可追踪并撤销属性的密文策略属性基加密方案

荣 静[1], 殷新春[2]
(1. 扬州大学广陵学院, 江苏 扬州 225009; 2. 扬州大学信息工程学院, 江苏 扬州 225009)

摘 要: 针对属性基加密系统中用户权限变更、用户故意或无意地泄露自己的密钥信息等行为,提出了一种可以追踪用户并撤销其属性的密文策略属性基加密(ciphertext policy attribute-based encryption,CP-ABE)方案,即通过算法追踪到用户身份后,撤销该用户属性集合中的某一个或几个属性,从而实现取消用户相应权限的目的. 利用 Shamir 门限方案等技术设计追踪算法追踪用户的 ID,将用户的 ID 添加到相关属性的撤销列表中,加密者在加密时输入所涉及的每个属性的撤销列表,从而实现追踪用户并对其属性进行细粒度的直接撤销,利用对偶系统加密技术证明了该方案是选择安全的.

关键词: 密文策略属性基加密; 用户追踪; 直接撤销; 属性撤销; 撤销列表; 选择安全
中图分类号: TP 309 **文献标志码**: A **文章编号**: 0254 - 0037(2019)02 - 0143 - 10
doi: 10.11936/bjutxb2017060061

Ciphertext Policy Attribute-based Encryption Scheme With Tracing and Attribute Revocation

RONG Jing[1], YIN Xinchun[2]
(1. Guangling College, Yangzhou University, Yangzhou 225009, Jiangsu, China;
2. College of Information and Engineering, Yangzhou University, Yangzhou 225009, Jiangsu, China)

Abstract: Focused on the issue that users' privilege changes and users leak their private keys intentionally or unintentionally in attribute-based encryption (ABE) system, a traceable and attribute revocable scheme was proposed in ciphertext policy attribute-based encryption (CP-ABE). After tracing user's ID, the user's one or several attributes was revoked to control the user's privilege. The technology Shamir threshold scheme was used to trace the user's ID and put the ID into the attributes revocation lists. The attributes revocation lists were input by the encryptor when running the encrypt algorithm. Finally tracing and direct fine grained revocation were accomplished, and the scheme selective security in the dual system encryption was proved.

Key words: ciphertext policy attribute-based encryption; user traceable; attribute revocation; directe revocation; revocation list; selective security

基于属性的密码技术是通过将属性集合与用户私钥或者密文关联起来实现加密解密. 其在实现细粒度的访问控制等方面具有不可取代的优势,目前应用基于属性的密码体制的领域越来越多,如医疗健康数据[1]、云存储数据[2-4]等. 属性基加密分为密文策略属性基加密(ciphertext policy attribute-based encryption,CP-ABE)和密钥策略属性基加密(key policy attribute-based encryption, KP-ABE). 在 CP-

收稿日期: 2017-06-29
基金项目: 国家自然科学基金资助项目(61472343)
作者简介: 荣 静(1990—), 女, 助教, 主要从事信息安全方面的研究, E-mail:714539271@qq.com

ABE 方案中,加密方可以根据自己的需求设定一个访问策略,仅当解密方的密钥能够满足这一访问策略时方能解密成功. 基于这种特性,CP-ABE 机制更符合实际生产生活中的情况,因此,对 CP-ABE 的研究也更深入而广泛.

在实际的生产生活应用中常常存在用户的权限变更、用户故意或无意地泄露自己的私钥等情况,此时便需要在属性基加密(attribute-based encryption,ABE)方案中添加追踪与撤销机制. 追踪机制[5-7]主要用于追踪可疑用户,如当用户参与共谋攻击、用户为私利卖出自己的密钥等情况时,系统可以追踪到用户从而进一步处理该用户. 文献[5]提出了一种白盒可追踪的 CP-ABE 方案,该方案支持任意的单调访问结构,并在合数阶双线性映射下证明了此方案在密文不可区分的选择明文攻击(ciphertext indistinguishability under chosen phaintext attacks,IND-CPA)下是安全的,该算法通过引入一个标识表来存放用户的 ID;文献[7]提出了一种黑盒可追踪的 CP-ABE 方案,并在标准模型下证明了方案的安全性. 当发生用户私钥泄露、用户参与了共谋攻击或者用户离岗、用户的权限变更等情况时,便需要撤销机制[8-12]. 撤销机制按分类方式不同可以分为直接撤销和间接撤销及用户撤销和属性撤销. 用户撤销可以通过撤销用户的 ID 等来实现,属性撤销在实现时比用户撤销复杂一些,且属性撤销可以更加灵活地改变用户权限. 文献[11]提出了一种可以直接撤销用户的 CP-ABE 方案,该方案采用线性秘密分享和二叉树技术,在标准模型下证明了方案的安全性;文献[12]提出了一个支持属性层用户撤销且将解密外包的 ABE 方案,主要采用线性秘密分享方案和密钥加密密钥(key encrypting key,KEK)树来实现;文献[9]提出了一种支持细粒度直接撤销用户属性的 CP-ABE 方案,该方案基于对偶系统加密技术,在合数阶双线性映射下是选择安全的. 同样,文献[13]也提出了一种可追踪并撤销的密文策略属性基加密方案,但其所实现的直接撤销方案是针对用户的. 本文利用 Shamir 门限方案等技术实现了一种在 CP-ABE 中可以追踪用户并对其属性进行细粒度的直接撤销的方案,即系统检测出某一用户私钥存在可疑行径后,通过 Shamir 门限技术追踪到用户的 ID,然后撤销用户属性集合中的相关属性,以实现相应权限的撤销,从而达到不撤销用户,仅改变其访问权限的目的;该方案还可以应用于系统中用户权限变更等情况,如某企业员工被降职,便可以撤销其某些属性以实现其访问权限的变更.

1 背景知识

1.1 合数阶双线性映射

定义 1 合数阶双线性映射[14]. 令 $\mathcal{G}$ 表示一个群生成元,该 $\mathcal{G}$ 算法输入安全参数 λ,输出双线性群 G 的相关描述,表述为 (N, G, G_T, e). 其中:$N = p_1p_2p_3$ 是 3 个互不相同的素数 p_1、p_2、p_3 的乘积;G 和 G_T 是 2 个阶为 N 的循环群;$e: G \times G \rightarrow G_T$ 是一个满足以下 2 个性质的双线性映射:

1) 双线性:对 $\forall a, b \in Z_N$, $\forall u, v \in G$,有 $e(u^a, v^b) = e(u,v)^{ab}$.

2) 非退化性:$\exists h \in G$,使得 $e(h,h)$ 在群 G_T 中的阶为 N.

在以上定义中,双线性映射 e 是对称的,即 $e(u^a, v^b) = e(u,v)^{ab} = e(u^b, v^a)$. 假定 G、G_T 和 e 中的运算都是关于参数 λ 可有效计算的,令 G_{p_1}、G_{p_2} 和 G_{p_3} 分别表示阶为 p_1、p_2 和 p_3 的 G 的子群,这些子群在双线性映射 e 下是相互正交的,即若 $h_i \in G_{p_1}$, $h_j \in G_{p_2}$,且 $i \neq j$,那么 $e(h_i, h_j)$ 在 G_T 中是单位元素,即 $e(h_i, h_j) = 1$. 若 g_1 生成 G_{p_1},g_2 生成 G_{p_2},g_3 生成 G_{p_3},那么 G 中的每个元素 h 都可表示为 $g_1^{c_1}g_2^{c_2}g_3^{c_3}$, $c_1, c_2, c_3 \in Z_N$,并称 $g_i^{c_i}$ 为 h 中的 G_{p_i} 部分.

1.2 单调访问结构

定义 2 访问结构[15]. 将参与者的集合记为 $\{p_1, p_2, \cdots, p_n\}$,若对于 $\forall B, C$ 有:如果 $B \in A$ 且 $B \subseteq C$,则 $C \in A$,称集合 $A \subseteq 2^{\{p_1, p_2, \cdots, p_n\}}$ (A 不为空)是单调的. 在 A 中的集合称为授权子集,不在 A 中的集合称为未授权子集.

1.3 线性秘密分享方案

由于线性秘密分享方案(linear secret share scheme, LSSS)[15]能够实现任意的单调访问结构,所以本文将采用 LSSS 来实现访问结构.

定义 3 LSSS. 属性空间 U 上的一个秘密分享方案称为线性的,如果:

1) 每个属性上的分享构成一个向量.

