支持恶意用户追踪的属性基云数据共享方案

宁建廷[1),4)] 黄欣沂[1)] 魏立斐[2)] 马金花[1)] 荣 静[3)]

[1)](福建师范大学计算机与网络空间安全学院 福州 350117)

[2)](上海海洋大学信息学院 上海 201306)

[3)](扬州大学广陵学院 江苏 扬州 225009)

[4)](中国科学院信息工程研究所信息安全国家重点实验室 北京 100093)

摘 要 云存储服务为互联网用户提供了更加灵活、可扩展的外包数据访问,获得学术界和工业界的关注.为防止用户数据被泄露到开放的网络中,数据所有者往往选择先将数据加密再上传到云存储服务器.与传统加密方式相比,属性基加密为云端加密数据提供更细粒度的数据访问控制机制,可实现"一对多"的访问策略模式.然而,传统属性基加密系统不具备同时"追踪"泄露解密密钥以及制造解密黑盒恶意用户的能力.本文重点研究在云存储服务的数据共享机制中部署密文策略属性基加密时出现的解密权限泄露问题.本文将白盒可追踪性和黑盒可追踪性作为一个整体集成到传统的密文策略属性基加密中,并提供两种追踪机制:白盒追踪机制和黑盒追踪机制.若泄露的解密权限为解密密钥,则可利用白盒追踪;若泄露的解密权限为解密设备,则需要利用黑盒追踪.本文所提出系统的主要特点包括:(1)云共享数据的授权访问.数据所有者的文件被加密并存储到云服务器上,满足指定访问策略的授权用户才能访问文件;(2)白盒可追踪性.若恶意内部人员故意泄露其解密密钥,则可基于白盒追踪机制执行追踪;(3)黑盒可追踪性.若泄露解密设备并被发现,则可通过黑盒追踪机制捕获共同构造该设备的恶意内部人员;(4)公开追踪性.白盒追踪算法是公开的,任何人都能运行该追踪算法而无需额外的秘密信息;(5)轻量级追踪代价.在白/黑盒追踪机制中,系统不需要额外增加与用户秘密信息相关的公共参数,且整个系统的公共参数与用户数量无关,这使得系统更为实用.本文所构造的方案既保证了对云加密共享数据的细粒度访问控制,又保证了对"泄露"解密权限的可追踪性.最后,安全分析和性能评估验证了本文所提出方案的安全性和效率.

关键词 云数据共享;密文策略属性基加密;可追踪性;解密权限滥用;数据访问控制

中图法分类号 TP309 **DOI 号** 10.11897/SP.J.1016.2022.01431

Tracing Malicious Insider in Attribute-Based Cloud Data Sharing

NING Jian-Ting[1),4)] HUANG Xin-Yi[1)] WEI Li-Fei[2)] MA Jin-Hua[1)] RONG Jing[3)]

[1)](*College of Computer and Cyber Security, Fujian Normal University, Fuzhou* 350117)

[2)](*Information School, Shanghai Ocean University, Shanghai* 201306)

[3)](*Guangling College, Yangzhou University, Yangzhou, Jiangsu* 225009)

[4)](*State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing* 100093)

Abstract Providing more flexibility and scalability on outsourced data access for Internet users, cloud-based storage service has drawn increasing interest from academia and industry. To protect user data from being leaked to an open network, end-to-end encryption technique may be leveraged. Compared to traditional encryption (e.g., AES and RSA), a new cryptographic primitive called attribute-based encryption offers more fine-grained level on data sharing (one-to-many mode via access policy) which may be considered as a highly promising cloud-based data access

收稿日期:2020-12-30;在线发布日期:2021-10-08.本课题得到国家自然科学基金项目(62032005,61972094,61872087,61972241)、福建省自然科学基金项目(2020J02016)和福建省科协第二届青年人才托举工程资助.**宁建廷**,博士,教授,主要研究领域为密码学与信息安全.E-mail:jtning@fjnu.edu.cn.**黄欣沂**(通信作者),博士,教授,主要研究领域为密码学和信息安全.E-mail:xyhuang@fjnu.edu.cn.**魏立斐**,博士,副教授,主要研究方向为应用密码学和数据安全.**马金花**,博士,讲师,主要研究方向为密码学和信息安全.**荣 静**,硕士,讲师,主要研究方向为密码学.

control. However, traditional ciphertext-policy attribute-based encryption systems may not "catch" any malicious insiders who are keen to "leak" decryption privilege to an unauthorized party. In this paper, we investigate the decryption privilege leakage problem when deploying ciphertext-policy attribute-based encryption in data sharing mechanism for cloud storage service. We integrate white-box and black-box traceability as a whole into conventional ciphertext-policy attribute-based encryption, and two types of tracing (i.e., white-box and black-box tracing mechanisms) are provided in our proposed system. In particular, if the leaked decryption privilege is advertised to be a decryption key, one may make use of the white-box tracing; but if the decryption privilege is in the form of a decryption device, the black-box tracing shall be leveraged. The main features of our proposed system include (1) Authorized access over shared data in the cloud. A data owner's files are encrypted and stored in a cloud server, and only authorized cloud users satisfying a specified access policy can gain access to the files; (2) White-box Traceability. If a malicious insider deliberately leaks his decryption key and the key is found, the traitor who leaks the key will be traced by the white-box tracing mechanism; (3) Black-box Traceability. If a decryption device is leaked and found, the malicious insiders who jointly build the device can be caught with the black-box tracing mechanism; (4) Public tracing. The white-box tracing algorithm is public in the sense that anyone can run it without the need of "extra" secret information; and (5) Almost-no-storage for traceability. The proposed system does not need additional public parameters which are related to each of the user's secret information in white/black-box traceability. The public parameters of the whole system are independent of the number of users, which makes the system more practical. Our novel construction guarantees fine-grained access control over cloud-based encrypted data but also the traceability of "leaked" decryption privilege. The security analysis and performance evaluation outline the security and efficiency of the construction.

Keywords cloud data sharing; ciphertext-policy attribute-based encryption; traceability; misuse of decryption privilege; data access control.

1 引 言

云存储服务凭借其高可靠性、高通用性、高可扩展性以及大容量存储等为数据的管理提供了更加便捷、高效的方法，其正逐渐取代传统的数据外包技术[1]. 利用云存储服务，互联网用户将数据外包到云端之后，可以随时随地通过网络访问数据，且无需承担本地数据管理的负担[2-3]. 同时，根据数据所有者指定的数据共享策略，授权的云用户可以共享数据.

然而，在提供灵活的数据访问和数据共享机制的同时，云存储服务中外包数据的机密性也面临着很大的安全威胁. 数据所有者可能会担心其外包数据被泄露到开放网络中、或者被非授权用户访问，该安全隐患是云存储服务得以普及的重要障碍之一.

属性基加密(Attribute-Based Encryption，ABE)能够实现"模糊描述"的数据共享[4-5]系统. ABE 将传统公钥加密(Public Key Encryption，PKE)技术扩展到更常见的情景下，通过指定访问策略使得数据在一组接收者之间共享. 具体而言，数据所有者可以指定一个访问策略(或属性集)对数据进行加密，被授权的接收者可以使用属性集(或访问策略)的关联密钥访问数据. 然而，ABE 所提供的这种"一对多"的固有特性将会阻碍其在实际应用中的广泛推广：这种"一对多"的数据共享模式无法提供一种可靠的方法来识别将权限泄露给非授权用户的恶意内部人员. 例如，系统内部人员可能会出售其解密权以牟取私利.

在常规的公钥加密系统中，如果系统内部人员(具有一对公/私钥)将其私钥泄露给其他人，则可以通过使用与证书相关的公钥来验证私钥，从而轻松地追踪到该内部人员. 类似地，在基于身份的加密中，可以通过嵌入私钥中的身份信息来捕获恶意内部人员. 在对称加密中，可以通过生成密钥的设备来识别泄露的密钥. 但是对于 ABE 而言，很难分辨谁

是恶意的，因为泄露的密钥可能对应一组(共享相同的解密权限)具有不同描述的系统内部人员.

可通过以下两个场景来描述上述情况：

假设患有心脏病的患者当前的就诊记录在社区医院 A，但由于其住址变更等原因需要到社区医院 B 就诊. 社区医院 A 的工作人员很可能并不了解社区医院 B 的运营模式，因为一个新患者可能不会被立即分配给某个医生. 此时医院 A 的护理护士可以设定访问策略(例如："'老年服务' AND '心脏病科' OR '药房' OR '紧急情况'")对患者的电子健康就诊记录进行加密，然后将加密后的信息上传到云服务器. 经过授权的社区医院 B 的工作人员就可以访问患者的就诊记录，从而了解患者的具体情况，这样便保证了数据的机密性. 假设至少存在一名医生(拥有属性{老年服务，心脏病科，主任})和一名护士(拥有属性{紧急情况，实习生})与策略匹配，他们便可以使用各自的密钥去访问这名患者的健康记录. 但如果患者的健康记录被泄露给某医药公司，便很难追踪是谁泄露了该记录.

另一种类似情况发生在付费电视应用程序场景中. 例如 Netflix、腾讯视频、爱奇艺等付费电视服务提供商可以按照"'15 岁以上' AND '付费用户' AND '仅限英国境内'"的访问策略对某些电视频道和电影节目进行加密，然后将其上传到云端. 满足访问策略的付费用户可使用其密钥从云端观看节目. 然而，某恶意用户(具有属性{25 岁，伦敦，付费用户})可能会与没有订阅该服务的朋友共享自己的密钥，或者甚至将密钥"隐藏"在设备中，并在亚马逊上销售该设备. 由于满足访问策略的用户是一个群体，难以发现或追踪到确切的恶意用户.

基于上述两个云数据共享场景，在 ABE 系统中设计识别恶意用户机制是极其必要的，从而有效威慑解密权限滥用现象. 因此，为解决在云存储服务的数据共享机制中部署密文策略(Ciphertext Policy)属性基加密(CP-ABE)这一密码体制时可能面临的解密权限泄露问题，本文提出同时支持白盒追踪机制和黑盒追踪机制的加密数据共享系统. 如果泄露的解密权限为解密密钥，则采用白盒追踪；如果泄露的解密权限以解密设备的形式存在，则采用黑盒追踪. 一般而言，白盒追踪相较于黑盒追踪更为高效.

本文提出的系统具有以下主要特点：

(1) 云共享数据的授权访问. 数据所有者将文件加密并存储在云服务器中，满足访问策略的授权云用户才可以访问这些文件.

(2) 白盒可追踪性. 如果恶意用户故意泄露其解密密钥，通过运行白盒追踪机制，可以准确地追踪到该恶意用户.

(3) 黑盒可追踪性. 如果解密设备被泄露，通过运行黑盒追踪机制，可以准确地追踪到构建该设备的恶意用户.

