文，敌手$\mathcal{A}$赢得上述游戏的概率为$\Pr\left[v = v' | Z = e(g^s, g)^{a^{L+1}}\right] = \frac{1}{2} + \kappa$。

若$Z = R_{G_T}$，从$\mathcal{A}$的视角看，挑战密文是随机生成的，不包含任何有关v的信息，$\mathcal{A}$赢得游戏的概率为$\Pr\left[v = v' | Z = R_{G_T}\right] = \frac{1}{2}$。

总体而言，$\mathcal{A}$赢得游戏的优势如式(13)所示。

$$\mathrm{Adv}_{\mathcal{A}}^{\mathrm{CPA}}=\left|\frac{1}{2}\left(\frac{1}{2}+\kappa\right)+\frac{1}{2}\cdot\frac{1}{2}-\frac{1}{2}\right|=\frac{\kappa}{2} \quad (13)$$

由于κ是不可忽略的，$\frac{\kappa}{2}$也是不可忽略的。式(13)意味着存在一个PPT敌手能以不可忽略优势解决L-DPBDHE问题，这与L-DPBDHE假设直接矛盾，本文方案的IND-CPA得证。证毕。

5.2 白盒可追踪性证明

定理2 若所有PPT敌手仅能以可忽略优势ϵ赢得白盒可追踪游戏，则方案是白盒可追踪的。

证明 如果存在PPT敌手$\mathcal{A}$在q次密钥查询之后能够以不可忽略优势ϵ赢得白盒可追踪游戏，则可以构造另一个PPT算法$\mathcal{B}$以不可忽略优势攻破l-SDH假设。给定素数阶循环群G、G_T，生成元g以及双线性映射$e:G\times G\to G_T$。$\mathcal{B}$收到l-SDH问题的一个实例$\left(p,G,G_T,e,g_1,{g_1}^{\beta},{g_1}^{\beta^2},\cdots,{g_1}^{\beta^l}\right)$，其中$g_1\in G,\beta\in Z_p$，算法$\mathcal{B}$的目标是输出一个元组满足$\left(c_i,D_i={g_1}^{\frac{1}{\beta+c_i}}\right)$。对每一个$i=0,1,2,\cdots,l$，设置$B_i={g_1}^{\beta^i}$。$\mathcal{B}$与敌手$\mathcal{A}$展开如下交互。

初始化阶段：$\mathcal{B}$随机选择q个不同的元素$c_1,c_2,\cdots,c_q\in Z_p$，并设置一个关于$y$的多项式$f(y)=\prod_{i=1}^{q}(y+c_i)=\sum_{i=0}^{q}\alpha_i y^i$，其中$\alpha_i$为多项式$f(y)$各项的系数。然后，$\mathcal{B}$设置$g=\prod_{i=0}^{q}(B_i)^{\alpha_i}={g_1}^{f(\beta)},g^{\beta}={g_1}^{f(\beta)\cdot\beta}$。接着，$\mathcal{B}$随机选择$\alpha\in Z_p,h\in G$，计算$e(g,h)^{\alpha}$和$g^{\alpha}$，得到公共参数$\mathrm{PP}=\{e,g,h,e(g,h)^{\alpha},g^{\alpha},g^{\beta},h^{\beta},f_1,H_0,H_1\}$和系统主密钥$\mathrm{MSK}=\{\beta\}$。与此同时，$\mathcal{B}$运行$\mathrm{Setup}_{\mathrm{AA}_j}$算法得到属性授权机构的公私钥对$\{\mathrm{APK}_j,\mathrm{ASK}_j\}$。最后，$\mathcal{B}$将公共参数PP和属性授权机构公钥$\{\mathrm{APK}_j\}$发送给$\mathcal{A}$。另外，$\mathcal{B}$初始化白盒追踪表$T_{\mathrm{W}}$。

查询阶段：敌手$\mathcal{A}$向$\mathcal{B}$发起$(\mathrm{GID},\mathrm{S}_{\mathrm{GID}})$的解密密钥查询。不失一般性地，考虑第$i$次查询，其中$i\leqslant q$。$\mathcal{B}$进行如下操作。

1)$\mathcal{B}$构造一个关于y的多项式$f_i(y)=\frac{f(y)}{y+c_i}=\prod_{j=1,j\neq i}^{q}(y+c_j)=\sum_{j\in[0,q]}\bar{\alpha}_j y^j$，计算$\sigma_i=g^{\frac{1}{\beta+c_i}}=\sum_{j\in[0,q-1]}(B_j)^{\bar{\alpha}_j}={g_1}^{f_i(\beta)}$。

2)$\mathcal{B}$计算密钥组件$\mathrm{SK}_0=c_i,\mathrm{SK}_1=h^{u+(\beta+c_i)\delta}$和$\mathrm{SK}_4=\sigma_i{}^{\alpha+u}g^{\delta}$。