2) 存在一个 $l \times n$ 的矩阵 $\boldsymbol{M}$,对于所有 $i \in \{1, 2, \cdots, l\}$,$\boldsymbol{M}_i$ 是矩阵 $\boldsymbol{M}$ 的第 i 行,ρ 是集合 $\{1, 2, \cdots, l\}$ 到 U 的映射函数. 列向量 $\boldsymbol{v} = (s, v_2, \cdots, v_n)^{\perp}$ 中,$s \in Z_N$ 是待分享的秘密,$v_2, \cdots, v_n \in Z_N$ 是随机选取的值,则 $\boldsymbol{M} \cdot \boldsymbol{v}$ 表示秘密 s 的 l 个分享,$\boldsymbol{M}_i \cdot \boldsymbol{v}$ 表示属性

$\rho(i)$分享的秘密值.

根据文献[15]，LSSS 可以线性恢复秘密：令 $\prod$ 表示访问结构 A 的秘密分享方案，$S \in A$ 表示经授权的属性集合. 定义一个集合 $I = \{i \mid \rho(i) \in S \wedge i \in \{1,2,\cdots,l\}\}$，则存在常量集合 $\{\omega_i \in Z_N\}_{i \in I}$，使得对任何有效的分享 $\{\lambda_i \in \boldsymbol{M}_i \cdot \boldsymbol{v}\}_{i \in I}$，依据秘密分享方案 $\prod$ 有 $\sum_{i \in I} \omega_i \lambda_i = s$，该常量集合 $\{\omega_i\}$ 可根据矩阵 $\boldsymbol{M}$ 在多项式时间内获得. 未授权集合则得不到该常量集合. 本文中将采用 Z_N 上的秘密分享方案.

1.4　困难性假设

l-SDH(l-strong Deffie-Hellman)假设[16]：假设 G 是一个阶为素数 p 的双线性群，g 是 G 的一个群生成元，随机选取 $x \in Z_p^*$，构成一个 $l+1$ 元组 $(g, g^x, g^{x^2}, \cdots, g^{x^l})$，将其作为输入，输出一个配对

$$(c, g^{1/(c+x)}) \in Z_p \times G$$

若 $\Pr[\mathscr{B}(g, g^x, g^{x^2}, \cdots, g^{x^l}) = (c, g^{1/(c+x)})] \geqslant \varepsilon$，则算法 $\mathscr{B}$ 解决 l-SDH 问题的优势为 ε.

定义 4　若在 t 时间内没有算法能够在 G 中拥有至少 ε 的优势解决 l-SDH 问题，则 (l, t, ε)-SDH 假设成立.

定理 1[17]　令 N 等于 m 个两两互不相同的素数的乘积(即 $N = p_1 p_2 \cdots p_m$)，对任意的属性 $i \in \{1, 2, \cdots, m\}$，有 $p_i > 2^\lambda$，得到的合数阶双线性群为 (N, G, G_T, e). 令 $\{A_i\}$ 是群 G 上的随机变量，$\{B_i\}$、T_0、T_1 是群 G_T 上的随机变量，且所有随机变量的阶小于等于 t，从而得到 $(N, G, G_T, e, \{A_i\}, \{B_i\})$. 在一般模型下，给出算法 $\mathscr{B}$，随机选取 T_b(其中 $b \in \{0, 1\}$)，$\mathscr{B}$ 猜测 b 的值为 b'，设猜对的概率为 $\delta = |\Pr[b' = b] - 1/2|$. 若 T_b 独立于 $\{B_i\} \cup \{e(A_i, A_j)\}$ 且算法 $\mathscr{B}$ 在上述实验下最多进行了 q 次，则可以构造一个算法 $\mathscr{B}'$，$\mathscr{B}'$ 能够以至少 $\delta - O(q^2 t / 2^\lambda)$ 的概率分解 N.

假设 2[18]　在合数阶双线性群 (N, G, G_T, e) 下，取 G_{p_1} 的生成元 g 和 G_{p_3} 的生成元 Y，得到 (N, G, G_T, e, g, Y)，若不存在算法可以在概率多项式时间内区分 G_{p_1} 和 $G_{p_1 p_2}$ 中的元素，则该假设成立.

假设 3[18]　在合数阶双线性群 (N, G, G_T, e) 下，取 G_{p_1} 的生成元 g、G_{p_2} 的生成元 X 和 G_{p_3} 的生成元 Y，然后随机选取 $s, c_1, c_2, d \in Z_N$，得到 $(N, G, G_T, e, g, g^s X^{c_1}, Y, X^{c_2} Y^d)$. 若不存在算法可以在概率多项式时间内区分 G 和 $G_{p_1 p_3}$ 中的元素，则该假设成立.

假设 4[13]　在合数阶双线性群 (N, G, G_T, e) 下，取 G_{p_1} 的生成元 g、G_{p_2} 的生成元 X 和 G_{p_3} 的生成元 Y，然后随机选取 $s, c_1, c_2, \alpha \in Z_N$，令 $U = \{1, 2, \cdots, n\}$，得到参数：$(g, X, g^s X^{c_1}, Y, \{g_i = g^{\alpha^i}\}_{i \in U}, g^{\alpha^{n+1}} X^{c_2}, T)$. 若不存在算法可在概率多项式时间内区分 $e(g,g)^{\alpha^{n+1} s}$ 和 G_T 中的一个随机元素，假设成立.

1.5　Shamir$(\bar{t}, \bar{n})$ 门限方案

定义 5[19]　$\bar{t}-1$ 阶曲线上的 $\bar{t}$ 个点足以确定这条曲线，即 $\bar{t}$ 个点能够确定一个 $\bar{t}-1$ 阶多项式. 在一个 Shamir$(\bar{t}, \bar{n})$ 门限方案中，一个秘密值可能会被分成 $\bar{n}$ 个或更多个部分，并将某一特定部分分给每个参与者，从而用来重构秘密值. 假设秘密值是有限域 F_p^* 中的一个元素，随机选取 $\bar{t}-1$ 个系数 $a_1, a_2, \cdots, a_{\bar{t}-2} \in F_p$ 且 $a_{\bar{t}-1} \in F_p^*$，令 a_0 为秘密值，得到的多项式为：$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{\bar{t}-1} x^{\bar{t}-1}$. 每个参与者分得一个上述多项式所表示的曲线上的点 (x, y)，其中 $y = f(x)$，给出含有 $\bar{t}$ 个点的子集，应用拉格朗日插值可以恢复出 a_0 的值.

1.6　算法模型

本文所提出的可追踪并撤销用户属性的密文策略属性基加密方案(T-RA-CPABE)主要从以下 5 个算法来实现.

Setup$(1^\lambda, m, n) \rightarrow$(PK, MSK)：系统建立算法. 输入安全参数 λ、系统中属性的数量 m 和用户的数量 n，输出系统的公钥参数 PK、系统的主私钥 MSK. 其中 PK 中蕴含了系统中的属性集合 $I = \{1, 2, \cdots, m\}$ 和用户的集合 $U = \{1, 2, \cdots, n\}$；系统初始化一个 Shamir$(\bar{t}, \bar{n})$ 门限方案实例 $\mathrm{INS}_{(\bar{t}, \bar{n})}$.

KeyGen$(\mathrm{ID}, \omega, \mathrm{MSK}, \mathrm{PK}) \rightarrow \mathrm{SK}_{\mathrm{ID}, \omega}$：密钥生成算法. 输入用户的 $\mathrm{ID} \in U$，该用户的属性集合 $\omega \subseteq I$ 以及公钥参数 PK、系统主私钥 MSK，输出用户 ID 的关于属性集合 ω 的私钥 $\mathrm{SK}_{\mathrm{ID}, \omega}$.

Encrypt$(M, A, \{R_i\}_{i \in \omega}, \mathrm{PK}) \rightarrow C$：加密算法. 输入明文 M，属性集合 I 上的访问结构 A，每一个属性 $i (i \in \omega)$ 的用户撤销列表 $\{R_i\}$ 和系统的公钥参数 PK，输出密文 C，C 中蕴含了访问结构 A 以及所涉及的每个属性的撤销列表.

Decrypt$(\mathrm{SK}_{\mathrm{ID}, \omega}, C, \mathrm{PK}) \rightarrow M$ 或特殊符号 $\perp$：解密算法. 输入用户私钥 $\mathrm{SK}_{\mathrm{ID}, \omega}$、密文 C 和公钥参数 PK，设 ω' 为用户 ID 所具备的与访问结构 A 相关且尚未被撤销的属性的集合，若 ω' 满足访问结构 A，则输出明文 M；否则输出 $\perp$.