(4) 公开追踪性. 白盒追踪算法是公开的，谁都可以运行且无需额外的秘密信息.

(5) 高效性. 系统不需要额外增加与用户秘密信息相关的公共参数，且整个系统的公开参数与用户数量无关，这使得系统更加实用.

本文第 2 节介绍 ABE 相关工作；第 3 节引入一些必要的预备知识；第 4 节描述系统架构和安全模型；第 5 节给出云存储服务中可追踪数据共享机制的定义及相应的安全模型；第 6 节为该系统的具体结构和安全性分析；第 7 节为系统的性能评估，包括理论分析部分和实验分析部分；第 8 节总结全文并展望今后的工作.

2　相关工作

2005 年，Sahai 等人[4]提出 ABE 的概念. 2006 年，Goyal 等人[5]形式化定义了 ABE 的两个分支：CP-ABE 和 KP-ABE(Key Policy). CP-ABE 机制中的访问策略与密文关联、用户密钥与属性集关联；KP-ABE 机制中用户密钥与访问策略关联，属性集则与密文关联. 随后，为满足不同方向的应用需求，许多 CP-ABE 和 KP-ABE 系统被陆续提出，包括具有完全安全性的新证明技术[6-9]、去中心化的多授权中心技术[10-13]、安全外包计算[14-17]以及其他应用[18-21]等. 同时，ABE 还被用于为基于云的应用提供数据共享服务[22-27]. 然而，这些方案尚未为 ABE 系统面临的解密权限泄露问题提供有效的解决方法.

为防止用户合谋以非法共享密钥，Li 等人[28]在 2009 年提出可问责 CP-ABE 概念. 2011 年，Li 等人[29]构造出可问责多授权 CP-ABE 方案. 为追踪泄露解密密钥的内部人员，一系列白盒可追踪 CP-ABE 方案被提出[30-38]；为追踪泄露解密设备的内部人员，文献[39-43]提出了几种黑盒可追踪 CP-ABE 系统. 但上述系统均未同时考虑白盒和黑盒追踪.

为解决上述问题(即同时实现 CP-ABE 系统的白盒和黑盒可追踪性)，一个简单的实现方法是将白盒可追踪的构造与黑盒可追踪的构造直接结合，但该方法可能导致系统参数激增而降低效率. 此外，利

用该方法构造的系统是否可证明安全犹未可知.针对该问题,本文将给出一个安全有效的解决方案.

3 预备知识

3.1 符号介绍

令 $\boldsymbol{Z}_p^{l\times n}$ 表示 Z_p 上的 $l\times n$ 阶矩阵集合.N^* 表示正整数集合.$[q]$ 表示 $\{1,2,\cdots,q\}$,$[q_1,q_2]$ 表示 $\{q_1,q_1+1,\cdots,q_2\}$,其中 $q_1<q_2$,且 q,q_1,q_2 是正整数.令 $\boldsymbol{y}=(y_1,y_2,\cdots,y_n)$ 表示行向量,$\boldsymbol{y}^{\perp}=(y_1,y_2,\cdots,y_n)^{\perp}$ 表示列向量.y_i 表示向量 $\boldsymbol{y}$ 的第 i 个元素.

3.2 访问结构

定义 1. 访问结构(Access Structure)[44].令 S 表示属性集合,S 的非空子集构成集合 $A\subseteq 2^S$,称 A 是 S 上的一个访问结构.如果任意两个集合 B、C 满足

若 $B\in A$ 且 $B\subseteq C$,那么 $C\in A$.

则称集合 A 为单调集合,此时 A 是一个单调访问结构.包含在 A 中的集合即授权集合,不包含在 A 中的集合即非授权集合.本文只涉及单调访问结构.

3.3 线性秘密分享方案

定义 2. 线性秘密分享方案(Linear Secret Sharing Schemes)[44].若 S 是一个属性集,Z_p 为秘密域(p 是素数),则实现 S 上访问结构的一个秘密分享方案 Π 是线性秘密分享方案须满足:

(1)每个属性分享得到的秘密($s\in Z_p$)碎片构成 Z_p 上的一个向量;

(2)每个访问结构 A 都有一个 l 行 n 列矩阵 $\boldsymbol{M}$,称为秘密分享方案 Π 的秘密生成矩阵.对于每个 $i=1,\cdots,l$,定义一个函数 ρ,将 $\boldsymbol{M}$ 的第 i 行映射到 S 中的一个属性 $\rho(i)$ 上.考虑列向量 $\boldsymbol{v}=(s,r_2,\cdots,r_n)$,其中 $s\in Z_p$ 是待分享的秘密,$r_2,\cdots,r_n\in Z_p$ 是随机值.$\boldsymbol{Mv}\in\boldsymbol{Z}_p^{l\times 1}$ 是秘密 s 根据 Π 得到的 l 个碎片组成的向量.碎片 $(\boldsymbol{Mv})_j$ 分给属性 $\rho(j)$,其中 $j\in[l]$.

据文献[44],每个线性秘密分享方案都能进行线性秘密恢复:假设 Π 是 A 上的一个线性秘密分享方案,$S'\in A$ 是一个授权属性集,令 $I\subset\{1,\cdots,l\}$ 为 $I=\{i\in[l]\wedge\rho(i)\in S'\}$.则存在常量集合 $\{\omega_i\in Z_p\}_{i\in I}$,从而对于秘密 s 的任意有效分享 $\{\lambda_i=(\boldsymbol{Mv})_i\}_{i\in I}$,有 $\sum_{i\in I}\omega_i\lambda_i=s$.常量 $\{\omega_i\}_{i\in I}$ 在 $\boldsymbol{M}$ 的大小的多项式时间内可以计算出来.对未授权集合 S'',不存在这样的常量集合.本文中使用 $(\boldsymbol{M},\rho)$ 来表示访问策略.

3.4 素数阶双线性群

G 和 G_T 是 p 阶循环群(p 是素数).g 为群 G 的一个生成元,$e:G\times G\to G_T$ 是一个双线性映射.双线性映射 e 满足:

(1)双线性.对于 $\forall g_1,g_2\in G,a,b\in Z_p$,有 $e(g_1^a,g_2^b)=e(g_1,g_2)^{ab}$.

(2)非退化性.$e(g,g)\neq 1$.1 是 G_T 的单位元.

若 G 上的群操作和双线性映射 $e:G\times G\to G_T$ 都可以有效计算,则 G 为一个素数阶双线性群.

3.5 抗共谋指纹编码方案

定义 3. 抗共谋指纹编码方案(Collusion Resistant Fingerprinting Code Scheme,FC)[45].对一个码字 $f\in\{0,1\}^d$,写成 $f_1\cdots f_i\cdots f_d$ 形式,其中 f_i 是 f 中的第 i 个位,$i\in[d]$.令 $D=\{f^{(1)},\cdots,f^{(n)}\}$ 是一组码字的集合.定义:如果对于一个码字 $f\in\{0,1\}^d$ 的所有 $i\in[d]$,都存在一个 $j\in[n]$,使得 $f_i=f_i^{(j)}$,则这个码字 f 对 D 是可行的.对于一组码字 $D\subseteq\{0,1\}^d$ 定义可行集合 $F(D)$,$F(D)$ 是所有对 D 可行的码字的集合.一个抗共谋指纹编码方案包含以下两个算法:

$FC.Gen(\varepsilon,\mathcal{N},d)\to(\{f^{(1)},\cdots,f^{(\mathcal{N})}\},rk)$:生成算法.输入安全参数 $\varepsilon\in\{0,1\}$、系统中的用户数 $\mathcal{N}$,码字长度 d,输出一组码字 $D=\{f^{(1)},\cdots,f^{(\mathcal{N})}\}$ 和恢复密钥 rk.

$FC.Recover(f^*,rk)\to N'$:恢复算法.输入一个码字 f^* 和恢复密钥 rk,其中 $f^*\in\{0,1\}^d$.输出一个子集 $N'\subset[\mathcal{N}]$,N' 是被指控创建了码字 f^* 的用户集合.

抗共谋指纹编码方案的安全性通过挑战者 $\mathcal{C}$ 和敌手 $\mathcal{A}$ 之间的博弈游戏来描述:令 N 是 $\mathcal{N}$ 的一个子集.

第一阶段.挑战者 $\mathcal{C}$ 调用 $FC.Gen(\varepsilon,\mathcal{N},d)$ 得到 $(\{f^{(1)},\cdots,f^{(\mathcal{N})}\},rk)$ 并发送 $D'=\{f^{(i)}\}_{\forall i\in N}$ 给敌手 $\mathcal{A}$.

第二阶段.敌手 $\mathcal{A}$ 给出一个码字 $f^*\in F(D')$.

如果算法 $FC.Recover(f^*,rk)$ 的输出 $N'\not\subset[N]$ 或者输出为空时,则认为 $\mathcal{A}$ 赢得了游戏.令 $Adv_{\mathcal{A}}^{FC}$ 表示 $\mathcal{A}$ 赢得上述游戏的优势.如果对所有敌手、所有 $\mathcal{N}>0$、$\varepsilon\in(0,1)$、所有子集 $N\subseteq[\mathcal{N}]$ 都有 $Adv_{\mathcal{A}}^{FC}<\varepsilon$ 成立,则称指纹编码方案是完全抗共谋的.如果每一个子集 $N\subseteq[\mathcal{N}]$ 的大小不超过 t,认为指纹编码方案是 t-抗共谋的.

3.6 复杂性假定

假定 1. 决策 q-并行 BDHE 假定(Decisional q-Parallel Bilinear Diffie-Hellman Exponent assumption[46],决策性 q-并行 BDHE).定义如下:首先根

据安全参数取一个 p 阶群 G（p 是素数），然后随机选取一个群元素 $g\in G$ 以及 $q+2$ 个随机指数 $c,s,b_1,b_2,\cdots,b_q\in Z_p$. 如果给敌手一个群描述 (p,G,G_T,e) 和一个包含下列项的 z

$$g,g^s$$
$$g^c,\cdots,g^{c^q},g^{c^{q+2}},\cdots,g^{c^{2q}}$$
$$g^{s\cdot b_j},g^{c/b_j},\cdots,g^{(c^q/b_j)},g^{(c^{q+2}/b_j)},\cdots,g^{(c^{2q}/b_j)},\ \forall 1\le j\le q$$
$$g^{c\cdot s\cdot b_k/b_j},\cdots,g^{(c^q\cdot s\cdot b_k/b_j)},\ \forall 1\le j,k\le q,k\ne j$$

敌手很难区分 $e(g,g)^{sc^{q+1}}\in G_T$ 和 $R\in G_T$，R 是从 G_T 中随机选取的元素.

若 $|Pr[\mathcal{A}(z,e(g,g)^{sc^{q+1}})=0]-Pr[\mathcal{A}(z,R)=0]|\ge\varepsilon$，则输出 $\beta\in(0,1)$ 的算法 $\mathcal{A}$ 在求解上述假定时，优势为 ε.