3)对S_{GID}中的每一个属性$\mathrm{att}_\tau\in S_{\mathrm{GID}}$，$\mathcal{B}$计算$\mathrm{SK}_{\tau,2},\mathrm{SK}_{\tau,3}$。

4)模拟器$\mathcal{B}$将解密密钥$\mathrm{SK}_{\mathrm{GID}}=\left\{S_{\mathrm{GID}},\mathrm{SK}_0,\mathrm{SK}_1,\left\{\mathrm{SK}_{\tau,2},\mathrm{SK}_{\tau,3}\right\}_{\mathrm{att}_\tau\in S_{\mathrm{GID}}},\mathrm{SK}_4\right\}$发送给$\mathcal{A}$，将$(c_i,\mathrm{GID})$添加到追踪表$T_{\mathrm{W}}$。

密钥伪造阶段：$\mathcal{A}$伪造密钥SK'，并将SK'提交给$\mathcal{B}$。用$\epsilon_{\mathcal{A}}$表示事件“$\mathcal{A}$赢得上述交互式游戏”，这意味着SK'通过密钥完整性检查且算法$\mathrm{WTrace}(\mathrm{PP},\mathrm{SK}')$的输出不属于集合$\{\perp,\mathrm{GID}_1,\mathrm{GID}_2,\cdots,\mathrm{GID}_q\}$。此时，需要考虑2种情况。

若$\epsilon_{\mathcal{A}}$没有发生，即$\mathcal{A}$没有赢得交互式游戏。$\mathcal{B}$随机选择元组(c_s,D_s)作为l-SDH的一个解。

若事件$\epsilon_{\mathcal{A}}$发生了，即$\mathcal{A}$赢得交互式游戏。此时，$\mathcal{B}$设置一个关于y的多项式$f(y)=\varphi(y)(y+c')+\varphi-1,\varphi(y)=\sum_{0}^{q-1}\varphi_i y^i,\varphi-1\in Z_p$。与此同时，由于$f(y)=\prod_{i=1}^{q}(y+c_i),c_i\in Z_p$且$c'\notin\{c_1,c_2,\cdots,c_q\}$，则$y+c'$不能整除$f(y)$。$\mathcal{B}$计算$\frac{1}{\varphi-1}$，然后进行如下的计算$\sigma=\left(\frac{SK'_{\mathrm{GID},4}}{g^{\delta}}\right)^{\frac{1}{\alpha+u}}={g_1}^{\frac{f(\beta)}{\beta+c'}}={g_1}^{\varphi(\beta)}{g_1}^{\frac{\varphi-1}{\beta+c'}}$，$c_s=c',D_s={g_1}^{\frac{1}{\beta+c'}}=\left(\sigma\cdot\prod_{i=0}^{q-1}B_i^{-\varphi_i}\right)^{\frac{1}{\varphi-1}}$。此时，有$e({g_1}^{\beta}\cdot{g_1}^{c_s},D_s)=e(g_1,g_1)$，显然$(c_s,D_s)$是$l$-SDH的一个正确的解。

令$\epsilon(c_s,D_s)$表示事件“(c_s,D_s)是l-SDH的一个正确的解”。当(c_s,D_s)为$\mathcal{B}$随机生成的解时，事件$\epsilon(c_s,D_s)$发生的概率可忽略，记为0。当敌手$\mathcal{A}$在交互式游戏中获胜且$\gcd(\varphi-1,p)=1$的情况下，模拟器$\mathcal{B}$输出问题的解(c_s,D_s)必然满足$e({g_1}^{\beta}\cdot$

$g_1^{c_s},D_s)=e(g_1,g_1)$，事件$(c_s,D_s)$发生的概率记为1。综上所述，$\mathcal{B}$解决$l$-SDH问题的概率$\Pr[\epsilon(c_s,D_s)]$，如式(14)所示。$\mathcal{B}$能以不可忽略优势解决$l$-SDH问题，这与$l$-SDH假设矛盾，故所提方案是白盒可追踪的。证毕。

$$\begin{aligned}&\Pr[\epsilon(c_s,D_s)]=\Pr[\epsilon(c_s,D_s)|\overline{\mathcal{A}\,\text{win}}]\cdot\\&\Pr[\overline{\mathcal{A}\,\text{win}}]+\Pr[\epsilon(c_s,D_s)|\mathcal{A}\,\text{win}\wedge\gcd(\varphi-1,p)\neq1]\cdot\\&\Pr[\mathcal{A}\,\text{win}\wedge\gcd(\varphi-1,p)\neq1]+\\&\Pr[\epsilon(c_s,D_s)|\mathcal{A}\,\text{win}\wedge\gcd(\varphi-1,p)=1]\cdot\\&\Pr[\mathcal{A}\,\text{win}\wedge\gcd(\varphi-1,p)=1]=0+0+1\cdot\\&\Pr[\mathcal{A}\,\text{win}\wedge\gcd(\varphi-1,p)=1]=\\&\Pr[\mathcal{A}\,\text{win}\wedge\gcd(\varphi-1,p)=1]=\epsilon_{\mathcal{A}}\end{aligned}\tag{14}$$

