软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2023,34(7):3329−3342 [doi: 10.13328/j.cnki.jos.006526]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

一种支持分级用户访问的文件分层 CP-ABE 方案*

刘帅南[1], 刘 彬[1], 郭 真[1], 冯朝胜[1,2], 秦志光[2], 卿 昱[3]

[1](四川师范大学 计算机科学学院, 四川 成都 610101)

[2](网络与数据安全四川省重点实验室 (电子科技大学), 四川 成都 610054)

[3](中国电子科技集团公司第三十研究所, 四川 成都 610041)

通信作者: 冯朝胜, E-mail: csfenggy@sicnu.edu.cn

摘 要: 文件分层的密文策略基于属性的加密 (FH-CP-ABE) 方案实现了同一访问策略的多层次文件加密, 节省了加解密的计算开销和密文的存储开销. 然而, 目前的文件分层 CP-ABE 方案不支持分级用户访问, 且存在越权访问的问题. 为此, 提出一种支持分级用户访问的文件分层 CP-ABE 方案. 在所提方案中, 通过构造分级用户访问树, 并重新构造密文子项以支持分级用户的访问需求, 同时消除用户进行越权访问的可能性. 安全性分析表明, 所提方案能够抵御选择明文攻击. 理论分析和实验分析均表明, 与相关方案相比, 所提方案在计算和存储方面具有更高的效率.

关键词: 基于属性的加密; 文件分层; 分级用户访问; 越权访问; 选择明文攻击

中图法分类号: TP309

中文引用格式: 刘帅南, 刘彬, 郭真, 冯朝胜, 秦志光, 卿昱. 一种支持分级用户访问的文件分层CP-ABE方案. 软件学报, 2023, 34(7): 3329–3342. http://www.jos.org.cn/1000-9825/6526.htm

英文引用格式: Liu SN, Liu B, Guo Z, Feng CS, Qin ZG, Qing Y. File Hierarchy CP-ABE Scheme Supporting Graded User Access. Ruan Jian Xue Bao/Journal of Software, 2023, 34(7): 3329–3342 (in Chinese). http://www.jos.org.cn/1000-9825/6526.htm

File Hierarchy CP-ABE Scheme Supporting Graded User Access

LIU Shuai-Nan[1], LIU Bin[1], GUO Zhen[1], FENG Chao-Sheng[1,2], QIN Zhi-Guang[2], QING Yu[3]

[1](School of Computer Science, Sichuan Normal University, Chengdu 610101, China)

[2](Network and Data Security Key Laboratory of Sichuan Province (University of Electronic Science and Technology of China), Chengdu 610054, China)

[3](The 30th Research Institute of China Electronics Technology Group Corporation, Chengdu 610041, China)

Abstract: The file hierarchy ciphertext policy attribute-based encryption (FH-CP-ABE) scheme realizes multi-level files encryption with the single access policy, which saves the computation cost of encryption and decryption and the storage cost of ciphertext. Nevertheless, the existing file hierarchy CP-ABE scheme cannot support graded user access, while suffers due to the unauthorized access. For this reason, a file hierarchy CP-ABE scheme that supports graded user access is proposed. In the proposed scheme, the graded user access tree is constructed, and the ciphertext subsections are reconstructed to support the access requirements of graded users, thus eliminate the possibility of users to conduct unauthorized access. The security analysis shows that the proposed scheme can resist selective chosen-plaintext attack. Both theoretical and experimental analyses show that the proposed scheme is more efficient in terms of computation and storage compared to related scheme.

Key words: attribute-based encryption (ABE); file hierarchy (FH); graded user access; unauthorized access; selective chosen-plaintext attack

在实际应用中, 共享到云端的多个文件通常具有层次结构. 然而, 利用传统的 CP-ABE 方案实现多文件加密时需构造多个不同的访问策略, 导致加解密效率和存储效率低下. 文件分层 CP-ABE 方案 (FH-CP-ABE) 将多个具有

* 基金项目: 国防科技重点实验室基金 (6142103010709); 国家自然科学基金 (61373163)
收稿时间: 2021-04-14; 修改时间: 2021-06-21, 2021-10-05; 采用时间: 2021-11-01; jos 在线出版时间: 2022-10-14
CNKI 网络首发时间: 2022-11-15