Trace$(\mathrm{PK}, \mathrm{INS}_{(\bar{t}, \bar{n})}, \mathrm{SK}, \mathrm{MSK}) \rightarrow$ID 或特殊符号 $\perp$：追踪算法. 输入公钥参数 PK，系统的主私钥 MSK，用户的私钥 SK 和一个 Shamir 门限方案

$\mathrm{INS}_{(\bar{t},\bar{n})}$,若 SK 满足密钥检测且能够恢复 $\mathrm{INS}_{(\bar{t},\bar{n})}$ 的秘密,则输出与密钥相对应的 ID,否则输出 $\perp$.

1.7 安全模型

针对本文所提出的算法,主要使用 2 种安全模型,分别用来证明算法的选择安全性和算法中追踪机制的安全性. 算法的选择安全性主要依据可证明安全模型,具体见 1.7.1;可追踪性安全模型见 1.7.2.

1.7.1 选择安全模型

Init:攻击者声明将要挑战的访问结构 A^*,并对任意的 $x\in\{1,2,\cdots,l\}$,指定属性 $\rho(x)$ 的用户撤销列表 $R^*_{\rho(x)}$(l 和 ρ 的含义见 1.3).

Setup:挑战者通过调用 T-RA-CPABE 方案中的 Setup 算法,将得到的公钥参数 PK 发送给攻击者.

Phase 1:攻击者请求某用户(ID)关于属性集合 ω 的私钥,限制是属性集合 $\omega'=\{i\mid i\in\omega\cap\mid A^*\mid,\mathrm{ID}\notin R_i^*\}$ 不可满足 A^*,挑战者通过 T-RA-CPABE 方案中的 KeyGen 算法求得私钥 SK,并发送给攻击者.

Challenge:攻击者提交 2 个等长的明文 M_0 和 M_1,挑战者随机选取一个明文并加密,并将得到的密文 C 发送给攻击者.

Phase 2:同 Phase 1.

Guess:攻击者猜测被加密的明文 M 是 M_0 还是 M_1,猜对则说明攻击者获胜.

定义 6 当且仅当任意概率多项式时间内攻击者获胜的概率是可以忽略的,本文中所提出的 T-RA-CPABE 方案是选择安全的.

1.7.2 可追踪性安全模型

Setup:挑战者运行 T-RA-CPABE 方案中的 Setup 算法并将得到的系统公钥参数发送给攻击者.

Key Query:攻击者多次请求与属性集合相应的密钥.

Key Forgery:攻击者输出解密的密钥 SK_*. 若 SK_* 满足密钥检测(密钥检测是为了检测密钥是否合法),且 $\mathrm{Trace}(\mathrm{PK},\mathrm{INS}_{(\bar{t},\bar{n})},\mathrm{MSK},\mathrm{SK}_*)\notin\{\mathrm{ID}_1,\cdots,\mathrm{ID}_q\}$,则攻击者赢得游戏. 将攻击者赢得游戏的优势定义为:$\Pr[\mathrm{Trace}(\mathrm{PK},\mathrm{INS}_{(\bar{t},\bar{n})},\mathrm{MSK},\mathrm{SK}_*)\notin\{\perp,\mathrm{ID}_1,\cdots,\mathrm{ID}_q\}]$.

定义 7 若在概率多项式时间内没有攻击者能够在上述游戏中获取一个不可忽略的优势,则 T-RA-CPABE 方案中的可追踪算法安全.

2 方案具体实现

本节主要介绍 T-RA-CPABE 方案的具体实现,包括 5 个算法:Setup、KeyGen、Encrypt、Decrypt 和 Trace.

$\mathrm{Setup}(\lambda,m,n)\rightarrow\mathrm{PK},\mathrm{MSK}$:输入一个安全参数 λ、系统中属性的数量 m 和用户的数量 n,输出系统的公钥参数 PK,系统的主私钥 MSK. 算法运行群生成元 $\mathscr{G}$ 得到 (N,G,G_T,e),其中合数 $N=p_1p_2p_3$(p_1、p_2、p_3 是 3 个两两不同的素数),$e:G\times G\rightarrow G_T$ 是双线性映射. g 和 Y 分别是 G_{p_1} 子群和 G_{p_3} 子群的生成元.

令 $I=\{1,2,\cdots,m\}$ 表示属性集合,$U=\{1,2,\cdots,n\}$ 表示用户集合. 算法随机选取 $\alpha,a\in Z_N$;对任意的属性 $i\in I$,算法随机选取 $\tau_i,\gamma_i\in Z_N$,计算 $f_i=g^{\tau_i}$,$h_i=g^{\gamma_i}$;对任意 $j\in U$,计算 $g_j=g^{(\omega^j)}$.

系统选择一个概率加密方案[20](Enc,Dec),通过 2 个不同密钥 $\bar{k}_1$、$\bar{k}_2$ 将一个二进制串转换为 Z_N^*,算法初始化一个 Shamir $(\bar{t},\bar{n})$ 门限方案实例 $\mathrm{INS}_{(\bar{t},\bar{n})}$,并将 $f(x)$ 和 $\bar{t}-1$ 个点 $\{(x_1,y_1),(x_2,y_2),\cdots,(x_{\bar{t}-1},y_{\bar{t}-1})\}$ 保密. 得到系统的公钥参数为

$$\mathrm{PK}=\{N,g,g^a,e(g,g)^\alpha,\{f_i\}_{i\in I},\{h_i\}_{i\in I},\{g_j\}_{j\in U}\}$$

系统的主私钥为

$$\mathrm{MSK}=\{\alpha,a,\{\tau_i,\gamma_i\}_{i\in I},\bar{k}_1,\bar{k}_2,Y\}$$

从上面的参数可以看出:τ_i、f_i 和 γ_i、h_i 都是关于属性的参数,τ_i 和 f_i 是属性自身的相关计算参数,而 γ_i 和 h_i 是属性撤销的相关计算参数;α、g_j 是用户身份的相关计算参数;Y 可以随机化密钥.

$\mathrm{KeyGen}(\mathrm{ID},\omega,\mathrm{MSK},\mathrm{PK})\rightarrow\mathrm{SK}_{\mathrm{ID},\omega}$:输入某一用户的 $\mathrm{ID}\in U$、该用户的属性集合 $\omega\subseteq I$ 以及 PK 和 MSK,输出该用户的私钥 $\mathrm{SK}_{\mathrm{ID},\omega}$. 系统计算:$x=\mathrm{Enc}_{\bar{k}_1}(\mathrm{ID})$,$y=f(x)$,$c=\mathrm{Enc}_{\bar{k}_2}(x\parallel y)$,此时 c 就相当于一个随机值. 系统随机选取 $t\in Z_N$、$r_i\in Z_N$(任意属性 $i\in\omega$)及 $Y_0,Y_1,Y_2,Y_{i,1},Y_{i,2}\in G_{p_3}$,计算:$K_0=g^{\alpha/(a+c)}Y_0$,$K=c$,$L_0=(g^t)^{1/(a+c)}Y_1$,$L=(g^t)^{a/(a+c)}Y_2$,$K_{i,1}=g^{at+\alpha^{\mathrm{ID}}\gamma_i+\tau_i r_i}Y_{i,1}$,$K_{i,2}=g^{r_i}Y_{i,2}$,最终得到的用户私钥为

$$\mathrm{SK}_{\mathrm{ID},\omega}=(K_0,K,L_0,L,K_{i,1},K_{i,2})$$

在 $K_{i,1}$ 中包含了所有与属性相关的参数:τ_i、f_i、γ_i 和 h_i,且还包含了用户的 ID,由此 $K_{i,1}$ 将用户的身份与属性绑定在了一起.

$\mathrm{Encrypt}(M,(\boldsymbol{A}_{l\times k},\rho),\{R_{\rho(x)}\}_{x\in\{1,2,\cdots,l\}},\mathrm{PK})\rightarrow C$:输入某明文 M,属性集合 I 上的访问结构 $(\boldsymbol{A},\rho)$,其中 $\boldsymbol{A}$ 是 $l\times k$ 矩阵,ρ 是矩阵 $\boldsymbol{A}$ 的每一行到属性的映射,且 ρ 不能将 $\boldsymbol{A}$ 中的 2 行映射为同一个属性;$R_i\subseteq U$ 是每一个属性 $i\in\omega$ 的用户撤销列表,算法输

出密文 C. 随机选取向量 $\boldsymbol{v}=(s,v_2,v_3,\cdots,v_k)\in Z_N^k$，令 $\boldsymbol{A}_x$ 表示矩阵 $\boldsymbol{A}$ 中的第 x 行，则根据 LSSS，$\lambda_x=\boldsymbol{A}_x\cdot\boldsymbol{v}$ 是每一行分享得到的秘密值，计算：$C_0=Me(g_1,g_n)^s e(g,g)^{\alpha s}$，$C_1=g^s$，$C_2=(g^a)^s$，$C_{x,0}=g^{\lambda_x}$，$C_{x,1}=f_{\rho(x)}^{\lambda_x}$.