5.3　黑盒可追踪性证明

定理 3　若所有PPT敌手仅能以可忽略优势ϵ赢得黑盒可追踪游戏，则方案是黑盒可追踪的。

证明　挑战者$\mathcal{C}$选择双线性群描述信息$\text{GD}=\{G,G_T,e,p,g\}$。挑战者$\mathcal{C}$随机挑选$L+2$随机元素$s,a,b_1,b_2,\cdots,b_L\in Z_p$，并构造L-DPBDHE元组Tu，如式(12)所示。敌手$\mathcal{A}$发送挑战访问策略$\Psi^*=\{\boldsymbol{M}^*,\rho^*\}$给$\mathcal{C}$，其中矩阵$\boldsymbol{M}$是一个$l^*$行$n^*$的矩阵。此后，敌手向挑战者查询解密密钥的属性集合均不满足Ψ^*。

设置：挑战者$\mathcal{C}$随机选择元素$\tilde{\alpha},\tilde{\beta}\in Z_p$并令$\alpha=\tilde{\alpha}+a,\beta=\tilde{\beta}+b,h=g^{a^L}$，其中元素$a$和$L$属于L-DPBDHE元组$Tu$。然后，挑战者$\mathcal{C}$选择哈希函数$H_0:\{0,1\}^*\rightarrow Z_p,H_1:\{0,1\}^*\rightarrow G,H:G_T\rightarrow\{0,1\}^*$与映射函数$f_2:[l]\rightarrow U_A$，并设置系统公共参数$\text{PP}=\{e,g,e(g,h)^{\alpha}=e(g,g)^{(\tilde{\alpha}+a)\cdot a^L},g^{\alpha}=g^{\tilde{\alpha}+a},g^{\beta}=g^{\tilde{\beta}+b},h=g^{a^L},h^{\beta}=g^{a^L(\tilde{\beta}+b)},f_2,H_0,H_1,H\}$和系统主密钥$\text{MSK}=\{\beta=\tilde{\beta}+b\}$。接着，$\mathcal{B}$运行$\text{Setup}_{\text{AA}_j}$算法得到属性授权机构的公私钥对$\{\text{APK}_j,\text{ASK}_j\}$。最后，$\mathcal{B}$将公共参数PP和属性授权机构公钥$\{\text{APK}_j\}$发送给$\mathcal{A}$。

阶段1：$\mathcal{A}$适应性地向$\mathcal{C}$发送密钥查询请求。

挑战：敌手$\mathcal{A}$发送给挑战者$\mathcal{C}$一条消息m。$\mathcal{C}$随机挑选一个比特$v\in\{0,1\}$。

若$v=0$，计算普通密文，即调用算法Encrypt加密明文数据m，具体步骤与5.1节的计算过程相同。最后，$\mathcal{C}$将密文CT_m发送给$\mathcal{A}$。

若$v=1$，计算追踪密文TCT，步骤如下。

1) $\mathcal{C}$计算$\overline{C}_0=m\cdot e(g,h)^{\bar{s}\alpha}=m\cdot R\cdot e(g^{\bar{s}},g)^{\tilde{\alpha}\cdot a^L}$和$\overline{C}_4=h^{\bar{s}}=g^{a^L\bar{s}},\overline{C}_5=h^{\beta\bar{s}}=g^{a^L\bar{s}(\tilde{\beta}+b)}$。

2) $\mathcal{C}$随机选择整数元素集合$\{v_i\}_{2\leqslant i\leqslant n}$与$\{w_i\}_{2\leqslant i\leqslant n}$，构造向量$\boldsymbol{v}=(\bar{s},\bar{s}a+v_2,\cdots,\bar{s}a^{n-1}+v_n)^{\text{T}}$和$\boldsymbol{w}=(0,w_2,\cdots,w_n)^{\text{T}}$。

3)对访问策略矩阵$\boldsymbol{M}^*$的第x行M_x^*，$\mathcal{C}$计算$\hat{\lambda}_x=\sum_{i\in[n]}M_{x,i}^*\hat{s}a^{i-1}+\sum_{i\in[2,n]}M_{x,i}^*\hat{s}\boldsymbol{v}_i=\sum_{i\in[n]}M_{x,i}^*\hat{s}a^{i-1}+\hat{\lambda}_x^*,\hat{\omega}_x=\sum_{i\in[n]}M_{x,i}^*w_i=\hat{\omega}_x^*$。