层次关系的访问策略进行整合, 实现了在同一访问策略下加密多个层次文件, 节省了加解密的计算开销和密文的存储开销. 然而, 若用户仅存在等级关系, 而属性不存在包含关系, 那么 FH-CP-ABE 方案无法满足分级用户的访问需求. 同时, FH-CP-ABE 方案中用户通过解密操作计算可获得 $e(g,g)^{\alpha q_{child_i}(0)}$, 同时可获取密文子项 $\widehat{C}_{(x,y),i}=e(g,g)^{\alpha\left(q_{(x,y)}(0)+q_{child_i}(0)\right)}\cdot H_2\left(e(g,g)^{\alpha q_{(x,y)}(0)}\right)$ 及 $\widehat{C}_{(x,y),j}=e(g,g)^{\alpha\left(q_{(x,y)}(0)+q_{child_j}(0)\right)}\cdot H_2\left(e(g,g)^{\alpha q_{(x,y)}(0)}\right)$, 那么该用户通过计算 $\widehat{C}_{(x,y),j}/\widehat{C}_{(x,y),i}\cdot e(g,g)^{\alpha q_{child_i}(0)}$ 可获得 $e(g,g)^{\alpha q_{child_j}(0)}$. 用户即使不满足同一层次中其他访问子策略, 也可通过以上计算完成越权访问.

针对上述问题, 提出一种支持分级用户的文件分层 CP-ABE 方案. 本文的具体贡献包括:

(1) 构造分级用户访问树. 基于分层访问树, 引入控制节点和虚拟属性的概念, 构造用户分级访问模型以支持上述访问需求.

(2) 设计一种支持分级用户访问的文件分层 CP-ABE 方案. 基于分级用户访问树和 CP-ABE 方案, 重新构造密文子项以支持多层次文件的加密, 同时消除用户进行越权访问的可能性.

(3) 证明方案的有效性. 安全性分析表明. 本文方案能够抵御选择明文攻击. 性能分析表明, 本文方案与相关方案相比, 在加解密和存储方面均有更高的效率.

本文首先指出文件分层 CP-ABE 方案存在的主要问题, 并给出解决问题的方法和主要贡献. 本文第 1 节对相关研究进行了总结. 第 2 节给出本文所用的预备知识. 第 3 节给出本文的系统模型. 第 4 节给出所提方案的具体构造. 第 5 节给出所提方案的安全性分析. 第 6 节给出所提方案的性能分析. 最后对本文进行总结.

1 相关研究

如今, 互联网和信息技术的飞速发展推动了社会的进步, 越来越多的人将数据存储在云端. 然而, 全球范围内爆发的众多信息泄露事故, 使得云存储中数据的安全性和隐私性成为云计算发展中所面临的重大难题. 解决数据安全存储的一个直接有效的方法就是在数据上传前进行加密, 但是传统的加密机制仅适用于“一对一”及粗粒度的访问控制. 为解决这一问题, 基于属性的加密 ABE (attribute-based encryption)[1]被提出. 随后, 密钥策略基于属性的加密 KP-ABE (ciphertext policy attribute-based encryption)[2]和密文策略基于属性的加密 CP-ABE (ciphertext policy attribute-based encryption)[3]相继被提出. 其中, KP-ABE 将访问策略隐藏在密钥中, 而 CP-ABE 方案将访问策略隐藏在密文中.