令 $S_{\rho(x)}=U-R_{\rho(x)}$，则

1）若 $S_{\rho(x)}\neq U$，随机选取 $\eta_x,s_x\in Z_N$，计算

$$C_{x,2}=g^{\eta_x}\left(h_{\rho(x)}\prod_{j\in S_{\rho(x)}}g_{n+1-j}\right)^{\lambda_x}$$

$$C_{x,3}=g^{s_x}$$

$$C_{x,4}=g^{\eta_x}\left(\prod_{j\in R_{\rho(x)}}g_{n+1-j}\right)^{s_x}$$

2）若 $S_{\rho(x)}=U$，则令 $\eta_x=s_x=0$，计算

$$C_{x,2}=\left(h_{\rho(x)}\prod_{j\in S_{\rho(x)}}g_{n+1-j}\right)^{\lambda_x}$$

$$C_{x,3}=1$$

$$C_{x,4}=1$$

最后得到的密文为

$$C=(C_0,C_1,C_2,\{C_{x,0},C_{x,1},C_{x,2},C_{x,3},C_{x,4},R_{\rho(x)}\}_{x\in\{1,2,\cdots,l\}})$$

Decrypt$(\mathrm{SK}_{\mathrm{ID},\omega},C,\mathrm{PK})\to M$ 或 $\perp$：输入用户私钥 $\mathrm{SK}_{\mathrm{ID},\omega}$，密文 C 和系统的公钥参数 PK，解析 $\mathrm{SK}_{\mathrm{ID},\omega}$ 和密文 C，令 $X=\{x\mid\rho(x)\in\omega,\mathrm{ID}\notin R_{\rho(x)}\}$，则 $\omega'=\{\rho(x)\}$，此时 ω' 即是该用户与访问结构 A 相关的未被撤销的属性的集合，若 ω' 不满足访问结构，则输出 $\perp$；否则，令 μ_x 为矩阵 $\boldsymbol{A}$ 的第 x 行恢复系数，计算

$$\prod_{x\in X}\left(\frac{e(K_{\rho(x),1},C_{x,0})}{e(K_{\rho(x),2},C_{x,1})}\frac{e\left(C_{x,0},\prod_{j\in S_{\rho(x)},j\neq\mathrm{ID}}g_{n+1-|j-\mathrm{ID}|}\right)}{e(g_{\mathrm{ID}},C_{x,2})}\cdot\frac{e(g_{\mathrm{ID}},C_{x,4})}{e\left(C_{x,3},\prod_{j\in R_{\rho(x)}}g_{n+1-|j-\mathrm{ID}|}\right)}\right)^{\mu_x}=\frac{e(g^{at},g^s)}{e(g_1,g_n)^s}$$

所以可以得到明文

$$M=C_0\frac{1}{e(K_0,C_1^KC_2)}\frac{1}{e(L_0^KL,C_2)}E$$

在上面的计算过程中，由于用户私钥中嵌入的随机值 Y 都是子群 G_{p_3} 中的元素，根据合数阶群的定义，不同的子群之间存在两两正交关系，因此，随机值 Y 可以被约去.

系统判断是否密钥泄露、是否追踪用户的方法如下：

1）时刻检测密钥使用情况[21]. 对于每一个密钥的访问时间段，系统可以通过分析数据得到一个范围并记录下来，一旦检测到某一个密钥在其他时间段多次访问时，便执行 2），比如某一密钥持有者通常是工作日上午 9:00 访问，这一段时间却经常在凌晨 6:00 也有访问痕迹，这便有所异常；对于某一文件，系统记录该密钥通常访问频率是每周 1 次，却突然在同一天内频繁访问，执行 2）；对于某些非常重要的敏感文件，系统不仅监测其访问时间和频次，还会监测其访问地址，若某一密钥频繁在午夜访问，或突然大量频繁地访问、访问地址变更等，执行 2）；除了系统检测，当用户发现自己可以打开的某些共享文件被其他未被授权的人打开并下载（可以通过访问的异常 IP 等信息来判断）时，也可以由系统进行追踪用户；等等.

2）对该密钥进行完整性检测. 若该密钥形式如下：$\mathrm{SK}=(K_0,K,L_0,L,K_{i,1},K_{i,2})$，且通过下面的密钥检测，则追踪，并将追踪信息记录下来.

完整性检测. 对于每一个解密密钥，其必须满足以下条件：

1）$K\in Z_N,K_0,L_0,L\in G$.

2）$e(g,L)=e(g^a,L_0)\neq 1$.

3）$e(g^ag^K,K_0g^t)=e(g,g)^{\alpha}e(L_0^KL,g)\neq 1$.

Trace$(\mathrm{PK},\mathrm{INS}_{(\bar{t},\bar{n})},\mathrm{SK},\mathrm{MSK})\to\mathrm{ID}$ 或 $\perp$：追踪算法，系统将进行如下操作[6]：

1）由 SK 中的 $x\parallel y=\mathrm{Dec}_{\bar{k}_2}(K)$ 抽取得到 $(x^*=x,y^*=y)$.

2）若 $(x^*,y^*)\in\{(x_1,y_1),(x_2,y_2),\cdots,(x_{\bar{t}-1},y_{\bar{t}-1})\}$，则通过计算 $\mathrm{Dec}_{\bar{k}_1}(x^*)$ 获取一个 ID，否则执行 3）.

3）通过插入 $\bar{t}-1$ 个点 $\{(x_1,y_1),(x_2,y_2),\cdots,(x_{\bar{t}-1},y_{\bar{t}-1})\}$ 和 $(x^*=x,y^*=y)$ 来恢复 $\mathrm{INS}_{(\bar{t},\bar{n})}$ 的秘密值 a_0^*，若 $a_0^*=f(0)$，则计算 $\mathrm{Dec}_{\bar{k}_1}(x^*)$ 从而获取 ID 值，否则输出 $\perp$.

通过追踪算法，当某用户有不正当行为时，追踪到用户的 ID 并将其 ID 添加到相应的属性撤销列表中，一旦用户的某个属性被撤销，这个属性所对应访问结构中的矩阵某一行 x 的映射 $\rho(x)$ 便不存在，用户也就还原不了秘密 s 的值，解密不了密文，从而实现对用户的追踪与该用户的属性的撤销过程.

3　安全性证明

根据假设 2～4，采用对偶系统加密技术来实现方案的选择安全性证明，下面将简要介绍对偶系统加密技术，之后在 l-SDH 假设下证明可追踪机制的安全性.

3.1 对偶系统加密

2009 年,Waters[22] 提出对偶系统加密技术,该技术是通过一系列的模拟游戏来实现的. 先定义相关的半功能密钥和半功能密文,接着再详细描述对偶系统加密.

半功能密钥:为生成半功能密钥,首先调用 T-RA-CPABE 中 KeyGen 算法生成正常的密钥,包括 K_0、K、L_0、L、$K_{i,1}$、$K_{i,2}$;然后对用户的任意属性 i,随机选取 $X_0, X_1, X_2, X_{i,1}, X_{i,2} \in G_{p_2}$,得到 2 种半功能密钥.

1) 半功能密钥一. $K'_0 = K_0X_0$, $K' = K$, $L'_0 = L_0X_1$, $L' = LX_2$, $K'_{i,1} = K_{i,1}X_{i,1}$, $K'_{i,2} = K_{i,2}X_{i,2}$,即除了 K 每个元素都与 G_{p_2} 中的一个随机元素相乘,得到半功能密钥一:$K'_0 = g^{\alpha/(a+c)}X_0Y_0$, $K' = c$, $L'_0 = (g^t)^{1/(a+c)} \cdot X_1Y_1$, $L' = (g^t)^{a/(a+c)}X_2Y_2$, $K'_{i,1} = g^{at+\alpha^{\mathrm{ID}}\gamma_i+\tau_i r_i}X_{i,1}Y_{i,1}$, $K'_{i,2} = g^{r_i}X_{i,2}Y_{i,2}$.

当运行解密算法时,G_{p_2} 中的元素将会被消去,因此,半功能密钥一仍然能够解密半功能密文.