定义 4.　若在概率多项式时间内，没有算法能以一个不可忽略的优势解决决策 q-并行 BDHE 问题，则决策 q-并行 BDHE 假定成立.

4　系统架构及安全模型

4.1　系统架构

系统中有 4 个参与实体：云服务器（Cloud Server，CS）、授权机构（Authority，AU）、数据所有者（Data Owner）、数据用户（Data User）. 系统架构如图 1 所示. AU 生成系统公开参数与秘密参数，并为数据用户颁发解密密钥，其他实体信任 AU 来执行（黑盒）追踪过程. 数据所有者根据一些预定义的访问策略对其数据进行加密，并进一步将加密的数据外包给 CS. 被授权的数据用户可以解密存储在 CS 中的共享（加密）数据以访问明文. 若数据用户的属性集合满足在共享（加密）数据上定义的访问策略，则为授权用户. 若发现解密设备泄露，AU 可以运行黑盒追踪算法找出制造该设备的恶意用户. 若解密密钥泄露，任何用户（包括 AU）都可以运行白盒追踪算法找到泄露密钥的用户.

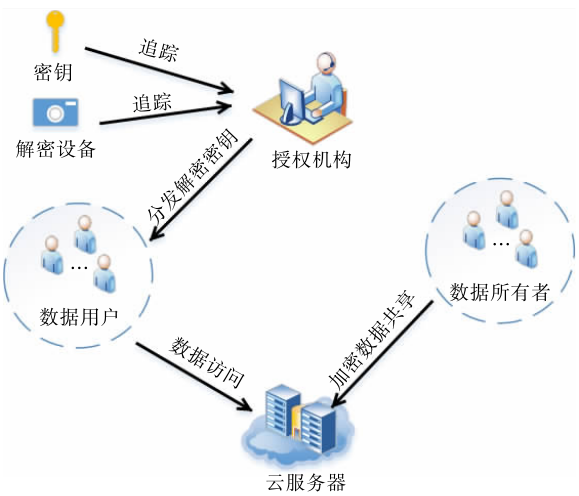

图 1　系统架构

4.2　安全模型与设计目标

完全受信任的 AU 生成系统参数并诚信地执行追踪过程. CS 是诚信且好奇的，其可能基于外包（加密）数据尝试推断出更多关于数据所有者的隐私信息，但仍会诚信地存储共享数据. 数据所有者需要对其外包的数据进行加密，以保护其敏感数据并防止非授权访问. 授权数据用户可能会（为牟利）故意泄露其解密权限，例如将其解密密钥出售给第三方，甚至构建解密设备（与其他授权数据用户一起）. 实际上，泄露的解密密钥或解密设备很可能吸引黑市的潜在买家. 为简单起见，假设数据所有者可以确定其外包数据被异常访问（例如，通过异常 IP 地址、异常访问时间等），并且追踪过程可以进一步访问泄露的解密密钥或解密设备（例如从 eBay、Twitter 等）.

本文设计的目标是为云存储服务中的数据共享提供一个可追踪的细粒度访问控制系统，该系统满足以下要求：

（1）安全保证：① 保护数据所有者数据的机密性；② 提供所泄露的解密密钥和解密设备的可追踪性.

（2）功能保障：提供对外包（加密）数据的细粒度访问控制.

（3）计算成本效益：将用于追踪的计算成本最小化.

（4）追踪过程的系统参数尽可能少：缩短识别系统恶意用户的系统参数.

5　云存储服务中可追踪数据共享机制

5.1　用于数据共享的可追踪 CP-ABE 系统的定义

用于数据共享的可追踪 CP-ABE 系统能够追踪到泄露解密密钥或者非法制造解密设备的用户. 本文扩展了传统的（不可追踪）CP-ABE 系统，通过给每个系统用户分配唯一的标识和码字来识别用户，并进一步设计 WTrace 和 BTrace 算法，分别实现白盒追踪机制和黑盒追踪机制. 依据文献[46]中 CP-ABE 系统符号，用于数据共享的可追踪 CP-ABE 系统归纳为以下六个算法：

$Setup(\lambda,U,\mathcal{N})\to(pp,msk)$. 由 AU 执行 Setup 算法建立系统. 该算法输入一个安全参数值 λ，属

性空间描述 U 以及系统中的用户数量 $\mathcal{N}$，输出公开参数 pp、主密钥 msk. 该算法为每个用户分配一个唯一元组(标识，码字)和一个属性集 S. 唯一元组(标识，码字)将被作为用于标识用户的秘密信息.

$KeyGen(msk,S,f)\rightarrow sk$. 由 AU 执行 KeyGen 算法生成用户的密钥. 该算法输入主密钥 msk，用户的属性集合 S 以及码字 f，输出用户密钥 sk. 假设 S 隐式地包含在 sk 中.

$Encrypt(pp,A,m)\rightarrow ct$. 由数据所有者执行 Encrypt 算法对消息进行加密. 该算法输入公开参数 pp，(属性空间 U 上的)访问结构 A 和消息 m，输出所得密文 ct.

$Decrypt(sk,ct)\rightarrow m/\perp$. 由数据用户执行 Decrypt 算法对密文进行解密. 该算法输入解密密钥 sk 和密文 ct. 如果 sk 中的属性集合满足 ct 中的访问结构，则输出消息 m，否则输出 $\perp$.

$WTrace(pp,sk)\rightarrow f/\perp$. 任意获得密钥 sk 的用户或者机构都可运行该白盒追踪算法 WTrace. 该算法输入公开参数 pp 和解密密钥 sk，输出与 sk 对应的码字 f 或者输出 $\perp$，其中码字 f 用以识别泄露密钥 sk 的恶意用户.

$BTrace_{\mathcal{D}}(pp,A_{\mathcal{D}})\rightarrow N'/\perp$. 该算法为黑盒追踪算法 BTrace，$\mathcal{D}$ 是泄露的解密设备. 该算法输入公开参数 pp 和 $\mathcal{D}$ 所声明的访问策略 $A_{\mathcal{D}}$. 这是一个与 $\mathcal{D}$ 交互的预言算法，其输出(使用其密钥)创建解密设备 $\mathcal{D}$ 的恶意用户集合 N'.

5.2 云存储服务中可追踪数据共享系统的定义

云存储服务中可追踪数据共享系统(Traceable Data Sharing System for Cloud Storage Service，TDS)是基于第 5.1 节中定义的数据共享可追踪 CP-ABE 系统的加密数据访问控制系统. TDS 系统提供了对外包(加密)数据的细粒度访问控制. 此外，一旦解密权(以解密密钥或解密设备的形式)被泄露，其可以追踪到密钥的所有者或构建该设备的恶意用户. 细粒度访问控制机制、白盒追踪机制和黑盒追踪机制分别如图 2、图 3 和图 4 所示.

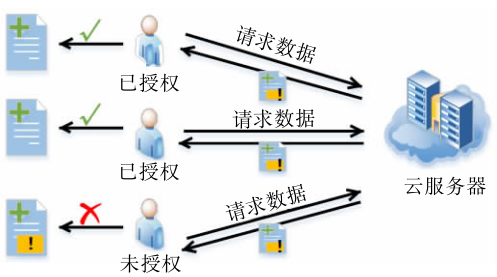

图 2 细粒度访问控制机制

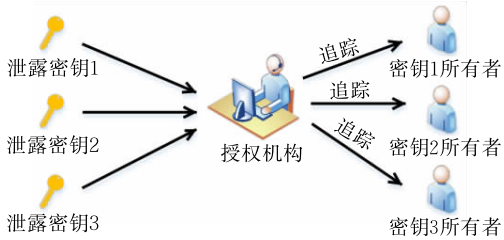

图 3 白盒追踪机制

图 4 黑盒追踪机制

令 Ω 为第 4.1 节中定义的数据共享可追踪 CP-ABE 系统，TDS 系统分为以下六个过程：

(1) 系统建立. AU 运行 Ω 的 Setup 算法，以建立整个系统.

(2) 数据用户注册. 对于每个注册的数据用户，AU 运行 Ω 的 KeyGen 算法为数据用户发出解密密钥 sk.

(3) 数据外包. 若要将共享(加密)数据外包给 CS，数据所有者需要调用 Ω 的 Encrypt 算法以生成密文 ct 并将 ct 发送给 CS.

(4) 访问外包的数据. 为了访问存储在 CS 中的共享(加密)数据，数据用户需要运行 Ω 的 Decrypt 算法以获取明文.

(5) 白盒追踪. 对于一个被泄露的解密密钥 sk，运行 Ω 的 WTrace 算法来追踪泄露 sk 的恶意用户.

(6) 黑盒追踪. 对于泄露的黑盒设备 $\mathcal{D}$，AU 运行 Ω 的 BTrace 算法来追踪构建该设备的恶意用户.

5.3 安全性定义

5.3.1 IND-CPA 安全

TDS 系统的安全性应首先满足底层 CP-ABE 方案的标准语义安全，即选择明文攻击下的密文不可区分性安全(IND-CPA). IND-CPA 安全游戏类似于文献[46]中的 CP-ABE 系统，由敌手 $\mathcal{A}$ 和挑战者 $\mathcal{C}$ 之间的安全游戏 $Game_{\mathrm{IND}}$ 来描述：

初始化阶段. $\mathcal{C}$ 运行 Setup 算法，并将公开参数 pp 提供给 $\mathcal{A}$.

阶段 1. 对每一个 $i\in[q_1]$，$\mathcal{A}$ 针对属性集和码字元组 $(S_i,f^{(i)})$ 进行解密密钥查询. $\mathcal{C}$ 运行 KeyGen 算法生成一组与 $\{(S_i,f^{(i)})\}_{\forall i\in[q_1]}$ 对应的解密密钥 $\{sk_i\}_{\forall i\in[q_1]}$，并发送给 $\mathcal{A}$.

挑战阶段. $\mathcal{A}$ 提交两个等长消息 m_0、m_1 和一

个访问策略 A^*，限制条件是阶段 1 中的属性集合 $S_1,\cdots,S_{q_1}$ 中的任何一个都不能满足访问策略 A^*. $\mathcal{C}$ 随机选择 $\beta\in\{0,1\}$，运行 Encrypt 算法，在 A^* 下加密 m_β，并将加密后的密文发送给 $\mathcal{A}$.

阶段 2. $\mathcal{A}$ 再次针对属性集和码字元组 $\{(S_i,f^{(i)})\}_{\forall i\in[q_1+1,q]}$ 进行解密密钥查询请求，其限制条件仍是所请求的属性集都不满足 A^*. $\mathcal{C}$ 运行 KeyGen 算法，构造一组与 $\{(S_i,f^{(i)})\}_{\forall i\in[q_1+1,q]}$ 对应的解密密钥 $\{sk_i\}_{\forall i\in[q_1+1,q]}$，发送给 $\mathcal{A}$.