4)对访问策略矩阵$\boldsymbol{M}^*$的第i行$\boldsymbol{M}_i^*$，令$r_i=b_i,t=g^a,\gamma_{f_2(i)}=\hat{s}\cdot\tilde{\gamma}_{f_2(i)},H_0(\rho(i))=\dfrac{a^i}{b_i}$，则有$\hat{C}_{i,1}=g^{\hat{s}b_i\cdot\tilde{\gamma}_{f_2(i)}+\hat{\lambda}_i},\hat{C}_{i,2}=g^{b_i}(g^a)^{\hat{\omega}_i^*},\hat{C}_{i,3}=g^{a^i}$。

阶段2：与阶段1相似，敌手$\mathcal{A}$适应性地提交密钥查询请求，$\mathcal{C}$回复$\mathcal{A}$密钥SK。

猜测：敌手$\mathcal{A}$输出对于v的猜测v'，此处，有2种情况需要考虑。

若$v=v'=0$，即$\text{CT}_m=\text{Encrypt}(\cdot)$。从$\mathcal{A}$的视角看，挑战密文$\text{CT}_m$是一个合法的密文，$\mathcal{A}$获胜概率为$\Pr[v=v'|R=e(g,g)^{sa^{L+1}}]=\dfrac{1}{2}+\epsilon$。

若$v=v'=1$，即$\text{CT}_m=\text{BKEncrypt}(\cdot)$。从$\mathcal{A}$的视角看，挑战密文是群$G_T$上的随机元素，$\mathcal{A}$获胜的概率为$\Pr[v=v'|R=R_{G_T}]=\dfrac{1}{2}$。

总体来说，敌手$\mathcal{A}$赢得游戏的优势为$\left(\dfrac{1}{2}+\epsilon\right)-\dfrac{1}{2}=\epsilon$。由于$\epsilon$是不可忽略的，存在PPT敌手能以不可忽略优势解决L-DPBDHE问题，这与L-DPBDHE假设相冲突，所提方案的黑盒可追踪性得证。证毕。

6　性能分析

本节首先对本文方案与相关方案进行理论分析，然后实验测试方案的实际表现。

6.1　理论分析

本节挑选了相关的最新工作[13,15,24,28]进行对比。为方便起见，本节用ZYMB22表示文献[13]的方案，HMX23表示文献[15]的方案，FL25表示文献[24]的方案，KWY23表示文献[28]的方案。

表2~表4分别为功能、存储和计算3个方面的对比，其中，$|G|$、$|G_T|$分别表示群G、G_T中元素的长度，N_s表示属性空间中的属性个数，N_e表示参与加密的属性个数，N_u表示参与生成解密密钥的属性个数，N_I表示参与解密或参与黑盒追踪的属性个数，$|A|$表示属性授权机构个数，d表示用户ID的长度，E和E_T分别表示在群G、G_T上的指数运算时间开销，P表示配对运算时间开销。

表2 功能对比

方案	大属性空间	多授权机构	可追踪	策略表达
HMX23	×	√	白盒/黑盒	访问树
ZYMB22	√	×	白盒/黑盒	LSSS
KWY23	×	×	黑盒	LSSS
FL25	×	×	白盒	LSSS
本文方案	√	√	白盒/黑盒	LSSS

由表2可知，本文方案不但兼容白盒追踪和黑盒追踪，而且支持大属性空间和多授权机构。兼容白盒与黑盒追踪的特性，使得本文方案更适用于泄露密钥和解密设备这2种经典的密钥泄露场景。大属性空间的性质使得本文方案是动态可扩展的，允许系统运行时动态增加属性而不需要重新初始化整个系统。多授权机构的特性降低了单点故障风险。

由表3可知，所有方案的密文和密钥长度均随着访问策略的属性个数线性增长。其中，本文方案的密文长度比HMX23多$2|G|$，但比ZYMB22低；密钥长度与ZYMB22一致，比HMX23多$|G|$。虽然如此，但是HMX23方案不支持大属性空间，而本文方案和ZYMB22支持，从而不需要预先定义与属性相关的参数，其公共参数长度明显小于HMX23。

由表4可知，本文方案的加密开销高于HMX23和FL25，但低于ZYMB22；密钥生成开销方面与HMX23相近，且明显低于ZYMB22；解密开销略高于HMX23和FL25，但明显低于ZYMB22。本文方案的白盒追踪计算开销主要来源于密钥完整性检查，由于HMX23支持白盒追踪但没有给出具体的密钥完整性检查算法，而KWY23不支持白盒追踪，故表4仅列举了本文方案和ZYMB22、FL25的白盒追踪开销。黑盒追踪的开销包括解密设备执行解密和追踪者进行分析的开销。本文方案、HMX23和ZYMB22的追踪开销均正比于追踪的属性个数，其中本文方案和HMX23的追踪开销相近，皆低于ZYMB22。而KWY23的追踪开销与身份ID的长度d成正比，其代价无疑是所有方案中最高的。