2) 半功能密钥二. 半功能密钥二为

$$K'_0 = g^{\alpha/(a+c)}Y_0, K' = c, L'_0 = (g^t)^{1/(a+c)}Y_1$$

$$L' = (g^t)^{a/(a+c)}Y_2, K'_{i,1} = g^{\alpha^{\mathrm{ID}}\gamma_i+\tau_i r_i}X_{i,1}Y_{i,1}, K'_{i,2} = g^{r_i}Y_{i,2}$$

为生成半功能密文,首先调用加密算法生成正常的密文,包括 C_0、C_1、C_2、$C_{x,0}$、$C_{x,1}$、$C_{x,2}$、$C_{x,3}$、$C_{x,4}$,令 X 为 G_{p_2} 子群的生成元,随机选取向量 $\boldsymbol{v} = (s, v_2, v_3, \cdots, v_k) \in Z_N^k$ 和 $\boldsymbol{u} = (c, u_2, u_3, \cdots, u_k) \in Z_N^k$,得到半功能密文 C' 为

$$C'_0 = C_0, C'_1 = C_1X^c$$

$$C'_2 = C'_1(X^c)^a, C'_{x,0} = C_{x,0}X^{\boldsymbol{A}_x\cdot\boldsymbol{u}}$$

$$C'_{x,1} = (C'_{x,0})^{\rho(x)}, C'_{x,2} = C_{x,2}X^{\left(\gamma_i+\sum_{j\in S_{\rho(x)}}\alpha^{n+1-j}\right)\boldsymbol{A}_x\cdot\boldsymbol{u}}$$

$$C'_{x,3} = C_{x,3}, C'_{x,4} = C_{x,4}$$

即

$$C'_1 = g^sX^c, C'_2 = (g^sX^c)^a, C'_{x,0} = g^{\boldsymbol{A}_x\cdot\boldsymbol{v}}X^{\boldsymbol{A}_x\cdot\boldsymbol{u}}$$

$$C'_{x,1} = (C'_{x,0})^{\rho(x)}, C'_{x,3} = C_{x,3}, C'_{x,4} = C_{x,4}$$

$$C'_{x,2} = g^{\eta_x}\left(h_{\rho(x)}\prod_{j\in S_{\rho(x)}}g_{n+1-j}\right)^{\boldsymbol{A}_x\cdot\boldsymbol{v}}X^{\left(\gamma_i+\sum_{j\in S_{\rho(x)}}\alpha^{n+1-j}\right)\boldsymbol{A}_x\cdot\boldsymbol{u}}$$

在对偶系统加密技术中,正常密钥能解密半功能密文,半功能密钥能解密正常密文,而半功能密钥却不能解密半功能密文. 值得一提的是,半功能密文和半功能密钥只适用于安全性证明过程中,在实际使用过程中并不需要.

用 $\mathrm{Game}_{\mathrm{real}}$ 表示 1.7.1 节中的真实游戏;Game_k 表示在游戏中返回给攻击者的密文和前 k 个密钥是半功能的,其他密钥是正常的;假设攻击者请求密钥 q 次(那么 $0 \leqslant k \leqslant q$),$\mathrm{Game}_0$ 与 $\mathrm{Game}_{\mathrm{real}}$ 相似,不同之处是 Game_0 中挑战者返回给攻击者的是半功能密文,所以 $\mathrm{Game}_0 = \mathrm{Game}_0^2$;$\mathrm{Game}_{\mathrm{final}}$ 类似于 Game_q,只是 $\mathrm{Game}_{\mathrm{final}}$ 中返回给攻击者的密文是对一条随机消息加密生成的半功能密文,此时攻击者的优势可忽略.

在此设 Game_k^1 表示该游戏中前 $k-1$ 次密钥请求将返回半功能密钥二,第 k 次密钥请求将返回半功能密钥一,其他的都返回正常的密钥;Game_k^2 表示该游戏中前 k 次密钥查询返回半功能密钥二,其他的返回正常的密钥.

3.2 选择安全性证明

定理 2 若 1.4 节中的假设 2 ~ 4 都成立,那么本文提出的 T-RA-CPABE 算法是选择安全的.

通过证明 $\mathrm{Game}_{\mathrm{real}}$ 与 Game_0 不可区分、Game_{k-1}^2 和 Game_k^1 不可区分、Game_k^1 和 Game_k^2 不可区分(即 $\mathrm{Game}_{\mathrm{real}}$ 与 Game_0^2 不可区分,Game_0^2 与 Game_1^1 不可区分,Game_1^1 与 Game_1^2 不可区分,Game_1^2 与 Game_2^2 不可区分,依次类推,直到 Game_q^2),最后证明 Game_q^2 与 $\mathrm{Game}_{\mathrm{final}}$ 不可区分,而在 $\mathrm{Game}_{\mathrm{final}}$ 游戏中,攻击者赢得游戏的优势可以忽略,选择安全性证明过程到此结束. 下面通过引理 1 ~ 4 的证明来证明定理 2.

引理 1 在假设 2 下,在概率多项式时间内没有攻击者能够获取一个可以区分 $\mathrm{Game}_{\mathrm{real}}$ 和 Game_0 的不可忽略的优势 ε.

证明:若在概率多项式时间内存在攻击者能以不可忽略的优势 ε 区分 $\mathrm{Game}_{\mathrm{real}}$ 和 Game_0,则可以构造一个算法 $\mathcal{B}$ 攻破假设 2. 挑战者发送 (g, Y, T) 给 $\mathcal{B}$,其中 g 是 G_{p_1} 子群的生成元,Y 是 G_{p_3} 子群的生成元,T 要么是子群 G_{p_1} 的随机元素,要么是子群 $G_{p_1p_2}$ 的随机元素.

在 Init 阶段,攻击者声明访问结构 $(\boldsymbol{A}^*_{l\times k}, \rho)$,并对任意行 $x \in \{1, 2, \cdots, l\}$ 指定属性 $\rho(x)$ 的用户撤销列表 $R^*_{\rho(x)}$.

在 Setup 阶段,令属性集合 $\omega^* = \{\rho(x)\}_{x\in[1,2,\cdots,l]}$,若 $i \notin \omega^*$,则 $h_i = g^{\gamma_i}$,否则随机选取 $\beta_i \in Z_N$,令 $S_i = U - R_i$, $\gamma_i = \beta_i - \sum_{j\in S_i}\alpha^{n+1-j}$,即

$$h_i = g^{\beta_i}\left(\prod_{j\in S_i}g_{n+1-j}\right)^{-1}$$

得到系统公钥为

$$\mathrm{PK} = \{N, g, g^a, e(g,g)^\alpha, \{f_i\}_{i\in I}, \{h_i\}_{i\in I}, \{g_j\}_{j\in U}\}$$

系统主私钥为

$$\mathrm{MSK} = \{\alpha, a, \{\tau_i, \gamma_i\}_{i\in I}, \overline{k}_1, \overline{k}_2, Y\}$$

在 Phase 1 阶段，$\mathcal{B}$ 完全拥有系统的主私钥 MSK，可模拟任意密钥.

在挑战阶段，系统随机选取 $k-1$ 个元素 $v_2', v_3', \cdots, v_k'$，得到向量 $\boldsymbol{v}' = (1, v_2', v_3', \cdots, v_k')$，对随机选择的消息 M_b 进行加密（$b \in \{0,1\}$），得到密文

$$C_0^* = M_b e(g^{\alpha^{n+1}}, T) e(g^{\alpha}, T), C_1^* = T, C_2^* = T^a$$

$$C_{x,0}^* = T^{\boldsymbol{A}_x^* \cdot \boldsymbol{v}'}, C_{x,1}^* = (C_{x,0}^*)^{\rho(x)}, C_{x,2}^* = g^{\eta_x} T^{\beta_{\rho(x)} \boldsymbol{A}_x^* \cdot \boldsymbol{v}'}$$

$$C_{x,3} = g^{s_x}, C_{x,4} = g^{\eta_x} \left(\prod_{j \in R_{\rho(x)}^*} g_{n+1-j} \right)^{s_x}$$

$$R_{\rho(x)}^* x \in \{1,2,\cdots,l\}$$

若 $T \in G_{p_1}$，C^* 是一条正常的密文，则 $\mathcal{B}$ 模拟的是 $\text{Game}_{\text{real}}$；若 $T \in G_{p_1 p_2}$，C^* 是一条半功能密文，则 $\mathcal{B}$ 模拟的是 Game_0. 因此，算法 $\mathcal{B}$ 可以利用 ε 获得一个不可忽略的优势来打破假设 2. 这与假设 2 是一个不可攻破的困难问题矛盾，故不存在这样一个攻击者，引理 1 得证.

引理 2　在假设 3 下，概率多项式时间内没有攻击者能够获取一个可以区分 Game_{k-1}^2 和 Game_k^1 的不可忽略的优势 ε.

证明：若概率多项式时间内存在攻击者能以不可忽略的优势 ε 区分 Game_{k-1}^2 和 Game_k^1，则可以构造一个算法 $\mathcal{B}$ 攻破假设 3. 挑战者发送 $(g, g^s X^{c_1}, Y, X^{c_2} Y^d, T)$ 给 $\mathcal{B}$，其中 g、X 和 Y 分别是子群 G_{p_1}、G_{p_2} 和 G_{p_3} 的生成元，T 要么是群 G 的随机元素，要么是子群 $G_{p_1 p_3}$ 的随机元素.