猜测. 敌手 $\mathcal{A}$ 输出猜测的值 $\beta'\in\{0,1\}$.

在这个 $Game_{\mathrm{IND}}$ 游戏中，$\mathcal{A}$ 的优势定义为 $Adv=Pr[\beta'=\beta]-1/2$.

定义 5.　若没有概率多项式时间的敌手，能以不可忽略的优势赢得游戏，则 TDS 系统是 IND-CPA 安全的.

选择性 IND-CPA(Selectively IND-CPA)安全是通过添加一个初始化阶段来定义的，即敌手 $\mathcal{A}$ 必须在看到公开参数之前声明其在安全游戏中要挑战的访问策略 A^*.

5.3.2　黑盒追踪安全

黑盒追踪安全要求对于泄露的解密设备，该算法应输出构造该解密设备的恶意用户. 其安全性由敌手 $\mathcal{A}$ 和挑战者 $\mathcal{C}$ 之间的安全游戏 $Game_{\mathrm{BD}}$ 来描述：

系统初始化阶段. $\mathcal{A}$ 首先调用 FC.Gen 算法获取 $(D=\{f^{(1)},\cdots,f^{(\mathcal{N}+n)}\},rk)$，其中 n 是一个正整数. 然后从 D 中随机选取 $\mathcal{N}$ 个码字并发送给 $\mathcal{C}$. $\mathcal{C}$ 以 $\mathcal{N}$ 作为输入运行 Setup 算法，并将公开参数 pp 发送给 $\mathcal{A}$.

查询阶段. 对每一个 $i\in[q]$，$\mathcal{A}$ 重复地请求与(属性集，码字)元组 $(S_i,f^{(i)})$ 对应的解密密钥，$\mathcal{C}$ 运行 KeyGen 算法得到一组与 $\{(S_i,f^{(i)})\}_{\forall i\in[q]}$ 对应的解密密钥 $\{sk_i\}_{\forall i\in[q]}$ 发送给 $\mathcal{A}$. 令 N 表示其解密密钥已被 $\mathcal{A}$ 查询的用户集合.

挑战阶段. $\mathcal{A}$ 给出一个能够解密与访问策略 A^* 相关联密文的解密设备 $\mathcal{D}$.

黑盒追踪. $\mathcal{C}$ 运行与 $\mathcal{D}$ 交互的黑盒追踪算法 BTrace，并获取恶意用户的标识集合 N'.

我们称 $\mathcal{A}$ 赢得游戏 $Game_{\mathrm{BD}}$，若如下两个条件满足：

(1) $\mathcal{D}$ 是一种有用的解密设备. 它能够以很高的概率解密与访问策略 A^* 相关联的密文. 为了简单起见，假设成功解密的概率为 1.

(2) N' 为空或者 $N'\not\subset N$.

定义 6.　若没有概率多项式时间的敌手，能以不可忽略的优势赢得游戏，则 TDS 系统是黑盒可追踪安全的.

5.3.3　白盒追踪安全

白盒追踪安全要求对于泄露的解密密钥，算法应输出泄露该密钥的恶意用户. 其安全性敌手 $\mathcal{A}$ 和模拟器 $\mathcal{B}$ 之间的安全游戏 $Game_{\mathrm{WD}}$ 来描述，其过程如下所示：

初始化阶段. 在输入安全参数 λ 时，$\mathcal{B}$ 同时也将 λ 给 $\mathcal{A}$.

挑战阶段. $\mathcal{A}$ 输出公开参数 pp 并发送给 $\mathcal{B}$. 此外，$\mathcal{A}$ 输出对应于相同的属性集 S 和码字 f 的两个不同解密密钥 sk、sk'.

$\mathcal{A}$ 赢得游戏 $Game_{\mathrm{WD}}$，如果：

(1) 对于 $\mathcal{B}$ 产生的任何有效密文 ct，满足：

① $Decrypt(sk,ct)\neq\perp$；

② $Decrypt(sk',ct)\neq\perp$；

③ $Decrypt(sk,ct)=Decrypt(sk',ct)$.

如果(sk 和 sk′中的)属性集合 S 满足 ct 中的访问策略，则称密文 ct 是有效的.

(2) $\mathcal{A}$ 生成的任何两个不同的解密密钥 sk、sk'(对应于相同的属性集 S 和码字 f)，满足：

① $WTrace(pp,sk)\neq\perp$；

② $WTrace(pp,sk')\neq\perp$；

③ $WTrace(pp,sk)\neq WTrace(pp,sk')$.

定义 7.　若没有概率多项式时间的敌手，能以不可忽略的优势赢得游戏，则 TDS 系统是白盒可追踪安全的.

6　具体构造

本节论述了用于实现白盒/黑盒可追踪性的主要技术路线和技术细节.

6.1　采用的技术

受文献[45]启发，本文主要利用指纹编码技术来实现白盒/黑盒的可追踪性. 具体而言，本文通过包含标识和指纹码字的唯一元组来分配和识别每个云用户. 每个云用户的唯一(标识，码字)元组将用于识别特定用户，以防其行为异常. 在密钥生成阶段，对于每个云用户，唯一指纹码字都隐式地嵌入到其密钥中. 唯一的指纹码字将被逐比特地编码到密钥中，并进一步使用随机项 δ、r 作为“盲化因子”，以保护嵌入密钥中的码字. 将 g^{δ} 项添加到公开参数中以

消除随机元素，进而确保正确解密. 一旦密钥泄露，任何获得该密钥的人都可以执行配对操作以恢复嵌入在密钥中的码字的第 k 个索引的比特值，其中$k\in[1,d]$，从而实现白盒可追踪性.

为实现黑盒可追踪性，本文将相似项$\{C_{j,0},C_{j,1}\}_{\forall j\in[l]}$嵌入到密文中，同时使用户能够在给定的解密设备$\mathcal{D}$声明的访问策略下，通过加密算法，将一个随机选择的消息生成密文. 本文引入了一个巧妙的构造：因子 k，在$[1,d]$中随机选取，且将其作为加密算法的输入. 如果密文由 k 产生，同时$\{C_{j,b}\}_{\forall j\in[l]}$部分可以由$\mathcal{D}$解密，则表明在$\mathcal{D}$中存在一个码字的第 k 个索引位是 b，其中 $b\in\{0,1\}$.

重复执行上述步骤 d 次，本文可以恢复$\mathcal{D}$的整个码字(长度为 d). 通过使用底层指纹编码方案的恢复算法，可以识别出构造该解密设备$\mathcal{D}$的恶意用户. 黑盒追踪算法也可以用于恢复泄露的解密密钥. 但白盒追踪算法比黑盒追踪算法有效得多，如果解密工具是密钥而不是设备，则应利用白盒追踪算法.

本文采用文献[46]中 CP-ABE 的构造，因为其精致构造与高效的特性对于云存储服务中的数据共享而言非常有效. 本文的方法亦可应用于其他 CP-ABE 结构，其具备通用性.

6.2 用于数据共享的底层可追踪 CP-ABE 构造

$Setup(\lambda,U,\mathcal{N})\rightarrow(pp,msk)$. 首先输入安全参数 λ，属性空间 U 以及系统中的用户数量$\mathcal{N}$. 该算法选取一个素数 p 阶双线性群 G，并随机选取以下值：群 G 的一个生成元 $g\in G$、G 中的 U 个群元素 $h_1=g^{b_1},h_2=g^{b_2},\cdots,h_U=g^{b_U}\in G$ 以及 $\alpha,a,\delta\in Z_p,d\in N^*$. 该算法随后初始化抗共谋指纹编码方案 FC：调用 $FC.Gen(1/2^\lambda,\mathcal{N},d)$ 获取$(\{f^{(1)},\cdots,f^{(\mathcal{N})}\},rk)$. 公开参数 pp 设置为

$$(g,g^a,g^\delta,e(g,g)^\alpha,\{h_i\}_{\forall i\in[U]}),$$

主密钥 msk 为：(g^α,rk).

每个用户由一个属性集合 $S\subset U$ 和一个唯一元组来标识，这个元组由一个码字 $f\in D$ 和一个标识 $\tau\in[\mathcal{N}]$组成.

$KeyGen(msk,S,f)\rightarrow sk$. 输入主密钥 msk，一个属性集合 S 和一个码字 $f=f_1\cdots f_d$. 该算法随机选取 $r\in Z_p$，设置与属性集合 S 和码字 f 关联的解密密钥 sk 为

$$\left(K_1=g^\alpha g^{ar},K_2=g^r,\{K_{i,k}=(g^{b_i}g^{\delta(k+f_k)})^r\}_{\forall i\in S,\forall k\in[d]}\right).$$

$Encrypt(pp,(\boldsymbol{M},\rho),m)\rightarrow ct$. 输入公开参数 pp，访问策略$(\boldsymbol{M},\rho)$和一个消息 m，其中 $\boldsymbol{M}$ 是 $l\times n$ 矩阵，ρ 是 $\boldsymbol{M}$ 的每一行 M_i 到一个属性 $\rho(i)$ 的映射. 随机选取 $k\in[d]$和向量 $\boldsymbol{y}=(s,y_2,\cdots,y_n)$，其中 s 是线性秘密分享中被分享的随机秘密. 对于 $\boldsymbol{M}$ 的每一行 M_j，随机选取 $t_j\in Z_p$，得到密文 ct 为

$$\left(\begin{array}{l}C_0=m\cdot e(g,g)^{\alpha s},C_1=g^s,\\ \left\{\begin{array}{l}C_{j,0}=g^{aM_j\boldsymbol{y}}(g^{b_{\rho(j)}}g^{\delta k})^{-t_j},\\ C_{j,1}=g^{aM_j\boldsymbol{y}}(g^{b_{\rho(j)}}g^{\delta(k+1)})^{-t_j},\\ C_{j,2}=g^{t_j}\end{array}\right\}_{\forall j\in[l]}\\ (\boldsymbol{M},\rho),k\end{array}\right).$$

$Decrypt(sk,ct)\rightarrow m/\perp$. 输入与属性集 S 和码字 f 相关联的密钥 sk、与访问结构$(\boldsymbol{M},\rho)$和 k 相关联的密文 ct. 如果 S 不满足$(\boldsymbol{M},\rho)$，输出 $\perp$；否则，计算使得 $\sum_{\rho(i)\in S}\omega_iM_i=(1,0,\cdots,0)$ 的 $\omega_i\in Z_p$ 值.

如果码字 f 的第 k 个比特为 0，则计算：

$$\frac{e(K_1,C_1)}{\prod_{\rho(i)\in S}(e(K_2,C_{i,0})e(K_{\rho(i),k},C_{i,2}))^{\omega_i}}=e(g,g)^{\alpha s}.$$

否则计算：

$$\frac{e(K_1,C_1)}{\prod_{\rho(i)\in S}(e(K_2,C_{i,1})e(K_{\rho(i),k},C_{i,2}))^{\omega_i}}=e(g,g)^{\alpha s}.$$

最后，计算 $C_0/e(g,g)^{\alpha s}$ 以恢复消息 m.