表3 存储开销对比

方案	密文	密钥	公共参数
HMX23	$3N_e\|G\|+\|G_T\|$	$(2N_u+1)\|G\|$	$(N_s+4)\|G\|+\|G_T\|$
ZYMB22	$(4N_e+1)\|G\|+\|G_T\|$	$(2N_u+2)\|G\|$	$6\|G\|+\|G_T\|$
KWY23	$(2N_e+2d+1)\|G\|+\|G_T\|$	$(N_u+2d+2)\|G\|$	$(2+N_s+4d)\|G\|+\|G_T\|$
FL25	$(2N_e+4)\|G\|+\|G_T\|$	$(N_u+4)\|G\|$	$(N_s+7)\|G\|+\|G_T\|$
本文方案	$(3N_e+2)\|G\|+\|G_T\|$	$(2N_u+2)\|G\|$	$5\|G\|+\|G_T\|$

表4 计算开销对比

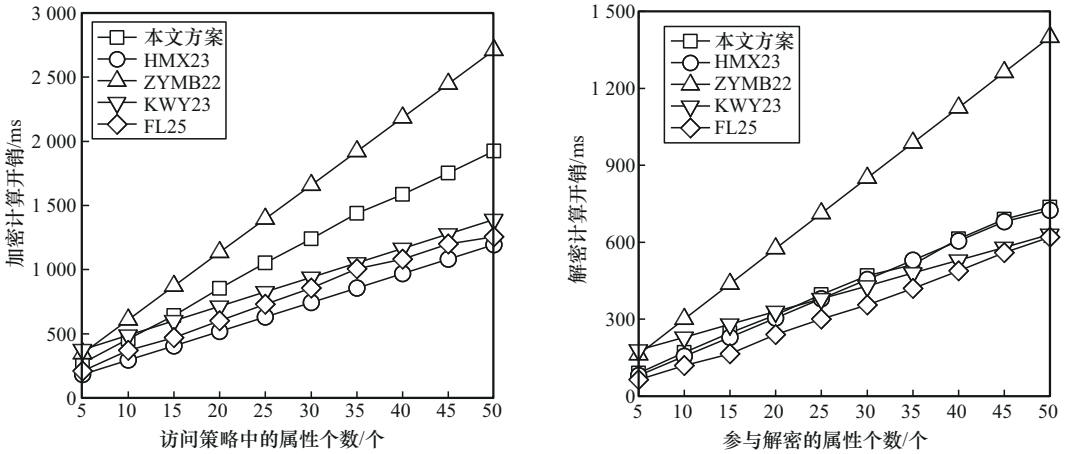
方案	加密	生成密钥	解密	白盒追踪	黑盒追踪
HMX23	$3N_eE+E_T$	$2(N_u+\|A\|)E$	$3N_IP$	—	$3N_IP+2E+E_T$
ZYMB22	$(7N_e+2)E$	$(3N_u+5)E$	$(4N_I+1)P+(2+N_I)E$	$(2N_u+3)P+(N_u+1)E$	$(4N_I+1)P+(2+N_I)E+E_T$
KWY23	$(3N_e+2+2d)E+E_T$	$(N_u+2d+2)E$	$(2N_I+2d+1)P+E_T$	—	$l\cdot((2N_I+2d+1)P+E_T)$
FL25	$(3N_e+5)E+E_T$	$(N_u+4)E$	$(2N_I+4)P$	$3P+E$	—
本文方案	$(5N_e+2)E+E_T$	$(2N_u+\|A\|+4)E$	$(3N_I+1)P$	$(N_u+\|A\|+2)P+N_uE$	$(3N_I+1)P+E_T$

6.2　实验分析

为进一步分析，本节对所有对比方案进行了仿真实验。实验中硬件环境为 Intel(R) Core(TM) i7-12700M CPU@2.1 GHz，RAM 为 16 GB，使用 Java 语言在 Windows 11 平台进行部署，选取基于双线性配对的 Java 密码学库（JPBC, Java pairing-based cryptography library）库中提供的 A 类椭圆曲线，抗碰撞哈希函数选择 SHA-256。由于 KWY23 的各项开销与用户 ID 长度 d 相关，参考 KWY23 后本文将 d 固定为 12。