Setup 阶段同引理 1.

在 Phase 1 阶段，对攻击者的前 $k-1$ 次密钥请求，返回的半功能密钥二如下：

$$K_0' = g^{\alpha/(a+c)} Y_0, K' = c, L_0' = (g^t)^{1/(a+c)} Y_1$$

$$L' = (g^t)^{a/(a+c)} Y_2, K_{i,1}' = g^{at + \alpha^{\text{ID}} \gamma_i + \tau_i r_i} (X^{c_2} Y^d)^{r_i}$$

$$K_{i,2}' = g^{r_i} Y_{i,2}$$

对攻击者的第 k 次密钥请求，系统随机选取 $t' \in Z_N$、$r_i' \in Z_N$（任意属性 $i \in \omega$）及 $Y_0', Y_1', Y_2', Y_{i,1}', Y_{i,2}' \in G_{p_3}$，返回的密钥如下：

$$K_0' = g^{\alpha/(a+c)} T Y_0', K' = c$$

$$L_0' = (g^{t'})^{1/(a+c)} T Y_1', L' = (g^{t'})^{a/(a+c)} T Y_2'$$

$$K_{i,1}' = g^{at' + \alpha^{\text{ID}} \gamma_i} T^a T^{r_i' \tau_i} Y_{i,1}', K_{i,2}' = T^{r_i'} Y_{1,2}'$$

在 Challenge 阶段，随机加密明文消息 M_b（其中 $b \in \{0,1\}$），得到的密文如下：

$$C_0^* = M_b e(g^{\alpha^{n+1}}, g^s X^{c_1}) e(g^{\alpha}, g^s X^{c_1}), C_1^* = g^s X^{c_1}$$

$$C_2^* = (g^s X^{c_1})^a, C_{x,0}^* = (g^s X^{c_1})^{\boldsymbol{A}_x^* \cdot \boldsymbol{v}'}, C_{x,1}^* = (C_{x,0}^*)^{\rho(x)}$$

$$C_{x,2}^* = g^{\eta_x} (g^s X^{c_1})^{\beta_{\rho(x)} \boldsymbol{A}_x^* \cdot \boldsymbol{v}'}, C_{x,3} = g^{s_x}$$

$$C_{x,4} = g^{\eta_x} \left(\prod_{j \in R_{\rho(x)}^*} g_{n+1-j} \right)^{s_x}, R_{\rho(x)}^*, x \in \{1,2,\cdots,l\}$$

若 $T \in G$，攻击者获取的第 k 个密钥是半功能密钥一，则 $\mathcal{B}$ 模拟的是 Game_k^1；若 $T \in G_{p_1 p_3}$，攻击者获取的第 k 个密钥是正常密钥，则 $\mathcal{B}$ 模拟的是 Game_{k-1}^2. 因此，算法 $\mathcal{B}$ 可以利用 ε 获得一个不可忽略的优势来打破假设 3. 因为这与假设 3 是一个困难问题矛盾，所以不存在这样一个攻击者，引理 2 得证.

引理 3　在假设 3 下，概率多项式时间内没有攻击者能够获取一个可以区分 Game_k^1 和 Game_k^2 的不可忽略的优势 ε.

证明：若概率多项式时间内存在攻击者能以不可忽略的优势 ε 区分 Game_k^1 和 Game_k^2，则可以构造一个算法 $\mathcal{B}$ 攻破假设 3. 挑战者发送 $(g, g^s X^{c_1}, Y, X^{c_2} Y^d, T)$ 给 $\mathcal{B}$.

在 Phase 1 阶段，前 $k-1$ 次密钥请求同引理 2；对第 k 次密钥请求，$\mathcal{B}$ 随机选取 $b_i \in Z_N$，其他参数同引理 2，得到的密钥如下：

$$K_0' = g^{\alpha/(a+c)} T Y_0'$$

$$K' = c, L_0' = (g^{t'})^{1/(a+c)} T Y_1', L' = (g^{t'})^{a/(a+c)} T Y_2'$$

$$K_{i,1}' = g^{at' + \alpha^{\text{ID}} \gamma_i} T^a T^{r_i' \tau_i} Y_{i,1}' (X^{c_2} Y^d)^{b_i}$$

$$K_{i,2}' = T^{r_i'} Y_{1,2}'$$

Challenge 阶段同引理 2.

若 $T \in G$，攻击者获取的第 k 个密钥是半功能密钥一，则 $\mathcal{B}$ 模拟的是 Game_k^1；若 $T \in G_{p_1 p_3}$，攻击者获取的第 k 个密钥是半功能密钥二，则 $\mathcal{B}$ 模拟的是 Game_k^2. 因此，算法 $\mathcal{B}$ 可以利用 ε 获得一个不可忽略的优势来打破假设 3. 因为这与假设 3 是一个困难问题矛盾，所以不存在这样一个攻击者，引理 3 得证.

引理 4　在假设 4 下，概率多项式时间内没有攻击者能够获取一个区分 Game_q^2 和 $\text{Game}_{\text{final}}$ 的不可忽略的优势 ε.

证明：若概率多项式时间内存在攻击者能以不可忽略的优势区分 Game_q^2 和 $\text{Game}_{\text{final}}$，则可以构造一个算法 $\mathcal{B}$ 攻破假设 4. 挑战者发送给 $\mathcal{B}$ 的参数为

$$(g, X, g^s X^{c_1}, Y, \{g_i = g^{\alpha^i}\}_{i \in U'}, g^{\alpha^{n+1}} X^{c_2}, T)$$

在 Setup 阶段，各参数同引理 1，不同之处是当 $i \in \omega^*$ 时，$\mathcal{B}$ 不知道 γ_i 的值，即 $\mathcal{B}$ 不完全具备系统的主私钥 MSK.

在 Phase 1 阶段，对攻击者所有的密钥请求，$\mathcal{B}$ 都返回一个半功能密钥二

$$K_0' = g^{\alpha/(a+c)} Y_0, K' = c$$

$$L_0' = (g^t)^{1/(a+c)} Y_1, L' = (g^t)^{a/(a+c)} Y_2$$

对任意的属性 $i \in \omega$,分 3 种情况讨论:

1) 若 $i \notin \omega^*$,则

$$K_{i,1}' = g^{at+\alpha^{\mathrm{ID}}\gamma_i+\tau_i r_i} X_{i,1} Y_{i,1}, K_{i,2}' = g^{r_i} Y_{i,2}$$

2) 若 $i \in \omega^*$, $\mathrm{ID} \in R_i^*$,则

$$K_{i,1}' = g^{at} g_{\mathrm{ID}}^{\beta_i} \left(\prod_{j \in S_i} g_{n+1+|\mathrm{ID}-j|} \right)^{-1} g^{\tau_i r_i} X_{i,1} Y_{i,1}$$

$$K_{i,2}' = g^{r_i} Y_{i,2}$$

3) 若 $i \in \omega^*$, $\mathrm{ID} \notin R_i^*$,则

$$K_{i,1}' = g^{at} g_{\mathrm{ID}}^{\beta_i} \left(\prod_{j \in S_i, j \neq \mathrm{ID}} g_{n+1+|\mathrm{ID}-j|} \right)^{-1} (g^{\alpha^{n+1}} X^{c_2})^{-1} \cdot g^{\tau_i r_i} X_{i,1}' Y_{i,1}$$

$$K_{i,2}' = g^{r_i} Y_{i,2}$$

在 Challenge 阶段,随机加密明文消息 M_b(其中 $b \in \{0,1\}$),得到的密文如下:

$$C_0^* = M_b Te(g^\alpha, g^s X^{c_1}), C_1^* = g^s X^{c_1}$$

$$C_2^* = (g^s X^{c_1})^a, C_{x,0}^* = (g^s X^{c_1})^{A_x^* \cdot v'}, C_{x,1}^* = (C_{x,0}^*)^{\rho(x)}$$

$$C_{x,2}^* = g^{\eta_x} (g^s X^{c_1})^{\beta_{\rho(x)} A_x^* \cdot v'}, C_{x,3} = g^{s_x}$$

$$C_{x,4} = g^{\eta_x} \left(\prod_{j \in R_{\rho(x)}^*} g_{n+1-j} \right)^{s_x}, R_{\rho(x)}^*, x \in \{1,2,\cdots,l\}$$

若 $T = e(g,g)^{\alpha^{n+1} s}$, C^* 是半功能密文,则 $\mathscr{B}$ 模拟的是 Game_q^2;若 $T \in G_T$, C^* 是随机加密而得,则 $\mathscr{B}$ 模拟的是 $\mathrm{Game}_{\mathrm{final}}$. 因此,算法 $\mathscr{B}$ 可以利用 ε 获得一个不可忽略的优势来打破假设 4. 这与假设 4 是一个困难问题相悖,故不存在这样一个攻击者,引理 4 得证.