基于属性的加密方案通常没有考虑将属性之间的关系列入研究范围. 然而, 在实际应用中, 属性之间往往存在层次关系, 因此有必要提出既能实现细粒度访问, 又能反映属性层次关系的方案. 2002 年, Horwritz 等人[4]提出了基于身份的分层加密 (hierarchical id-based encryption, HIBE) 的概念, 该方案将用户划分为不同层次, 每一层用户对应一个 PKG, 并从根节点 PKG 向各层节点 PKG 分发创建密钥和身份认证等任务, 提高了 IBE 的效率. 同年, Gentry 等人[5]提出了分层的基于身份加密 (HIBE) 方案, 该方案可抵御随机预言模型下的合谋攻击. 2004 年, Boneh 等人[6]构造了一个有效的 HIBE 方案, 并在标准模型下证明了其安全性. 然而, 该方案的密钥大小和密文大小会随着层次的增多而增大. 为解决这一问题, Boneh 等人[7]提出了密文大小恒定的 HIBE 方案. 2012 年, Tsai 等人[8]提出了支持公共撤销机制的 HIBE (RHIBE) 方案. 2009 年, Li 等人[9]首次提出了 ABE 与 HIBE 相结合的基于属性加密 (hierarchical attribute-based encryption, HABE) 方案, 该方案通过对属性进行分类, 并构造不同的属性树来实现层次化的 ABE. 然而, 该方案缺乏对访问控制的描述. 2010 年, Wang 等人[10]提出了云计算环境下的分层 ABE 方案. 2012 年, Wan 等人[11]提出了一种灵活、可扩展的分层访问控制 (HASBE), 该方案将 ASBE 扩展到用户的层次结构, 使其不仅具有可扩展的层次结构, 而且还继承了 CP-ASBE 中灵活、细粒度的访问控制. Deng 等人[12]于 2014 年提出了一种密文更短的分层 CP-ABE 方案, 该方案支持分层属性和密钥委托, 但该方案在加解密方面的性能并不理想. 2015 年, Wang 等人[13]提出了一种基于 LISS 的密文策略分层属性加密方案. 同年, Chandar 等人[14]提出了一种云环境中支持代理重加密的分层 ABE 方案, 该方案将 HABE 拓展到 KP-ABE 方案, 实现了用户访问权限的数据机密性和用户密钥的可靠性. 2017 年, Huang 等人[15]提出了一种基于属性加密 (ABE) 和基于属性签

名 (ABS) 的安全高效的数据协作方案. 同年, Lin 等人[16]提出了一种位置分层的基于属性加密的访问控制方案, 该方案中用户可以根据自己的特殊情况自定义对位置信息的访问, 然后实现位置信息的分层访问控制. 2020 年, Ali 等人[17]提出了全分布式可撤销分层 CP-ABE (FDR-CP-HABE) 方案, 该方案在密钥委托和用户撤销阶段具备高度的灵活性和可伸缩性. 上述分层 ABE 方案中的核心思路是由父授权域管理其子授权域, 上层授权域创建下一层域的密钥. 如此, 授权中心通过将密钥创建工作分配到多个域, 从而减轻了自身的负担.

在实际应用场景中, 云环境下共享的数据文件大都具有层次关系的特点, 为解决此问题, 文件分层 CP-ABE 方案被相继提出. 2014 年, Wang 等人[18]提出一种新型的文件分层访问控制 (FHAC) 方案, 该方案首次将分层的概念与访问结构融合在一起, 实现了单一访问策略下共享多个分层文件. 基于 FHAC 方案, Wang 等人[19]于 2016 年又提出一种高效的文件分层 CP-ABE (FH-CP-ABE) 方案, 该方案将多层访问结构树进行整合, 若用户能够解密访问结构中某个层次节点对应的文件, 那么所有较低层次节点对应的文件都可以被该用户解密. 但该方案存在越权访问的安全问题. 2017 年, Jiang 等人[20]提出了一种基于分层属性的直接撤销加密方案, 该方案不仅可以对多层次文件进行加密, 而且允许授权或非授权用户进行撤销. 2018 年, Sandhia 等人[21]提出了多授权的隐藏分层文件 CP-ABE 方案, 该方案可用于多个云服务提供商的数据保护. 同年, Guo 等人[22]提出一种应用于个人健康档案的多授权分层 CP-ABE 方案, 该方案在多授权机构下实现了对分层数据的加密, 避免了密钥托管问题, 同时可抵御 N–1 个授权机构的合谋攻击. 随后, Kang 等人[23]发现 Guo 的方案在解密部分存在漏洞, 因此提出了一种改进的文件分层 MA-ABE 方案, 既确保数据安全性和私密性, 又降低了解密开销. 2019 年, Chandrasekaran 等人[24]提出一种高效的非对称文件分层 CP-ABE 方案 (AFH-CP-ABE), 该方案使用优化的 Tate 配对实现文件分层加解密, 提高了加解密效率. 2020 年, He 等人[25]基于 LSSS 提出了高效的基于属性的分层访问控制方案 (AHAC), 该方案的加解密效率上略高于 FH-CP-ABE 方案. 同年, Challagidad 等人[26]提出了一种基于属性的多授权访问控制方案, 该方案由角色分层算法 (RHA) 和分层访问结构 (HAS) 组成. 以上方案均基于 FH-CP-ABE 方案的构造, 实现了不同特性的文件分层加密, 但是这些方案仍未有效地解决越权访问这一问题. 由于上述方案中同一层次均只能对一个文件进行加密, Li 等人[27]于 2019 年提出一种拓展的文件层次 CP-ABE (EFH-CP-ABE) 方案, 该方案允许在同一层次同时加密多个文件, 进一步提高了加解密效率.