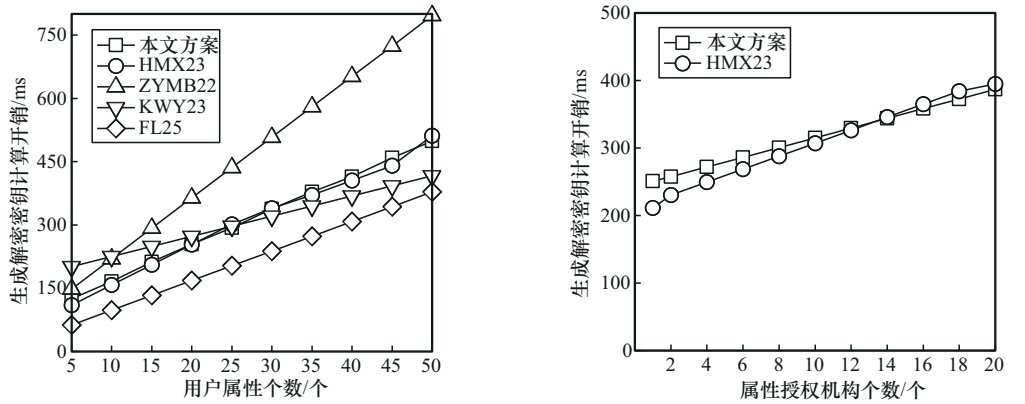
加密计算开销对比如图 4(a)所示。当访问策略中的属性个数较小时，各个方案的加密开销相近。随着属性个数的增长，ZYMB22 的加密计算开销最大，其次是本文方案，再次是 KWY23、FL25 和 HMX23。解密计算开销对比如图 4(b)所示，所有方案的解密计算开销均与参与解密的属性规模成正比。本文方案的解密计算开销与 HMX23 相近，ZYMB22 的解密计算开销最高，随属性个数的增长而快速增长。

图 5 为生成解密密钥的计算开销对比。由于本文方案和 HMX23 是支持多机构的 ABE 方案，故该阶段的实验由 2 个部分组成。首先，图 5(a)为仅 1 个属性授权机构参与时生成解密密钥计算开销随属性个数的变化情况。所有方案的生成解密密钥的计算开销均随属性个数线性增长，其中本文方案和 HMX23 的计算开销相近。当属性个数小于 25 时，本文方案生成解密密钥的计算开销高于 FL25，低于 KWY23 和 ZYMB22。当用户属性个数大于 25 时，本文方案略高于 KWY23 和 FL25，但远低于 ZYMB22。其次，图 5(b)为固定用户属性个数为 20 个时，生成解密密钥的计算开销随参与生成密钥的属性授权机构个数的变化情况。本文方案和 HMX23 的生成解密密钥计算开销随参与生成密钥的属性授权机构个数线性增长，当属性授权机构个数较小



(a) 访问策略中的属性个数与加密计算开销的关系　　(b) 参与解密的属性个数与解密计算开销的关系

图 4　加密和解密的计算开销对比

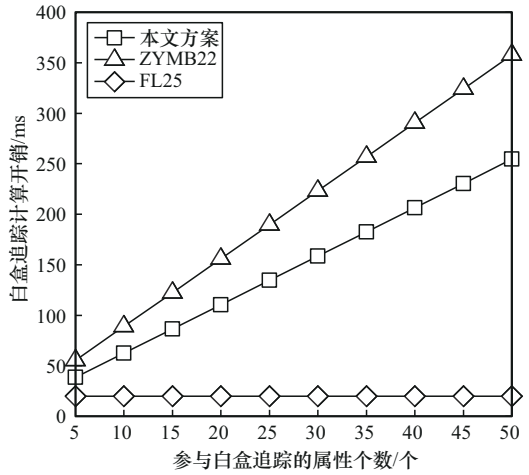


(a) 用户属性个数与生成解密密钥计算开销的关系　　(b) 属性机构个数与生成解密密钥计算开销的关系

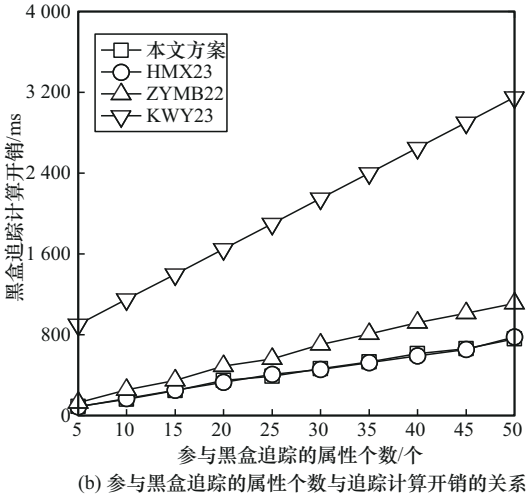
图 5　生成解密密钥的计算开销对比

时，本文方案的生成解密密钥计算开销略高于HMX23。但随着属性授权机构增多，二者计算开销逐渐相近。

图6(a)展示了白盒追踪的计算开销对比。为了方便比较，本文方案中的属性授权机构个数被固定为1。本文方案与ZYMB22的白盒追踪计算开销虽然皆随着参与白盒追踪的属性个数线性增长，但本文方案明显低于ZYMB22。由于FL25仅对部分密钥进行了完整性检查，其白盒追踪计算开销恒定。图6(b)为黑盒追踪计算开销结果，所有方案的黑盒追踪计算开销均随着参与追踪的属性个数线性增长，其中本文方案和HMX23的追踪计算开销相近，且均低于ZYMB22和KWY23。