定理 2 得证.

定理 3 若 l-SDH 假设成立,那么追踪算法安全($q < l$).

证明:假设在概率多项式时间内存在攻击者在 q 次请求密钥后可以获得一个不可忽略的优势赢得 1.7.2 中的追踪游戏. 设 $l = q+1$,那么可以构造一个算法 $\mathscr{B}$,其能够以不可忽略的优势攻破 l-SDH 假设.

在合数阶双线性群 (N, G, G_T, e) 下,随机选取 $a \in Z_p^*$, $\bar{g} \in G_{p_1}$,在子群 G_{p_1} 下得到一个 l-SDH 问题的实例

$$\mathrm{INS}_{l\text{-SDH}} = (N, G, G_T, e, \bar{g}, \bar{g}^a, \bar{g}^{a^2}, \cdots, \bar{g}^{a^l}, p_1, p_2, p_3)$$

并将该实例发给 $\mathscr{B}$.

算法 $\mathscr{B}$ 的目标是输出一个满足 $\omega_s = \bar{g}^{1/(a+c_s)}$ 的元组 $(c_s, \omega_s) \in Z_{p_1} \times G_{p_1}$. 算法 $\mathscr{B}$ 首先输入

$$\mathrm{INS}_{l\text{-SDH}} = (N, G, G_T, e, \bar{g}, \bar{g}^a, \bar{g}^{a^2}, \cdots, \bar{g}^{a^l}, p_1, p_2, p_3)$$

令 $H_\theta = \bar{g}^{a^\theta}, \theta \in \{1,2,\cdots,l\}$,然后算法 $\mathscr{B}$ 输入

$$(N, G, G_T, e, H_\theta)_{\theta \in \{1,2,\cdots,l\}}$$

随后攻击者与算法 $\mathscr{B}$ 之间的交互如下.

Setup:$\mathscr{B}$ 选取 q 个不同的值 $c_1, c_2, \cdots, c_q \in Z_N^*$,得到多项式 $f(x) = \prod_{\theta=1}^{q} (x + c_\theta)$,展开多项式 $f(x)$ 后可得 $f(x) = \sum_{\theta=0}^{q} \alpha_\theta x^\theta$,其中 $\alpha_0, \alpha_1, \cdots, \alpha_q \in Z_N$ 是多项式 $f(x)$ 的系数. $\mathscr{B}$ 计算:$g \leftarrow \prod_{\theta=0}^{q} (H_\theta)^{\alpha_\theta} = \bar{g}^{f(a)} \in G_{p_1}$, $g^a \leftarrow \prod_{\theta=1}^{q+1} (H_\theta)^{\alpha_{\theta-1}} = \bar{g}^{f(a)a}$.

系统随机选取 $\alpha \in Z_N$,对任意的属性 $i \in I$,随机选取 $\tau_i, \gamma_i \in Z_N$,计算 $f_i = g^{\tau_i}, h_i = g^{\gamma_i}$,对任意 $j \in U' = \{1,2,\cdots,n,n+2,\cdots,2n\}$,计算 $g_j = g^{\alpha^j}$,得到系统的公钥参数 PK,并将其发送给攻击者

$$\mathrm{PK} = \{N, g, g^a, e(g,g)^\alpha, \{f_i\}_{i \in I}, \{h_i\}_{i \in I}, \{g_j\}_{j \in U'}\}$$

系统初始化一个 Shamir$(\bar{t}, \bar{n})$ 门限方案实例

$$\{(x_1, y_1), (x_2, y_2), \cdots, (x_{\bar{t}-1}, y_{\bar{t}-1})\}$$

并对 $f(x)$ 以及 $\bar{t}-1$ 个点保密.

Key Query:攻击者向 $\mathscr{B}$ 请求 $(\mathrm{ID}_\theta, \omega_\theta)$ 对应的解密密钥,假设这是第 θ 次请求($\theta \leqslant q$),令多项式 $f_\theta(x) = f(x)/(x+c_\theta) = \prod_{j=1, j \neq \theta}^{q} (x + c_j)$,展开后可以得到 $f_\theta(x) = \sum_{j=0}^{q-1} \beta_j x^j$,其中 β_j 是多项式 $f_\theta(x)$ 的系数. 系统的计算公式为

$$\sigma_\theta = \prod_{j=0}^{q-1} (H_j)^{\beta_j} = \bar{g}^{f_\theta(a)} = \bar{g}^{f(a)/(a+c_\theta)} = g^{1/(a+c_\theta)}$$

系统随机选取 $t \in Z_N$、$r_i \in Z_N$(任意属性 $i \in \omega$)及 Y_0, Y_1, Y_2, $Y_{i,1}$, $Y_{i,2} \in G_{p_3}$,计算

$$K_0 = (\sigma_\theta)^\alpha Y_0 = g^{\alpha/(a+c_\theta)} Y_0, K = c_\theta$$

$$L_0 = (\sigma_\theta)^t Y_1 = (g^t)^{1/(a+c_\theta)} Y_1$$

$$L = (\sigma_\theta)^{at} Y_2 = (g^t)^{a/(a+c_\theta)} Y_2$$

$$K_{i,1} = g^{at+\alpha^{\mathrm{ID}}\gamma_i+\tau_i r_i} Y_{i,1}, K_{i,2} = g^{r_i} Y_{i,2}$$

将得到的密钥 $\mathrm{SK}_{\mathrm{ID},\omega} = (K_0, K, L_0, L, K_{i,1}, K_{i,2})$ 发送给攻击者.

Key Forgery:攻击者发送一个密钥 SK_* 给 $\mathscr{B}$,上述游戏中的公钥参数和私钥与真实游戏中的一致,令 ε_A 表示攻击者赢得游戏,那么 SK_* 的形式为 $(K_0, K, L_0, L, K_{i,1}, K_{i,2})_{i \in \omega_*}$ 且满足密钥检测且 $K \notin \{c_1, c_2, \cdots, c_q\}$. 若攻击者没有赢得游戏,则系统随机选取一个元组 $(c_s, \omega_s) \in Z_{p_1} \times G_{p_1}$;若攻击者赢得

游戏，使用长除法得到多项式：$f(x)=\gamma(x)(x+K)+\gamma_{-1}$，其中$\gamma(x)=\sum_{\theta=0}^{q-1}\gamma_\theta x^\theta$，$\gamma_{-1}\in Z_N$. 因为$f(x)=\prod_{\theta=1}^{q}(x+c_\theta)$，$c_\theta\in Z_N^*$ 且 $K\notin\{c_1,c_2,\cdots,c_q\}$，$\gamma_{-1}\neq 0$，所以 $x+K$ 没有除以 $f(x)$.

系统计算 gcd (γ_{-1},N) 的值，并按如下方式得到元组$(c_s,\omega_s)\in Z_{p_1}\times G_{p_1}$：

1）若 gcd $(\gamma_{-1},N)=1$，设 $L_0=g^tL_2L_3$，其中 $t\in Z_N$，$L_2\in Z_{p_2}$，$L_3\in Z_{p_3}$ 均未知，则：$L=g^{at}L_2'L_3'$，$K_0=g^{\alpha/(a+K)}K_2K_3$，其中 $L_2,K_2\in G_{p_2}$，$L_3,K_3\in G_{p_3}$. 系统计算$(1/\gamma_{-1})\bmod N$，然后得到(c_s,ω_s)为

$$\sigma\leftarrow((K_0)^{p_2p_3})^{(p_2p_3\alpha)^{-1}\bmod p_1}=g^{1/(a+K)}=\overline{g}^{\gamma(a)}\overline{g}^{\gamma_{-1}/(a+K)}$$

$$\omega_s\leftarrow\left(\sigma\prod_{\theta=0}^{q-1}H_\theta^{-\gamma_\theta}\right)^{1/\gamma_{-1}}=\overline{g}^{1/(a+K)}\in G_{p_1}$$

$$c_s\leftarrow K\bmod p_1\in Z_{p_1}$$

2）若 gcd $(\gamma_{-1},N)\neq 1$，则输出一个随机的$(c_s,\omega_s)\in Z_{p_1}\times G_{p_1}$.