$WTrace(pp,sk)\rightarrow f/\perp$：输入公开参数 pp、一个与属性集合 S 关联的被泄露的密钥 sk. 该算法执行如下操作：

(1) 验证 $e(K_1,g)=e(g,g)^\alpha e(K_2,g^a)$ 是否成立. 如果不成立，停止并输出 $\perp$；否则，执行(2).

(2) 随机选取 $i\in S$，对于每一个 $k\in[d]$，首先验证 $e(K_{i,k},g)=e((g^\delta)^kh_i,K_2)$ 是否成立. 如果成立，则置 $f_k=0$；否则验证 $e(K_{i,k},g)=e((g^\delta)^{k+1}h_i,K_2)$ 是否成立，如果成立，则置 $f_k=1$；否则，终止并输出 $\perp$.

$BTrace_{\mathcal{D}}(pp,(\boldsymbol{M},\rho)_{\mathcal{D}})\rightarrow N'/\perp$. 输入公开参数 pp 和一个(由泄露的解密设备声明的)访问策略$(\boldsymbol{M},\rho)_{\mathcal{D}}$. 给定一个与访问策略$(\boldsymbol{M},\rho)_{\mathcal{D}}$相关联的被泄露的解密设备$\mathcal{D}$，该解密设备$\mathcal{D}$有可能是功能不完整的解密设备. 为简单起见，本文假设$\mathcal{D}$是一个成熟的解密设备(即它可以 100% 成功解密)，对于所有 $k\in[d]$，算法执行如下操作：

(1) 从消息空间中随机选取消息 m.

(2) 运行 $Encrypt(pp,(\boldsymbol{M},\rho)_{\mathcal{D}},m;k)$得到密文 $ct=(C_0,C_1,\{C_{j,0},C_{j,1},C_{j,2}\}_{\forall j\in[l]},(\boldsymbol{M},\rho)_{\mathcal{D}},k)$.

(3) 随机选取 $\hat{C}_{j,1}\in G$，使用$\mathcal{D}$来解密输入的密

文 $ct'=(C_0,C_1,\{C_{j,0},\hat{C}_{j,1},C_{j,2}\}_{\forall j\in[l]},(\boldsymbol{M},\rho)_{\mathcal{D}},k)$，令 $m_{\mathcal{D}}$ 表示 $\mathcal{D}$ 的输出. 如果 $m_{\mathcal{D}}=m$，则置 $f_k=0$；否则执行(4).

(4) 随机选取 $\hat{C}_{j,0}\in G$，使用 $\mathcal{D}$ 来解密输入的密文 $ct''=(C_0,C_1,\{\hat{C}_{j,0},C_{j,1},C_{j,2}\}_{\forall j\in[l]},(\boldsymbol{M},\rho)_{\mathcal{D}},k)$，令 $m'_{\mathcal{D}}$ 表示 $\mathcal{D}$ 的输出. 如果 $m'_{\mathcal{D}}=m$，则置 $f_k=1$；否则终止并输出 $\perp$.

如果算法没有终止，则可以获取 $f^*=f_1\cdots f_d$. 随后调用 $FC.Recover(f^*,rk)$ 获取 $N'\subseteq[\mathcal{N}]$. 最后，输出 N'，找出构建解密设备 $\mathcal{D}$ 的恶意用户.

6.3　TDS 系统具体构造

令 Ω 为第 6.2 节中提出的用于数据共享的可追踪 CP-ABE 系统，TDS 系统的具体构造包括以下六个步骤：

(1) 系统建立. AU运行 Ω 的Setup算法，$Setup(\lambda,U,\mathcal{N})\rightarrow(pp,msk)$，生成公开参数 pp 和系统主密钥 msk.

(2) 数据用户注册. 当数据用户加入系统的请求获得批准后，系统会分配给该数据用户一个属性集合 $S\subseteq U$ 以及一个唯一元组，该唯一元组由一个码字 $f\in D$ 和一个标识 $\tau\in[\mathcal{N}]$ 组成. AU 运行 Ω 的 KeyGen 算法 $KeyGen(msk,S,f)\rightarrow sk$，为数据用户发放解密密钥 sk.

(3) 数据外包. 数据所有者执行如下步骤将其数据外包给 CS. 首先，使用对称加密算法(比如 AES)对文件 m_f 进行加密，此处对称加密密钥随机取自 $m\in G_T$，生成的密文表示为 ct_f. 数据所有者随后定义一个访问策略 $(\boldsymbol{M},\rho)$ 并运行 Ω 的 Encrypt 算法 $Encrypt(pp,(\boldsymbol{M},\rho),m)\rightarrow ct$，将加密后的文件 $ct_f\|ct$ 发送给 CS，其中 ct_f 和 ct 分别是外包文件的文件头和主体.

(4) 访问外包数据. 当数据用户请求外包(加密)文件时，CS 将请求的文件 $ct_f\|ct$ 返回给数据用户. 数据用户随后运行 Ω 的 $Decrypt(sk,ct)\rightarrow m$ 算法获取对称密钥 m，再使用密钥 m 解密 ct_f 获取明文.

(5) 白盒追踪. 对于一个被泄露的解密密钥 sk，可以运行 Ω 的 $WTrace(pp,sk)\rightarrow f$ 算法来识别泄露 sk 的恶意用户.

(6) 黑盒追踪. 对于泄露的解密设备 $\mathcal{D}$(访问策略为 $(\boldsymbol{M},\rho)_{\mathcal{D}}$)，AU 可以运行 Ω 的 BTrace 算法 $BTrace_{\mathcal{D}}(pp,(\boldsymbol{M},\rho)_{\mathcal{D}})\rightarrow N'$ 来追踪构建该设备的恶意用户.

6.4　安全分析

6.4.1　选择性 IND-CPA 安全

由于上述 TDS 系统的构建是基于文献[46]中的 CP-ABE 系统，为了简单起见，本文将把 TDS 系统的选择性 IND-CPA 安全证明简化为文献[46]中的 CP-ABE 系统的选择性 IND-CPA 安全证明. 令 Γ_1、Γ_2 分别表示文献[46]中的 CP-ABE 系统和 TDS 中的 CP-ABE 系统. Γ_1 的安全模型与 Γ_2 的安全模型几乎相同.

引理 1[46].　如果决策 q-并行 BDHE 假设成立，则 Γ_1 具备选择性 IND-CPA 安全.

引理 2.　如果 Γ_1 具备选择性 IND-CPA 安全，则 Γ_2 也具备选择性 IND-CPA 安全.

证明.　假设能找到概率多项式时间的敌手 $\mathcal{A}$，以 $Adv_{\mathcal{A}}\Gamma_2$ 优势来攻破 Γ_2. 构造一个具备 $Adv_{\mathcal{B}}\Gamma_1$ 优势攻破 Γ_1 的概率多项式时间算法 $\mathcal{B}$ ($Adv_{\mathcal{B}}\Gamma_1=Adv_{\mathcal{A}}\Gamma_2$).

建立算法. Γ_1 发送公开参数 $pp_1=(g,g^a,e(g,g)^a,\{h_i=g^{b_i}\}_{\forall i\in U})$ 给 $\mathcal{B}$，$\mathcal{B}$ 随机选取 $\delta,d\in Z_p$，调用 $FC.Gen(1/2^\lambda,\mathcal{N},d)$ 获取 $(D=\{f^{(1)},\cdots,f^{(\mathcal{N})}\},rk)$. $\mathcal{B}$ 将公开参数 $pp=(g,g^a,g^\delta,e(g,g)^a,\{h_i\}_{\forall i\in U},rk)$ 发给 $\mathcal{A}$.

查询阶段 1. 敌手 $\mathcal{A}$ 提交 (S,f) 给 $\mathcal{B}$，请求对应密钥，$\mathcal{B}$ 提交 S 给 Γ_1 获取密钥 $\tilde{sk}=(\tilde{K}=g^ag^{ar},\tilde{L}=g^r,\{\tilde{K}_i=h_i^r\}_{\forall i\in S})$.

对所有 $k\in[d]$，$\mathcal{B}$ 计算 $T_k=(\tilde{L})^{\delta(K+f_k)}$.

令 $K_1=\tilde{K}$，$K_2=\tilde{L}$，$\{K_{i,k}=\tilde{K}_i\cdot T_k\}_{\forall i\in[S],\forall k\in[d]}$，$\mathcal{B}$ 返回给 $\mathcal{A}$ 一个密钥 $sk=(K_1,K_2,\{K_{i,k}\}_{\forall i\in[S],\forall k\in[d]})$.

挑战. $\mathcal{A}$ 提交给 $\mathcal{B}$ 两个等长消息 (m_0,m_1) 和一个访问策略 $(\boldsymbol{M}^*,\rho)$. $\mathcal{B}$ 将 (m_0,m_1) 和 $(\boldsymbol{M}^*,\rho)$ 提交给 Γ_1，获取挑战密文 $\tilde{ct}=(\tilde{C}_0=m_be(g,g)^{as},\tilde{C}_1=g^s,\{\tilde{C}_{j,1}=g^{aM_j^*\mathbf{y}}h_{\rho(j)}^{-t_j},\tilde{C}_{j,2}=g^{t_j}\}_{\forall j\in[l]},(\boldsymbol{M}^*,\rho))$.

然后 $\mathcal{B}$ 随机选取 $k\in[d]$. 对于 $j\in[l]$，计算：$T'_j=(\tilde{C}_{j,2})^{-\delta k}$，$T''_j=(\tilde{C}_{j,2})^{-\delta(k+1)}$. 令 $C_0=\tilde{C}_0$，$C_1=\tilde{C}_1$，$\{C_{j,0}=\tilde{C}_{j,1}T'_j,C_{j,1}=\tilde{C}_{j,1}T''_j,C_{j,2}=\tilde{C}_{j,2}\}_{\forall j\in[l]}$，将挑战密文 $ct=(C_0,C_1,\{C_{j,0},C_{j,1},C_{j,2}\}_{\forall j\in[l]},(\boldsymbol{M}^*,\rho),k)$ 发给 $\mathcal{A}$.

查询阶段 2. 与查询阶段 1 相同.

猜测. $\mathcal{A}$ 猜测 $b'\in\{0,1\}$ 并发送给 $\mathcal{B}$，$\mathcal{B}$ 将 b' 给 Γ_1.

由于上述游戏中公开参数、密钥和挑战密文的分布与实际系统相同，可得到 $Adv_{\mathcal{B}}\Gamma_1=Adv_{\mathcal{A}}\Gamma_2$.

证毕.