(a) 参与白盒追踪的属性个数与追踪计算开销的关系



(b) 参与黑盒追踪的属性个数与追踪计算开销的关系

图6　白盒追踪和黑盒追踪的计算开销对比

表5列出了各方案的公共参数、密文和密钥的存储开销对比情况，为了方便比较，本文将属性个数和属性授权机构个数分别设置为10和1。由于支持大属性空间，本文方案和ZYMB22的公共参数的存储开销明显低于其他方案。本文方案的密文存储开销约为4 224 B，略高于HMX23和FL25，但低于其他方案。本文方案和ZYMB22的密钥存储开销相等，约为2 816 B，高于HMX23和FL25，但显著低于KWY23。另外，为更好地体现对大属性空间的支持特性，本文还给出了本文方案的存储开销与属性个数的关系，如表6所示。显然可见，本文方案的公共参数长度是恒定的，即与属性多少无关。本文方案的密文、密钥的存储开销均随着属性规模线性增长。

表5　　存储开销对比

方案	存储开销/B		
	公共参数	密文	密钥
HMX23	1 920	3 968	2 688
ZYMB22	896	5 376	2 816
KWY23	7 808	5 888	4 608
FL25	2 304	3 200	1 792
本文方案	1 152	4 224	2 816

表6　　本文方案的存储开销

属性个数	存储开销/B		
	公共参数	密文	密钥
10	1 152	4 224	2 816
20	1 152	8 064	5 376
30	1 152	11 904	7 936
40	1 152	15 744	10 496
50	1 152	19 584	13 056

7　结束语

本文提出一种可扩展且支持多种追踪机制的属性基加密方案。本文方案同时支持白盒可追踪和黑盒可追踪机制，对于泄露解密权限的恶意用户均能进行高效追踪。本文方案支持大属性空间的特性，公共参数的大小与属性空间无关，能够方便地实现加密系统的扩展。与此同时，本文方案基于多属性授权机构的设置，降低了单点故障风险。基于*l*-SDH假设证明了方案的白盒可追踪性，基于L-

DPBDHE 假设证明了方案的黑盒可追踪性和选择明文攻击下的密文不可区分性。经过实验对比发现，在支持多属性授权机构和大属性空间的基础上，本文方案的各项开销与最新方案相比均具有可比性。

参考文献：

[1] 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. New York: ACM Press, 2006: 89-98.

[2] BETHENCOURT J, SAHAI A, WATERS B. Ciphertext-policy attribute-based encryption[C]//Proceedings of the 2007 IEEE Symposium on Security and Privacy (SP'07). Piscataway: IEEE Press, 2007: 321-334.

[3] SUN J F, BAO Y Y, QIU W D, et al. Privacy-preserving fine-grained data sharing with dynamic service for the cloud-edge IoT[J]. IEEE Transactions on Dependable and Secure Computing, 2025, 22(2): 1329-1346.

[4] WANG H, XIE Y, LUO M, et al. EAPDS: efficient auditable and privacy-preservation data-sharing scheme based on attribute-based encryption for IoMT[J]. IEEE Internet of Things Journal, 2025, 12(14): 26844-26854.

[5] DIFFIE W, HELLMAN M. New directions in cryptography[J]. IEEE Transactions on Information Theory, 1976, 22(6): 644-654.

[6] LIU Z, CAO Z F, WONG D S. White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures[J]. IEEE Transactions on Information Forensics and Security, 2013, 8(1): 76-88.

[7] LIU Z, CAO Z F, WONG D S. Blackbox traceable CP-ABE: how to catch people leaking their keys by selling decryption devices on ebay[C]//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security - CCS'13. New York: ACM Press, 2013: 475-486.

[8] NING J T, CAO Z F, DONG X L, et al. Large universe ciphertext-policy attribute-based encryption with white-box traceability[C]//Proceedings of the 19th European Symposium on Research in Computer Security. Berlin: Springer, 2014: 55-72.

[9] QIAO H D, REN J C, WANG Z Y, et al. Compulsory traceable ciphertext-policy attribute-based encryption against privilege abuse in fog computing[J]. Future Generation Computer Systems, 2018, 88: 107-116.

[10] YANG Y B, ZHANG J W, LIU X M, et al. A scalable and auditable secure data sharing scheme with traceability for fog-based smart logistics[J]. IEEE Internet of Things Journal, 2023, 10(10): 8603-8617.