2 预备知识

2.1 访问结构

假设 $\{P_1,P_2,\ldots,P_n\}$ 是一组属性的集合. 对 $\forall B,C$: 如果 $B\in\mathbb{A}$ 且 $B\subseteq C$, 那么 $C\in\mathbb{A}$, 则一个集合簇 $\mathbb{A}\subseteq 2^{\{P_1,P_2,\ldots,P_n\}}$ 是单调的. 假设 $\mathbb{A}$ 是 $\{P_1,P_2,\ldots,P_n\}$ 的非空子集, 即 $\mathbb{A}\subseteq 2^{\{P_1,P_2,\ldots,P_n\}}\backslash\{\varnothing\}$. 在 $\mathbb{A}$ 中的集合称为授权集合, 否则称为未授权集合.

2.2 双线性映射

设 $\mathbb{G}_0,\mathbb{G}_1$ 和 $\mathbb{G}_T$ 为 p 阶 (p 为大素数) 乘法循环群, $\mathbb{G}_0,\mathbb{G}_1$ 的生成元分别为 g_0,g_1. 若 $e:\mathbb{G}_0\times\mathbb{G}_0\to\mathbb{G}_T$ 为双线性映射, 则满足以下性质.

(1) 双线性: 对于任何的 $u\in\mathbb{G}_0$, $v\in\mathbb{G}_1$, $\forall a,b\in F_p$, $e\left(u^a,v^b\right)=e(u,v)^{ab}$.

(2) 非退化性: 存在 $u\in\mathbb{G}_0$, $v\in\mathbb{G}_1$ 使 $e(u,v)\neq 1$.

(3) 可计算性: $\forall u\in\mathbb{G}_0$, $v\in\mathbb{G}_1$, $e(u,v)$ 可被有效计算出来.

上述双线性映射 $e:\mathbb{G}_0\times G_1\to\mathbb{G}_T$ 称为非对称双线性映射, 若 $\mathbb{G}_0=\mathbb{G}_1$, 即映射为 $e:\mathbb{G}_0\times\mathbb{G}_1\to\mathbb{G}_T$, 则称为对称双线性映射.

2.3 DBDH 假设

挑战者随机选取 $a,b,c\in\mathbb{Z}_p$, $R\in\mathbb{G}_T$, 并计算 g^a,g^b,g^c. 对于 $\left(g,g^a,g^b,g^c\right)$, 敌手必须将有效的元组 $e(g,g)^{abc}$ 与随机元组 R 区分开来.

在 $\mathbb{G}_0$ 上, 若满足以下公式, 算法 $\mathcal{B}$ 输出一个猜测 $\mu\in\{0,1\}$, 解决 DBDH 问题的优势为 ϵ.

$$\left| \begin{array}{c} \Pr\left[\mathcal{B}\left(g,g^a,g^b,g^c,T=e(g,g)^{abc}\right)=0\right] \\ -\Pr\left[\mathcal{B}\left(g,g^a,g^b,g^c,T=R\right)=0\right] \end{array} \right| \geqslant \epsilon.$$

如果没有概率多项式时间 (PPT) 在解决 DBDH 难题上具有不可忽略的优势, 那么 DBDH 假设成立.