定理 1. 若决策 q-并行 BDHE 假定成立，则没有一个概率多项式时间的敌手能使用 $l^* \times n^*$ 阶挑战矩阵以不可忽略的优势选择性地攻破上述系统，其中 $l^*, n^* \leq q$.

证明. 直接从引理 1 和引理 2 推出.

6.4.2 黑盒追踪安全

定理 2. 令 d 表示指纹编码的长度，$|\mathcal{M}|$ 表示消息空间的大小，$Adv_{\mathcal{A}}^{\mathrm{IND}}$ 表示任一能够攻破底层 CP-ABE 语义安全(即 IND-CPA)的敌手的优势. 如果 FC 是完全抗共谋(或 t-抗共谋)的指纹编码方案，则任一攻破上述系统的黑盒追踪安全的敌手的优势为：$Adv_{\mathcal{A}}^{\mathrm{BD}} \leq \varepsilon + d \cdot Adv_{\mathcal{A}}^{\mathrm{IND}} + d/|\mathcal{M}|$，其中 ε 是 FC 的安全参数.

证明. 设存在概率多项式时间的敌手 $\mathcal{A}$，目的是破坏上述系统的黑盒追踪安全性. 敌手 $\mathcal{A}$ 和挑战者 $\mathcal{C}$ 之间的游戏过程如下：

建立算法. $\mathcal{A}$ 调用 FC.Gen 算法获取 $(D = \{f^{(1)}, \cdots, f^{(\mathcal{N}+n)}\}, rk)$，其中 n 是一个正整数. 然后 $\mathcal{A}$ 从 D 中随机选取 $\mathcal{N}$ 个码字并发送给 $\mathcal{C}$. $\mathcal{C}$ 运行 $Setup(\lambda, U, \mathcal{N}) \rightarrow (pp, msk)$ 算法，将得公开参数 pp 给 $\mathcal{A}$.

密钥查询. $\mathcal{A}$ 请求查询密钥，由于挑战者知道主密钥，因此可以正确地答复密钥查询.

黑盒伪造. $\mathcal{A}$ 基于所查询的部分或全部密钥输出解密黑盒 $\mathcal{D}$. 令 N 表示与所查询的密钥相关联的码字集合，$F(N)$ 是 N 的可行集合. 挑战者 $\mathcal{C}$ 运行底层抗共谋指纹编码方案的恢复算法 FC.Recover，获得一个码字 $f^* \in F(N)$，从而得到 $Adv_{\mathcal{A}}^{\mathrm{BD}} \leq \varepsilon$. 否则，如果恢复算法失败，即 $f^* \notin F(N)$，此处需要限制概率 $Pr[f^* \notin F(N)]$.

与文献[45]类似，本文对 $Pr[f^* \notin F(N)]$ 进行了评估，并考虑了如下改进的黑盒追踪算法：给定一个黑盒 $\mathcal{D}$，$\mathcal{D}$ 可以解密任何与访问策略 $(\boldsymbol{M}^*, \rho)_{\mathcal{D}}$ 相关联的密文. 对所有 $k \in [d]$，按以下步骤操作：

(1) 从消息空间随机选取两个消息 m 和 m'.

(2) 如果 N 中的所有码字在第 k 个比特处都包含相同的符号，则运行 $Encrypt(pp, (\boldsymbol{M}, \rho)_{\mathcal{D}}, m; k)$ 获得：$ct = (C_0, C_1, \{C_{j,0}, C_{j,1}, C_{j,2}\}_{\forall j \in [l]}, (\boldsymbol{M}^*, \rho)_{\mathcal{D}}, k)$，令

$ct_0^* = (C_0, C_1, \{C_{j,0}, \hat{C}_{j,1}, C_{j,2}\}_{\forall j \in [l]}, (\boldsymbol{M}^*, \rho)_{\mathcal{D}}, k)$，

$ct_1^* = (C_0, C_1, \{\hat{C}_{j,0}, C_{j,1}, C_{j,2}\}_{\forall j \in [l]}, (\boldsymbol{M}^*, \rho)_{\mathcal{D}}, k)$，

其中 $\hat{C}_{j,0}$ 和 $\hat{C}_{j,1}$ 随机选取自 G.

否则，运行 $Encrypt(pp, (\boldsymbol{M}, \rho)_{\mathcal{D}}, m'; k)$，获得：$ct' = (C_0', C_1', \{C_{j,0}', C_{j,1}', C_{j,2}'\}_{\forall j \in [l]}, (\boldsymbol{M}^*, \rho)_{\mathcal{D}}, k)$，令

$ct_0^* = (C_0', C_1', \{C_{j,0}', \hat{C}_{j,1}', C_{j,2}'\}_{\forall j \in [l]}, (\boldsymbol{M}^*, \rho)_{\mathcal{D}}, k)$，

$ct_1^* = (C_0', C_1', \{\hat{C}_{j,0}', C_{j,1}', C_{j,2}'\}_{\forall j \in [l]}, (\boldsymbol{M}^*, \rho)_{\mathcal{D}}, k)$，

其中 $\hat{C}_{j,0}'$ 和 $\hat{C}_{j,1}'$ 随机选取自 G.

(3) 使用 $\mathcal{D}$ 来解密 (ct_0^*, ct_1^*)，输出 m^*. 对于 N 中所有第 k 个比特为 0 的码字，如果 $m^* = m'$，置 $f_k' = 1$；否则 $f_k' = 0$. 对于 N 中所有第 k 个比特为 1 的码字，如果 $m^* = m'$，置 $f_k' = 0$；否则 $f_k' = 1$.

令 $f' = f_1' f_2' \cdots f_d'$ 为从上述过程中获得的码字. 对于 N 中所有第 k 个比特为 0 的码字，$m^* = m'$ 的概率最大为 $1/|\mathcal{M}|$，$f_k' = 1$ 的最大概率为 $1/|\mathcal{M}|$. 类似地，对于 N 中所有第 k 个比特为 1 的码字，$m^* = m'$ 的概率最大为 $1/|\mathcal{M}|$，$f_k' = 0$ 的最大概率为 $1/|\mathcal{M}|$. 这里可忽略 N 中的一些码字在第 k 个比特处包含 1 或 0 的情况，可得到 $Pr[f^* \notin F(N)] \leq d/|\mathcal{M}|$.

以下给出任意概率多项式时间的敌手 $\mathcal{A}$ 在区分上述改进的黑盒追踪算法和原黑盒追踪算法的优势上界. 值得注意的是，密文的生成是这两种黑盒追踪算法的不同之处. 很容易看出，对于每个位置 $k \in [d]$，$\mathcal{A}$ 在区分这两个黑盒追踪算法的优势等于破坏系统语义安全的优势. 因此，区分这两种黑盒追踪算法的上界是 $d \cdot Adv_{\mathcal{A}}^{\mathrm{IND}}$.

因此，有 $Adv_{\mathcal{A}}^{\mathrm{BD}} \leq \varepsilon + d \cdot Adv_{\mathcal{A}}^{\mathrm{IND}} + d/|\mathcal{M}|$. 证毕.

6.4.3 白盒追踪安全

定理 3. 对于 TDS 系统，在白盒追踪安全游戏中，任意概率多项式时间敌手的优势都可忽略.

证明. 假设存在概率多项式时间的敌手 $\mathcal{A}$，目的是破坏上述系统的白盒追踪安全性. 在白盒追踪安全游戏中，$\mathcal{A}$ 输出与同一属性集合 S 和码字 f 关联的两个密钥：

$$sk = (K_1, K_2, \{K_{i,k}\}_{\forall i \in [S], \forall k \in [d]}) \text{ 和}$$

$$sk' = (K_1', K_2', \{K_{i,k}'\}_{\forall i \in [S], \forall k \in [d]}).$$

$\mathcal{A}$ 赢得安全游戏意味着满足：

(1) 对于 $\mathcal{B}$ 产生的任何有效密文 ct，满足：

① $Decrypt(sk, ct) \neq \perp$；

② $Decrypt(sk', ct) \neq \perp$；

③ $Decrypt(sk, ct') = Decrypt(sk', ct)$.

(2) 对于 $\mathcal{A}$ 产生的 sk 和 sk'，满足：

① $WTrace(pp, sk) \neq \perp$；

② $WTrace(pp, sk') \neq \perp$；

③ $WTrace(pp,sk)\neq WTrace(pp,sk')$.

令 $f=f_1f_2\cdots f_d$ 和 $f'=f'_1f'_2\cdots f'_d$ 分别为 sk 和 sk' 中的码字，由条件(1)知，对于所有 $k\in[d]$，有 $f_k=f'_k$，因此得知 sk 和 sk' 都是有用的密钥，且 $f=f'$. 如果一个密钥能够成功解密一个有效的密文，则它是有用的. 由于 sk 和 sk' 都是有用的密钥，可得到 $WTrace(pp,sk)=f$，$WTrace(pp,sk')=f'$. 然而，由条件(2)有 $WTrace(pp,sk)\neq WTrace(pp,sk')$，否定了 $f=f'$，因此 $\mathcal{A}$ 赢得 $Game_{WD}$ 的概率可以忽略.　　证毕.

7　性能评估

7.1　理论分析

在功能和性能方面，分别将本文所提系统与其他支持白盒或黑盒可追踪的相关工作进行比较. 表 1 比较了本文系统和其他一些具有白盒可追踪的相关工作. 对比结果表明，本文所提系统在白盒可追踪方面取得了更好的性能，更适用于在云环境中进行数据共享. 在方案构造方面，如 6.1 节中所述，本文提出的白盒追踪构造主要依赖于指纹编码技术. 不同文献实现白盒追踪的方式各有不同，文献[30,38]基于追踪列表实现白盒追踪，文献[32]基于合数阶下的非交互承诺，文献[34]基于交互式协议，文献[35]基于签名技术和门限秘密共享技术，文献[36-37]则直接将用户身份置于用户密钥中.

表 1　支持白盒追踪的相关工作比较①

文献	白盒追踪	素数阶群	追踪存储开销	追踪计算开销②
[30]	√	×	线性③	$(5+2\lvert S\rvert)P+Sea$
[32]	√	×	几乎不占	$(7+2\lvert S\rvert)P$
[34]	√	×	几乎不占	$(7+2\lvert S\rvert)P+Ext$
[35]	√	√	常量	$(5+4\lvert S\rvert)P+Dec$
[36]	√	√	常量	$(1+5\lvert S\rvert)P$
[37]	√	√	常量	$(1+6\lvert S\rvert)P$
[38]	√	√	常量	$(5+4\lvert S\rvert)P$
本文	√	√	无	$(3+2d)P$

注：① $\lvert S\rvert$ 表示解密密钥中属性集合的大小，P 表示配对操作，d 是指纹编码的长度.