[11] ZHANG K, LI H, MA J F, et al. Efficient large-universe multi-authority ciphertext-policy attribute-based encryption with white-box traceability[J]. Science China Information Sciences, 2017, 61(3): 032102.

[12] HE X, LI L X, PENG H P. An enhanced traceable CP-ABE scheme against various types of privilege leakage in cloud storage[J]. Journal of Systems Architecture, 2023, 136: 102833.

[13] LIU Z H, DING Y Y, YUAN M, et al. Black-box accountable authority CP-ABE scheme for cloud-assisted E-health system[J]. IEEE Systems Journal, 2023, 17(1): 756-767.

[14] ZHOU J, CAO Z F, DONG X L, et al. TR-MABE: White-box traceable and revocable multi-authority attribute-based encryption and its applications to multi-level privacy-preserving e-healthcare cloud computing systems[C]//Proceedings of the 2015 IEEE Conference on Computer Communications (INFOCOM). Piscataway: IEEE Press, 2015: 2398-2406.

[15] NASIRAEE H, ASHOURI-TALOUKI M, LIU X M. Optimal black-box traceability in decentralized attribute-based encryption[J]. IEEE Transactions on Cloud Computing, 2023, 11(3): 2459-2472.

[16] ROUSELAKIS Y, WATERS B. Practical constructions and new proof methods for large universe attribute-based encryption[C]//Proceedings of the ACM Conference on Computer and Communications Security. New York: ACM Press, 2013: 463-474.

[17] LIN H, CAO Z F, LIANG X H, et al. Secure threshold multi authority attribute based encryption without a central authority[J]. Information Sciences, 2010, 180(13): 2618-2632.

[18] DUAN P F, MA Z F, GAO H M, et al. Multi-authority attribute-based encryption scheme with access delegation for cross blockchain data sharing[J]. IEEE Transactions on Information Forensics and Security, 2024, 20: 323-337.

[19] GONG B, GUO C, GUO C, et al. SLIM: a secure and lightweight multi-authority attribute-based signcryption scheme for IoT[J]. IEEE Transactions on Information Forensics and Security, 2023, 19: 1299-1312.

[20] BAGCHI P, BERA B, MAHESHWARI R, et al. An efficient and secure post-quantum multi-authority ciphertext-policy attribute-based encryption method using lattice[C]//Proceedings of the IEEE INFOCOM 2023-IEEE Conference on Computer Communications Workshops. Piscataway: IEEE Press, 2023: 1-6.

[21] SAHAI A, WATERS B. Fuzzy identity-based encryption[C]//International Conference on Theory & Applications of Cryptographic Techniques. Berlin: Springer, 2005: 457-473.

[22] 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. New York: ACM Press, 2007: 195-203.

[23] WATERS B. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization[C]//International Workshop on Public Key Cryptography. Berlin: Springer, 2011: 53-70.

[24] MENG F, CHENG L X. TSR-ABE: traceable and server-aided revocable ciphertext-policy attribute-based encryption under static assumptions[J]. IEEE Transactions on Information Forensics and Security,

2024, 20: 955-967.

[25] MENG F, CHENG L X. STR-ABKS: server-aided traceable and revocable attribute-based encryption with keyword search[J]. IEEE Internet of Things Journal, 2024, 11(7): 12649-12659.

[26] LIU Z, WONG D S. Traceable CP-ABE on prime order groups: fully secure and fully collusion-resistant blackbox traceable[C]//International Conference on Information and Communications Security. Berlin: Springer, 2016: 109-124.

[27] NING J T, CAO Z F, DONG X L, et al. Traceable and revocable CP-ABE with shorter ciphertexts[J]. Science China Information Sciences, 2016, 59(11): 119102.

[28] FAN K, LI W H, BAI Y H, et al. EIV-BT-ABE: efficient attribute-based encryption with black-box traceability based on encrypted identity vector[J]. IEEE Internet of Things Journal, 2024, 11(9): 15229-15240.

[29] QU Z, KUMARI S, OBAIDAT M S, et al. Traceable attribute-based encryption with equality test for cloud enabled E-health system[J]. IEEE Journal of Biomedical and Health Informatics, 2024, 28(9): 5033-5042.

[30] HAN D Z, PAN N N, LI K C. A traceable and revocable ciphertext-policy attribute-based encryption scheme based on privacy protection[J]. IEEE Transactions on Dependable and Secure Computing, 2022, 19(1): 316-327.

[作者简介]

谢晴晴（1990– ），女，安徽宿州人，博士，江苏大学讲师，主要研究方向为区块链、应用密码学。

朱法铜（2000– ），男，江苏盐城人，江苏大学硕士生，主要研究方向为区块链、应用密码学。

冯霞（1983– ），女，江苏镇江人，博士，海南大学教授，主要研究方向为物联网安全、区块链、应用密码学。