3 系统模型

3.1 分级用户访问树的构造

某企业设立若干部门, 如财务部、技术部、人力部等. 该企业在云环境中共享多个数据文件, 其中数据文件 M 包括公司的重大项目信息等绝密文件 m_1, 财务部的财务信息等机密文件 m_2, 技术部的技术资料等保密文件 m_3 等. 以 m_1 和 m_2 为例, 若采用 CP-ABE 方案对上述实际应用中的 m_1 和 m_2 进行加密, 则需构造不同的访问策略. 如图 1 所示, m_1 的访问结构设为 $\mathcal{T}_1$ {“董事”AND“股东”}, m_2 的访问结构设为 $\mathcal{T}_2$ {(“董事”AND“股东”)OR(“财务部” AND“经理”)}. 显然, 若采用 CP-ABE 方案, m_1 和 m_2 则需要分别通过访问结构 $\mathcal{T}_1$ 和 $\mathcal{T}_2$ 加密, 产生密文: $CT_1=\left\{\mathcal{T}_1,\widetilde{C}_1,C_1,\forall y\in Y_1:C_y,C_y'\right\}$, $CT_2=\left\{\mathcal{T}_2,\widetilde{C}_2,C_2,\forall y\in Y_2:C_y,C_y'\right\}$, 其中 $\widetilde{C}_i,C_i,C_y,C_y'$ 为相应的密文子项, 属性集合 Y_1={董事, 股东}, Y_2 = {董事, 股东, 财务部, 经理}.

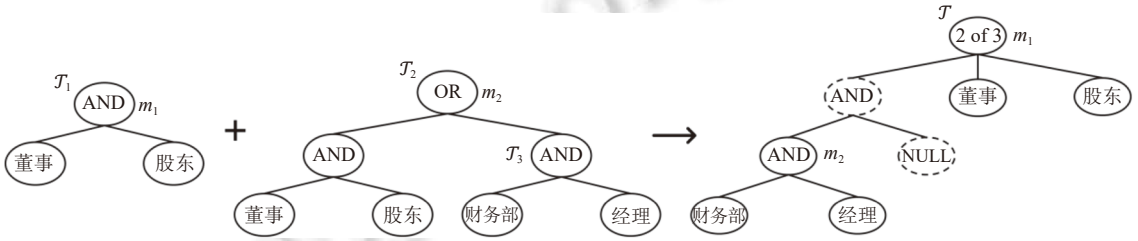


图 1 分级用户访问结构构造过程

文件 m_1 的安全等级高于文件 m_2, 访问结构 $\mathcal{T}_1$ 和 $\mathcal{T}_2$ 之间具有层次关系. 与 FH-CP-ABE 方案中的访问结构不同, 如果用户满足 $\mathcal{T}_1$ 则可访问 $\mathcal{T}_2$, 而用户满足 $\mathcal{T}_3$ 则不可访问 $\mathcal{T}_1$. 在实际应用中, 企业的规模越庞大, 数据文件的安全等级划分则越复杂, 采用 CP-ABE 方案消耗存储空间和计算时间也越大, 而现有的文件分层 CP-ABE 方案仍无法满足上述实际应用场景中分级用户的访问需求, 因此需构造新的分层访问结构.

本文方案引入控制节点和虚拟属性的概念, 将访问结构树 $\mathcal{T}_1$ 和 $\mathcal{T}_2$ 拓展为分级用户访问结构树 $\mathcal{T}$. 其中控制节点的门限为”AND”, 且用户属性集不包含虚拟属性. 低层次访问结构树 $\mathcal{T}_3$ 和一个虚拟属性作为控制节点的孩子节点构成一棵子树, 控制节点作为高层次访问结构树 $\mathcal{T}_1$ 的孩子节点与其他属性一同构成分级用户访问结构树 $\mathcal{T}$. 以控制节点为孩子节点的节点, 其门限值为 $(n-m)$ of n (n 为孩子节点个数, $0<m<n$), 当 $m=1$ 时该节点门限为“AND”, 当 $m=n-1$ 时该节点门限为“OR”. 如图 1 所示, 由于 $\mathcal{T}_1$ 的根节点门限为“AND”, 因此其门限值设置为 2 of 3.