② 对于追踪的配对计算，文献[30,34-35]需要在追踪过程之前运行密钥完整性检查，而本工作将密钥完整性检查集成到恢复过程中. Sea 表示在文献[30]中的搜索用户列表过程，Dec 表示在文献[35]中概率加密方案的解密操作，Ext 表示在文献[34]中承诺的提取操作.

③ 文献[30]中用于追踪的系统存储随用户数量的增加而线性增长.

表 2 比较了本文系统和其他具有黑盒可追踪的相关工作. 对比结果表明，本文所提系统具有比文献[39-41,43]显著更短的常量公开参数，且与系统中的用户数量无关. 此外，本文系统的密文大小比文献[39-40]的密文要短得多，且与系统中的用户数量无关. 这使得本文系统优于其他系统. 在方案构造方面，如 6.1 节中所述，本文提出的黑盒追踪构造主要基于指纹编码技术和密文刷选机制. 同样地，不同文献实现黑盒追踪的方式各有不用，文献[39-40]基于广播加密中的密文刷选机制，文献[41]基于层次身份基加密和密文刷选机制，而文献[43]则基于撤销列表.

表 2　支持黑盒追踪的相关工作比较①

文献	黑盒追踪	素数阶群	公开参数大小	密文大小
[39]	√	×	$\lvert U\rvert+3+4\sqrt{N}$	$2l+17\sqrt{N}$
[40]	√	×	$\lvert U\rvert+3+4\sqrt{N}$	$2l+17\sqrt{N}$
[41]	√	×	$\lvert U\rvert+8+N$	$2l+5$
[43]	√	√	$\lvert U\rvert+7+N$	$3l+3$
本文	√	√	$\lvert U\rvert+5$	$3l+3$

注：① $\lvert U\rvert$ 表示属性空间的大小，N 是系统中用户的数量，l 是访问策略的大小.

7.2　实验分析

本文使用基于配对的密码库[47]中的 A-型椭圆曲线，其中 Z_p、G、G_T 中元素的位长分别为 160 比特、1024 比特和 1024 比特. 实验在 Intel(R) Core(TM) i5-6300u CPU@2.4GHz，8GB RAM，Windows 10 64 位操作系统的计算机上执行. 使用的编程语言是 Java，JDK32-1.6.0 和 JPBC-2.0.0 版本. 时间以毫秒为单位. 为模拟最坏情况，以“‘S_1’ AND ‘S_2’… AND ‘S_n’”作为生成密文时使用的访问策略，其中 S_i 是一个属性. 为了保证数据的准确性，本文设置了 20 种不同的访问策略，每一个实例重复 30 次，然后取平均值. 本文设置属性空间的维度至少是设定访问控制策略中属性数目的 2 倍. 具体而言，当访问策略中属性的数目为 8、16、32 和 64 时，则相对应的属性空间的维度为 16、32、64 和 128. 本文设置的 20 种不同的访问策略是度量最坏情况访问策略中门限均为 AND 门限下的实验过程，对于访问策略中的不同属性数目，我们分别为每一种情形(即 $\lvert S\rvert=8$、$\lvert S\rvert=16$、$\lvert S\rvert=32$ 和 $\lvert S\rvert=64$)设置 5 组不同的具体属性. 本文实验目的是通过与基础 CP-ABE 方案(文献[46]，用 W11 表示)各阶段运行效率的对比来评估本文 TDS 系统的效率，其中 W11 中没有考虑叛徒追踪功能. 本文检查执行单个阶段的时间开销(包括加密时间和解密时间). 由于本文 TDS 系统考虑了叛徒追踪问题，所以增添了额外的时间开销. 图 5(a)显示 TDS 系统加密过程的时间成本仅略高于 W11 的时间成本. 图 5(b)显示 TDS 系统解密过程的时间开销几乎与 W11 的时间开销相同，其中指

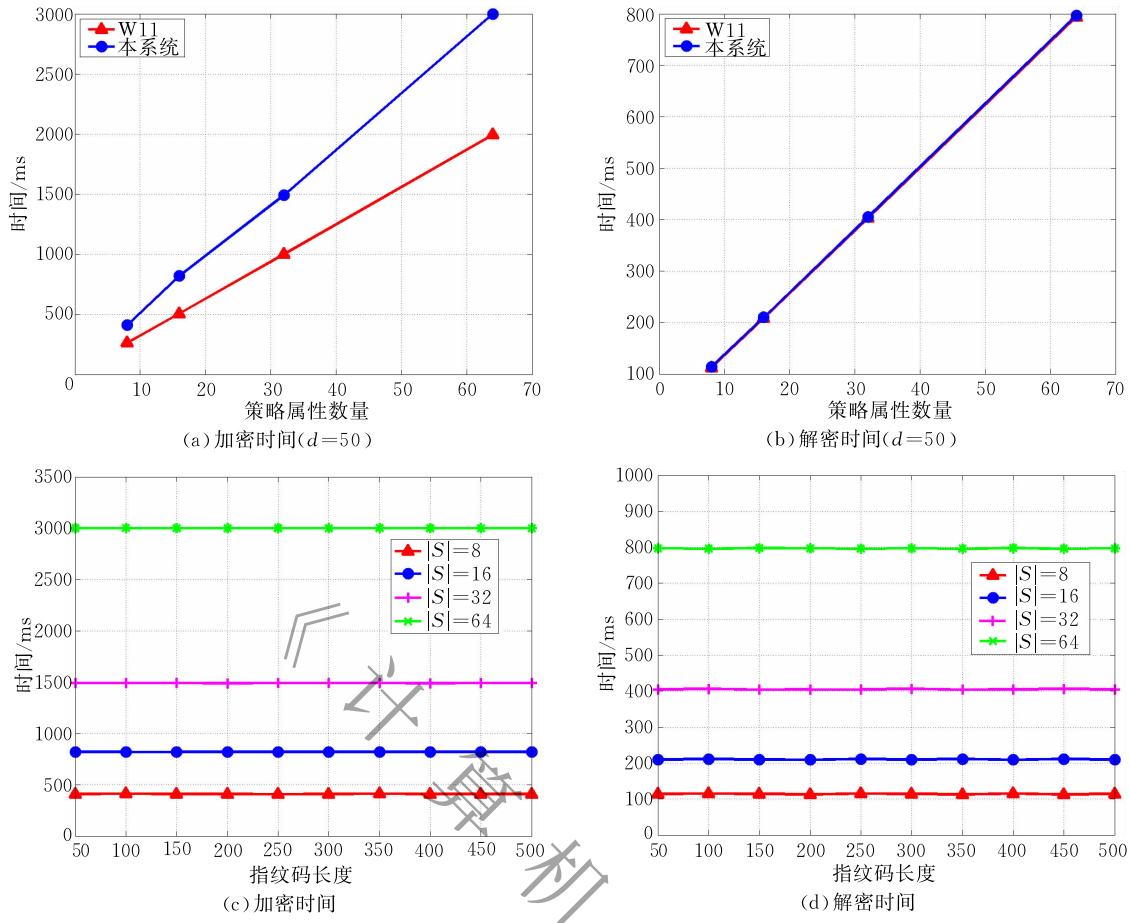

图 5 实验结果(图中|S|表示属性的数量)

纹编码的长度为 50.

综上所述,与文献[46]中的基础 CP-ABE 相比,本文 TDS 系统同时实现了白盒可追踪和黑盒可追踪机制,且没有产生显著的开销. 此外,图 5(c)显示加密外包文件的时间并不随着底层指纹编码长度的增长而变长. 同时,图 5(d)表明解密外包文件的时间也不随底层指纹编码长度的增长而变长. 实验结果表明本文 TDS 系统更适用于云数据共享.

8 总结与展望

针对云存储服务中基于 CP-ABE 数据共享机制中可能存在的解密权限泄露这一问题,本文提出了一种可追踪的细粒度数据共享访问控制系统,能够有效地追踪泄露解密密钥的恶意用户. 同时,对于泄露密钥的解密设备,该系统还可以有效地识别出构建该设备的恶意用户. 理论分析和仿真结果均证实了本文方案的可行性和有效性. 值得注意的是,当前大多数的 CP-ABE 系统都易受到解密权限泄露(或滥用)的攻击,设计一个新的 CP-ABE 模型,从根本上防止解密权限泄露(或滥用)的问题将是一个重要的研究方向. 在今后的工作中,我们将着眼于设计新的安全模型,从根本上解决解密权限泄露的问题.

致 谢 衷心感谢评审专家和编辑们对本文提出的宝贵意见和建议!

参 考 文 献

[1] Armbrust M, Fox A, Griffith R, et al. A view of cloud computing. Communications of the ACM, 2010, 53(4): 50-58

[2] Ning Jian-Ting, Poh G S, Huang Xin-Yi, et al. Update covery attacks on encrypted database within two updates using range queries leakage. IEEE Transactions on Dependable and Secure Computing, 2020, DOI: 10.1109/TDSC.2020.3015997

[3] Ning Jian-Ting, Chen Jia-Geng, Liang Kai-Tai, et al. Efficient encrypted data search with expressive queries and flexible update. IEEE Transactions on Services Computing, 2020, DOI: 10.1109/TSC.2020.3004988

[4] Sahai A, Waters B. Fuzzy identity-based encryption// Proceedings of the 24th Annual International Conference on

the Theory and Applications of Cryptographic Techniques. Aarhus, Denmark, 2005: 457-473

[5] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data//Proceedings of the 13th ACM Conference on Computer and Communications Security. Alexandria, USA, 2006: 89-98

[6] Attrapadung N. Dual system encryption via doubly selective security: Framework, fully secure functional encryption for regular languages, and more//Proceedings of the 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques. Copenhagen, Denmark, 2014: 557-577

[7] Lewko A B, Waters B. New techniques for dual system encryption and fully secure HIBE with short ciphertexts//Proceedings of the 7th Theory of Cryptography Conference. Zurich, Switzerland, 2010: 455-479

[8] Lewko A B, Waters B. New proof methods for attribute-based encryption: Achieving full security through selective techniques//Proceedings of the 32nd Annual Cryptology Conference. Santa Barbara, USA, 2012: 180-198

[9] Okamoto T, Takashima K. Fully secure functional encryption with general relations from the decisional linear assumption//Proceedings of the 30th Annual Cryptology Conference. Santa Barbara, USA. 2010: 191-208

[10] Chase M, Chow S S M. Improving privacy and security in multi-authority attribute-based encryption//Proceedings of the 16th ACM Conference on Computer and Communications Security. Chicago, USA, 2009: 121-130

[11] Han Jin-Guang, Susilo W, Mu Yi, et al. PPDCP-ABE: Privacy-preserving decentralized ciphertext policy attribute-based encryption//Proceedings of the 19th European Symposium on Research in Computer Security. Wroclaw, Poland, 2014, II: 73-90

[12] Lewko A B, Waters B. Decentralizing attribute-based encryption//Proceedings of the 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Tallinn, Estonia, 2011: 568-588