下面对算法 $\mathcal{B}$ 攻破 l-SDH 假设的优势进行评估：令 $\varepsilon_{l\text{-SDH}}(c_s,\omega_s)$ 表示(c_s,ω_s)可解决 l-SDH 问题，可通过 $e(\overline{g}^a\overline{g}^{c_s},\omega_s)=e(\overline{g},\overline{g})$ 是否成立来验证. 若算法 $\mathcal{B}$ 随机选取(c_s,ω_s)，则 $\varepsilon_{l\text{-SDH}}(c_s,\omega_s)$ 发生的概率可以忽略；只有当$(\text{Awin}\wedge\gcd(\gamma_{-1},N)=1)$满足 $e(\overline{g}^a\overline{g}^{c_s},\omega_s)=e(\overline{g},\overline{g})$ 的概率为 1 时，算法 $\mathcal{B}$ 才输出 (c_s,ω_s). 算法 $\mathcal{B}$ 解决 l-SDH 问题的概率为

$$\begin{aligned}&\Pr[\varepsilon_{l\text{-SDH}}(c_s,\omega_s)]=\\&\Pr[\varepsilon_{l\text{-SDH}}(c_s,\omega_s)\mid\overline{\text{Awin}}]\Pr[\overline{\text{Awin}}]+\\&\Pr[\varepsilon_{l\text{-SDH}}(c_s,\omega_s)\mid(\overline{\text{Awin}}\wedge\gcd(\gamma_{-1},N)\neq 1)]\cdot\\&\Pr[\overline{\text{Awin}}\wedge\gcd(\gamma_{-1},N)\neq 1]+\\&\Pr[\varepsilon_{l\text{-SDH}}(c_s,\omega_s)\mid(\overline{\text{Awin}}\wedge\gcd(\gamma_{-1},N)=1)]\cdot\\&\Pr[\overline{\text{Awin}}\wedge\gcd(\gamma_{-1},N)=1]=\varepsilon/2\end{aligned}$$

算法 $\mathcal{B}$ 能够以不可忽略的优势攻破 l-SDH 假设，这便与 l-SDH 难题不可解矛盾，因此，定理 3 得证.

由定理 2 和定理 3 的证明可以得出，本文中所提出的 T-RA-CPABE 方案是选择安全的.

4　效率分析

表 1 和表 2 分别给出了本方案的功能特征、性能特征[23-24]与其他文献的对比. 本文在功能上既实现了追踪可疑用户的 ID，也实现了对可疑用户进行属性撤销. 在性能上，虽然与文献[5]相比本文内存消耗较大，但这是在实现了细粒度属性撤销的情况下所带来的，且避免了使用标识表来存储用户 ID 所带来的弊端；而对同样实现细粒度属性撤销的文献[9]来说，本文不仅增加了追踪用户的功能，而且提高了方案的性能，大大减少了公钥参数的数量，这在拥有庞大用户群的系统中有很大优势.

表 1　功能特征比较

Table 1　Functional comparison

文章	追踪用户	撤销粒度
本文	√	属性撤销
文献[5]	√	×
文献[9]	×	属性撤销
文献[12]	×	属性层面的用户撤销
文献[13]	√	用户撤销

表 2　性能特征比较

Table 2　Performance comparison

文章	PK	SK	C	P
本文	$2\lvert m\rvert+\lvert n\rvert+3$	$2\lvert m\rvert+4$	$5l+3$	$5\lvert I\rvert$
文献[5]	$\lvert m\rvert+5$	$\lvert m\rvert+4$	$2l+3$	$2\lvert I\rvert+3$
文献[9]	$2\lvert m\rvert+2\lvert n\rvert+3$	$2\lvert m\rvert+1$	$5l+2$	$5\lvert I\rvert$

注：PK 为系统公钥参数；SK 为私钥的长度；C 为密文长度；P 为解密过程中配对次数；$\lvert m\rvert$ 为系统中属性数量；$\lvert n\rvert$ 为系统中用户的数量；l 为访问结构大小；$\lvert I\rvert$ 为解密密钥上满足一个密文访问结构的属性个数.

5　结论

1）本文在 CP-ABE 密码体制中实现了一种可以追踪并撤销用户属性的方案，在追踪算法中通过使用 Shamir 门限方案避免了使用标识表等方式来存放用户 ID 所带来的弊端，减轻了存储压力.

2）将追踪算法同细粒度直接撤销用户属性的方法结合起来，从而实现了一种可以追踪到用户并通过将其属性集合中的属性添加到撤销列表中的方式实现用户权限的变更.

参考文献：

[1] SETHIA D, SINGH S, SINGHAL V. ABE based raspberry Pi secure health sensor (SHS) [C] // Advances in Ubiquitous Networking 2. Singapore: Springer, 2017: 599-610.

[2] SABITHA S, RAJASREE M S. An efficient framework for verifiable access control based dynamic data updates in public cloud [C] // Distributed Computing and Internet Technology. Cham: Springer, 2017: 147-158.

[3] RUJ S, STOJMENOVIC M, NAYAK A. Decentralized access control with anonymous authentication of data stored in clouds[J]. IEEE Transactions on Parallel & Distributed Systems, 2014, 25(2): 384-394.

[4] KIM J, NEPAL S. A cryptographically enforced access control with a flexible user revocation on untrusted cloud storage[J]. Data Science & Engineering, 2016, 1(3): 149-160.

[5] 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.

[6] NING J T, CAO Z F, DONG X L, et al. Large universe ciphertext-policy attribute-based encryption with white-box traceability[C]// Computer Security-ESORICS. Cham: Springer, 2014: 55- 72.

[7] LIU Z, CAO Z, 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. New York: ACM Press, 2013: 475-486.

[8] YU S C, WANG C, REN K, et al. Attribute based data sharing with attribute revocation[C]// Proceedings of the ASIAN ACM Conference on Computer and Communications Security (ASIACCS 2010). New York: ACM Press, 2010: 262-270.

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

[10] 马海英, 曾国荪. 可追踪并撤销叛徒的属性基加密方案[J]. 计算机学报, 2012, 35(9): 1845-1855.
MA H Y, ZENG G S. An attribute based encryption scheme for traitor tracing and revocation together[J]. Chinese Journal of Computers, 2012, 35(9): 1845-1855. (in Chinese)

[11] LI Y, ZHU J, WANG X, et al. Optimized ciphertext-policy attribute-based encryption with efficient revocation[J]. International Journal of Security & Its Applications, 2013, 7(4): 281-287.

[12] 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, 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, 2015, 42(6): 6-10. (in Chinese)

[13] LIU Z, WONG D S. Practical ciphertext-policy attribute-based encryption: traitor tracing, revocation, and large universe[C]// Applied Cryptography and Network Security. Cham: Springer, 2015, 59(7): 127-146.

[14] BONEH D, GOH E, NISSIM K. Evaluating 2-DNF formulas on ciphertexts[C]// Proceedings of Tcc'05 LNCS. Berlin: Springer, 2005: 325-341.

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

[16] BONEH D, BOYEN X. Short signatures without random oracles[C]// Advances in Cryptology-EUROCRYPT 2004. Berlin: Springer, 2004: 56-73.

[17] KATZ J, SAHAI A, WATERS B. Predicate encryption supporting disjunctions, polynomial equations, and inner products[C]// Advances in Cryptology-EUROCRYPT 2008. Berlin: Springer, 2008: 146-162.

[18] LEWKO A, OKAMOTA T, SAHAI A, et al. Fully secure functional encryption: attribute-based encryption and (hierarchical) inner product encryption[C]// Advances in Cryptolog-EUROCRYPT 2010. Berlin: Springer, 2010: 62-91.

[19] SHAMIR A. How to share a secret[J]. Communications of the ACM, 1979, 22(11): 612- 613.

[20] GOLDWASSER S, MICALI S. Probabilistic encryption[J]. Journal of Computer & System Sciences, 1984, 28(2): 270-299.

[21] NING J T, CAO Z F, DONG X L, et al. White-box traceable CP-ABE for cloud storage service: how to catch people leaking their access credentials effectively[J]. IEEE Transactions on Dependable & Secure Computing, 2018, 15(5): 883-897.

[22] WATERS B. Dual system encryption: realizing fully secure IBE and HIBE under simple assumptions[C]// Advances in Cryptology-CRYPTO 2009. Berlin: Springer, 2009: 619-636.

[23] NING J T, DONG X L, CAO Z F, et al. White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(6): 1274-1288.

[24] 宁建廷. 可追踪与解密外包属性基加密研究[D]. 上海: 上海交通大学, 2016.
NING J T. The research on traceable and decryption outsourced attribute-based encryption[D]. Shanghai: Shanghai Jiaotong University, 2016. (in Chinese)

(责任编辑 梁 洁)