3.2 分级用户访问树的相关定义

设 $\mathcal{T}$ 是根节点为 R 的用户分级访问树, 其被划分为 k 个访问层次. $\mathcal{T}$ 中的一个节点表示为 x, 根节点为 x 的子树表示为 $\mathcal{T}_x$. 若 x 是叶子节点, 则表示属性; 若 x 是非叶子节点, 则表示门限. $\mathcal{T}$ 中非叶子节点 x 的子节点数目表示为 num_x, k_x 表示该节点的门限值, 其中 $1\leqslant k_x\leqslant num_x$, $k_x=1$ 表示“OR”门, $k_x=num_x$ 表示“AND”门, 叶子节点的门限值 $k_x=1$. 特别地, 若节点 x 的孩子节点包含控制节点, 那么 $k_x=num_x-1$ 表示“AND”门. L_i $(1\leqslant i\leqslant k)$ 表示 $\mathcal{T}$ 的层次节点. $\mathcal{T}$ 中节点 x 的父节点表示为 $parent(x)$. 与 $\mathcal{T}$ 中的叶子节点 x 相关联的属性表示为 $att(x)$. $index(x)$ 返回一个与节点 x 相关联的唯一值, 对于给定的密钥, 该值以任意方式赋给 x. 如果节点 x 的子节点至少包含一个门限, 则节点 x 为传输节点. $TN-CT(x)$ 表示传输节点 x 的门限子节集合, 表示为 $TN-CT(x)=\left\{ch_1,ch_2,\ldots,ch_j\right\}$. 图 2 为一个两层分级用户访问树, 其中节点 B,E,F 为属性 (E 为虚拟属性); 节点 R, A 为传输节点 (A 为控制节点), $TN-CT(A)=\{D\}$. 由于节点 R 的孩子节点中包含控制节点 A, 因此其门限为 2 of 3.

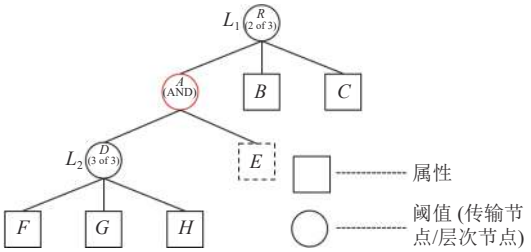


图 2　两层分级用户访问树示例

$\mathcal{T}_x(\gamma)=1$ 表示用户属性集合 γ 满足子树 $\mathcal{T}_x$. $\mathcal{T}_x(\gamma)$ 按如下方式进行递归计算. 若 x 表示非叶子节点, 计算节点 x 的所有孩子节点 x' 的 $\mathcal{T}_{x'}(\gamma)$, 当且仅当至少 k_x 个孩子节点返回 1 时, $\mathcal{T}_x(\gamma)=1$; 若 x 表示叶子节点, 当 $x\in\gamma$ 时, $\mathcal{T}_x(\gamma)=1$.

3.3　系统模型

本文方案的系统模型如图 3 所示. 该系统模型由 4 个实体组成: 授权机构 (CA)、云服务提供商 (CSP)、数据所有者 (DO) 和用户 (DU). 详细定义如下.

授权机构 (CA): 授权机构 CA 是完全可信的实体, 它能够诚实地执行相应的操作并给用户返回结果. CA 负责执行初始化算法 *Setup* 和密钥生成算法 *KeyGen*, 生成系统公钥和主密钥, 并为系统中的用户生成私钥.

云服务提供者 (CSP): 云服务提供者 CSP 是半可信的实体, 它可以诚实地执行分配的任务并返回正确的结果, 但云服务提供者是诚实且好奇的, 它希望从分配的任务中找出尽可能多的敏感内容. CSP 负责提供密文存储和传输服务.

数据所有者 (DO): 数据所有者有大量的数据需要在云端存储和共享, 其负责定义与密文相关的访问策略并执行加密 *Encrypt* 算法, 并将生成的密文上传至云服务提供商 CSP.

用户 (DU): 用户需要在云端访问大量数据. 其下载相应的密文, 并使用其私钥执行解密算法 *Decrypt* 进行解密.