[13] Zhong Hong, Cui Jie, Zhu Wen-Long, et al. Efficient and verifiable muti-authority attribute based encryption scheme. Journal of Software, 2018, 29(7): 2006-2017(in Chinese)
(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. 软件学报，2018，29(7)：2006-2017)

[14] Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts//Proceedings of the 20th USENIX Security Symposium. San Francisco, USA, 2011: 523-538

[15] Ning Jian-Ting, Cao Zhen-Fu, Dong Xiao-Lei, et al. Auditable σ-time outsourced attribute-based encryption for access control in cloud computing. IEEE Transactions on Information Forensics and Security, 2018, 13(1): 94-105

[16] Parno B, Raykova M, Vaikuntanathan V. How to delegate and verify in public: Verifiable computation from attribute-based encryption//Proceedings of the 9th Theory of Cryptography Conference. Taormina, Italy, 2012: 422-439

[17] Sun Yi, Chen Xing-Yuan, Du Xue-Hui, et al. Dynamic authenticated method for outsourcing data stream with access control in cloud. Chinese Journal of Computers, 2017, 40(2): 337-350(in Chinese)
(孙奕，陈性元，杜学绘等. 一种具有访问控制的云平台下外包数据流动态可验证方法. 计算机学报，2017，40(2)：337-350)

[18] Attrapadung N, Hanaoka G, Matsumoto T, et al. Attribute based encryption with direct efficiency tradeoff//Proceedings of the 14th International Conference on Applied Cryptography and Network Security. London, UK, 2016: 249-266

[19] Attrapadung N, Yamada S. Duality in ABE: Converting attribute based encryption for dual predicate and dual policy via computational encodings//Proceedings of the Cryptographer's Track at the RSA Conference. San Francisco, USA, 2015: 87-105

[20] Chen Jie, Gay R, Wee H. Improved dual system ABE in prime-order groups via predicate encodings//Proceedings of the 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques. Sofia, Bulgaria, 2015, II: 595-624

[21] Zhang Yan-Hua, Hu Yu-Pu, Chen Jiang-Shan. Hidden attribute-based signatures without anonymity revocation from lattices. Chinese Journal of Computers, 2018, 41(2): 481-492(in Chinese)
(张彦华，胡予濮，陈江山等. 格上无匿名性撤销的隐藏的属性签名. 计算机学报，2018，41(2)：481-492)

[22] Li Ji-Guo, Yao Wei, Zhang Yi-Chen, et al. Flexible and fine-grained attribute-based data storage in cloud computing. IEEE Transactions on Services Computing, 2017, 10(5): 785-796

[23] Liang Kaitai, Susilo W. Searchable attribute-based mechanism with efficient data sharing for secure cloud storage. IEEE Transactions on Information Forensics and Security, 2015, 10(9): 1981-1992

[24] Liu J K, Au M H, Susilo W, et al. Secure sharing and searching for realtime video data in mobile cloud. IEEE Network, 2015, 29(2): 46-50

[25] Wan Zhi-Guo, Liu Jun-E, Deng R H. HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Transactions on Information Forensics and Security, 2012, 7(2): 743-754

[26] Yang Yan-Jiang, Liu J K, Liang Kaitai, et al. Extended proxy-assisted approach: Achieving revocable fine-grained encryption of cloud data//Proceedings of the 20th European Symposium on Research in Computer Security. Vienna, Austria, 2015: 21-25

[27] Ning Jian-Ting, Huang Xin-Yi, Susilo W, et al. Dual access control for cloud-based data storage and sharing. IEEE Transactions on Dependable and Secure Computing, 2020, DOI: 10.1109/TDSC.2020.3011525

[28] Li Jin, Ren Kui, Kim K. A2BE: Accountable attribute based encryption for abuse free access control. IACR Cryptology ePrint Archive, 2009: 118

[29] Li Jin, Huang Qiong, Chen Xiaofeng, et al. Multi-authority ciphertext-policy attribute-based encryption with accountability//Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security (ASIACCS 2011). Hong Kong, China, 2011: 386-390

[30] Liu Zhen, Cao Zhen-Fu, Wong D S. White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures. IEEE Transactions on Information Forensics and Security, 2013, 8(1): 76-88

[31] Ning Jian-Ting, Cao Zhen-Fu, Dong Xiao-Lei, et al. CryptCloud+: Secure and expressive data access control for cloud storage. IEEE Transactions on Services Computing, 2021, 14(1): 111-124

[32] Ning Jian-Ting, Cao Zhen-Fu, Dong Xiao-Lei, et al. White-box traceable CP-ABE for cloud storage service: How to catch people leaking their access credentials effectively. IEEE Transactions on Dependable and Secure Computing, 2018, 15(5): 883-897

[33] Ning Jian-Ting, Cao Zhen-Fu, Dong Xiao-Lei, et al. Large universe ciphertext-policy attribute-based encryption with white-box traceability//Proceedings of the 19th European Symposium on Research in Computer Security. Copenhagen, Denmark, 2014, II: 55-72

[34] Ning Jian-Ting, Dong Xiao-Lei, Cao Zhen-Fu, et al. Accountable authority ciphertext-policy attribute-based encryption with white-box traceability and public auditing in the cloud//Proceedings of the 20th European Symposium on Research in Computer Security. Vienna, Austria, 2015, II: 270-289

[35] Ning Jian-Ting, Dong Xiao-Lei, Cao Zhen-Fu, et al. White-box traceable ciphertext-policy attribute-based encryption supporting flexible attributes. IEEE Transactions on Information Forensics and Security, 2015, 10(6): 1274-1288

[36] Zhang Kai, Li Hui, Ma Jian-Feng, et al. Efficient large-universe multi-authority ciphertext-policy attribute-based encryption with white-box traceability. Science China Information Sciences, 2018, 61(3): 032102

[37] Sethi K, Pradhan A, Bera P. Practical traceable multi-authority CP-ABE with outsourcing decryption and access policy updation. Journal of Information Security and Applications, 2020, 51: 102435

[38] Li Qi, Zhu Hong-Bo, Ying Zuo-Bin, et al. Traceable ciphertext-policy attribute-based encryption with verifiable outsourced decryption in eHealth cloud. Wireless Communications and Mobile Computing, 2018: 1-12. DOI: 10.1155/2018/1701675

[39] Liu Zhen, Cao Zhen-Fu, Wong D S. Blackbox traceable CP-ABE: How to catch people leaking their keys by selling decryption devices on eBay//Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security. Berlin, Germany, 2013: 475-486

[40] Liu Zhen, Cao Zhen-Fu, Wong D S. Traceable CP-ABE: How to trace decryption devices found in the wild. IEEE Transactions on Information Forensics and Security, 2015, 10(1): 55-68

[41] Ning Jian-Ting, Cao Zhen-Fu, Dong Xiao-Lei, et al. Traceable CP-ABE with short ciphertexts: How to catch people selling decryption devices on eBay efficiently//Proceedings of the 21st European Symposium on Research in Computer Security, 2016, II: 551-569

[42] Ning Jian-Ting, Cao Zhen-Fu, Dong Xiao-Lei, et al. Traceable and revocable CP-ABE with shorter ciphertexts. Science China Information Sciences, 2016, 59(11): 119102:1-119102:3

[43] Zhao Qian-Qian, Wu Gao-Fei, Ma Hua, et al. Black-box and public traceability in multi-authority attribute based encryption. Chinese Journal of Electronics, 2020, 29(1): 106-112

[44] Beimel A. Secure schemes for secret sharing and key distribution[Ph. D. dissertation]. Israel Institute of Technology, Technion, Haifa, Israel, 1996

[45] Boneh D, Naor M. Traitor tracing with constant size ciphertext//Proceedings of the 2008 ACM Conference on Computer and Communications Security. Alexandria, USA, 2008: 501-510

[46] Waters B. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization//Proceedings of the 14th International Conference on Practice and Theory in Public Key Cryptography. Taormina, Italy, 2011: 53-70

[47] Lynn B, et al. The pairing-based cryptography library. Internet: Crypto. stanford. edu/pbc/[Mar. 27, 2013], 2006

NING Jian-Ting, Ph. D., professor. His research interests include cryptography and information security.

HUANG Xin-Yi, Ph. D., professor. His research interests include cryptography and information security.

WEI Li-Fei, Ph. D., associate professor. His research interests include applied cryptography and data security.

MA Jin-Hua, Ph. D., lecturer. Her research interests include cryptography and information security.

RONG Jing, M. S., lecturer. Her research interest is cryptography.

Background

The topic studied in this paper belongs to the traceability of malicious secure data sharing by employing cloud storage service. To date cloud storage service has replaced traditional data outsourcing techniques and provides a scalable mechanism for Internet users to enforce convenient data management, and this is a hot research topic. Despite offering the benefits of flexible data access and sharing, the cloud-based storage service yields the concerns on the confidentiality of outsourced data, which is one of the main obstacles hindering its wide usage. Public key encryption (PKE) allows data owners to securely "transfer" sensitive data to a recipient. Attribute-based encryption (ABE) has been put forth to enable data sharing with knowledge of fuzzy description. It explores the traditional PKE into a more general setting where data can be shared with a group of receivers by specifying access policy. However, an intrinsic feature of ABE may hinder its wide deployment in real-world applications. The one-to-many data sharing mode cannot provide a solid approach to identify any malicious insiders who are granted decryption rights but being keen to leak the rights to unauthorized system users (note that the purpose of doing so may be various, for example, the system insider may sell its decryption rights for profit). This research work focuses on dealing with this problem.

In this paper, we investigate the decryption privilege leakage problem when deploying ciphertext-policy attribute-based encryption (CP-ABE) in data sharing mechanism for cloud storage service. Our proposed system provides two types of tracing, namely white-box and black-box tracing mechanisms. We state that the former is more efficient than the latter. Specifically, if the leaked decryption privilege is advertised to be a decryption key, one may make use of the white-box tracing; but if the decryption privilege is in the form of a decryption device, the black-box tracing shall be leveraged. The main features of our proposed system are as follows: (1) Authorized access over shared data in the cloud; (2) White-box Traceability; (3) Black-box Traceability; (4) Public tracing; and (5) Almost-no-storage for traceability.

The authors have been working on this research topic for more than 8 years, most of the research outputs were published in prestigious journals and conferences, including ACM CCS, IEEE TDSC, IEEE TIFS. The new construction presented in this work is proposed for enabling efficient traceability of malicious insiders in attribute-based cloud data sharing. Our novel construction guarantees fine-grained access control over cloud-based encrypted data but also the traceability of "leaked" decryption privilege.

This work is supported in part by the National Natural Science Foundation of China (Grant Nos. 62032005, 61972094, and 61872087), the Science Foundation of Fujian Provincial Science and Technology Agency (Grant No. 2020J02016) and the Young Talent Promotion Project of Fujian Science and Technology Association.