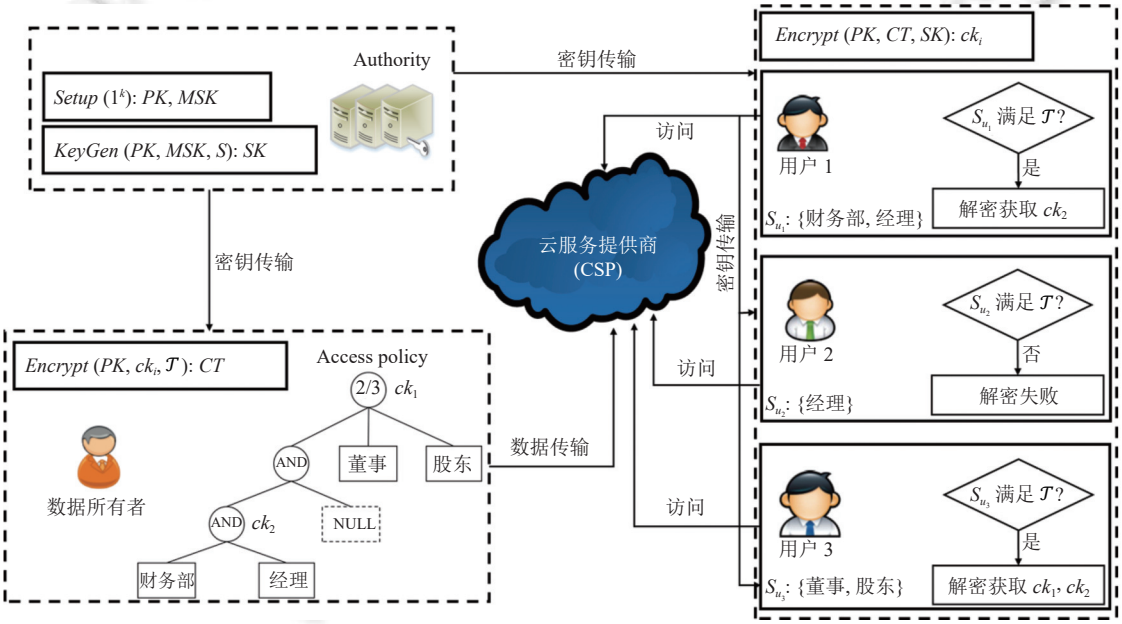


图 3　云计算中数据共享示例

3.4 算法定义

本文方案由 *Setup*, *KeyGen*, *Encrypt*, *Decrypt* 这 4 个算法构成, 详细描述如下.

(1) $Setup\left(1^k\right)\to(PK,MSK)$. CA 执行该算法. 输入为安全参数 k, 输出为系统公钥 PK 和主密钥 MSK.

(2) $KeyGen(PK,MSK,S)\to(SK)$. CA 执行该算法. 输入为 PK 和 MSK 及属性集合 S, 输出为用户私钥 SK.

(3) $Encrypt(PK,ck,\mathbb{A})\to(CT)$. DO 执行该算法. 输入为 PK, 内容密钥 $ck=\{ck_1,\ldots,ck_i\}$ 以及分级用户访问结构 $\mathbb{A}$, 输出为密文 CT.

(4) $Decrypt(SK,PK,CT)\to(ck_i(i\in[1,k]))$. DU 执行该算法. 输入为 SK, PK 和 CT. 如果用户属性 S 满足部分或全部访问结构 $\mathbb{A}$, 则输出部分或全部内容密钥 ck_i. 用户使用对称解密算法对内容密钥对应的文件 m_i 进行解密.

图 3 描述了 DO 在云环境中共享两个分层文件 m_1 和 m_2 的实际应用. 首先, 使用随机选择的内容密钥 ck_1 和 ck_2 分别加密这两个文件, 并生成密文 $\{E_{ck_1}(m_1)\}$ 和 $\{E_{ck_2}(m_2)\}$. 然后, 根据对应分级用户访问树 $\mathcal{T}$, 对内容密钥 ck_1 和 ck_2 进行加密, 具体过程如下: 首先, CA 通过 *Setup* 算法生成 PK 和 MSK. 其次, 通过 *KeyGen* 算法为每个用户创建 SK. 然后, DO 基于访问策略 $\mathcal{T}$, 使用 *Encrypt* 算法对内容密钥 $ck=\{ck_1,ck_2\}$ 进行加密, 生成一个整合的密文 CT, 并上传至 CSP. 如果用户需要访问云端的文件, 则从 CSP 中下载密文 $\{E_{ck_1}(m_1)\},\{E_{ck_2}(m_2)\}$ 和 CT. 在图 3 中, 用户 1 可以解密 ck_2, 得到 m_2; 用户 2 无法解密任何文件; 用户 3 可以解密 ck_1 和 ck_2, 得到 m_1 和 m_2.

3.5 安全模型

本节为本文方案定义了针对性 (selective) 选择明文攻击 CPA 安全模型, 即 sCPA.

准备: 敌手选择 $\mathbb{T}^*$ 作为挑战访问结构, 并将 $\mathbb{T}^*$ 发送给挑战者.

初始化: 挑战者执行 *Setup* 算法, 并将生成的公钥 PK 发送给敌手, 而系统主密钥 MSK 由挑战者秘密保存.

查询阶段 1: 敌手选择属性集合 $S_1,\ldots,S_u$ ($\forall i\in[1,\ldots,u]$, $S_i\notin\mathbb{T}^*$), 并重复向挑战者查询 SK. 挑战者运行 *KeyGen* 算法, 将生成的私钥 PK 发送给敌手.

挑战: 敌手向挑战者提交两个等长的消息 $\mathcal{M}_0$ 和 $\mathcal{M}_1$. 挑战者选择一个随机值 $b\in\{0,1\}$, 并使用挑战的访问结构 $\mathbb{T}^*$ 加密 $\mathcal{M}_b$ 生成密文. 挑战者将密文 CT^* 发送给敌手.

查询阶段 2: 敌手重复阶段 1 的查询, 但是敌手获取的私钥 SK 并不满足挑战访问结构 $\mathbb{T}^*$.

猜测: 敌手随机输出一个猜测 $b'\in\{0,1\}$.

在该游戏中, 敌手赢得该安全游戏的优势被定义为 $\Pr[b'=b]-1/2$.

4 方案构造

4.1 方案的具体构造

本文方案以文献 [3] 为基础, 由 4 个部分组成: 初始化、私钥生成、加密和解密. 详细算法如下.

$\mathbb{G}_0$ 是一个 p 阶乘法循环群, 生成元为 g. 令 $e:\mathbb{G}_0\times\mathbb{G}_0\to\mathbb{G}_T$ 为双线性映射. 对 $\forall i\in\mathbb{Z}_p$ 和属性集 $S=\{S_1,S_2,\ldots,S_k\in\mathbb{Z}_p\}$, 拉格朗日系数为 $\Delta_{k,S}=\prod_{i\in S,i\neq j}(x-j)(i-j)$. 定义哈希函数 $H_1:\{0,1\}^*\to\mathbb{G}_0$ 和 $H_2:\mathbb{G}_T\to\mathbb{Z}_p$.

(1) 初始化: $Setup\left(1^k\right)$

算法输入系统安全参数 k, 并选择随机数 $\alpha,\beta\in\mathbb{Z}_p$. 输出为公钥 $PK=\left\{\mathbb{G}_0,g,h=g^\beta,e(g,g)^\alpha\right\}$ 和主密钥 $MSK=\{\beta,g^\alpha\}$.

(2) 私钥生成: $KeyGen(PK,MSK,S)$

该算法输入 PK、MSK 和与用户相关联的属性集合 $S\left(S\subseteq\widetilde{A}\right)$. 为每个用户选择随机数 $r\in\mathbb{Z}_p$, 同时为用户每个属性 $j\in S$ 随机选择 $r_j\in\mathbb{Z}_p$. 生成用户私钥 $SK=\left\{D=g^\alpha\cdot h^r,\ \forall j\in S,\ D_j=g^r\cdot H_1(j)^{r_j},\ D'_j=h^{r_j}\right\}$.

(3) 加密: $Encrypt(PK,ck,\mathcal{T})$

该算法输入 PK, 内容密钥 $ck=\{ck_1,\ldots,ck_k\}$ 和分级用户访问树 $\mathcal{T}$. 输出一个整合的密文 CT.

在 $\mathcal{T}$ 中, 数据所有者从根节点向下设定 k 个层次节点, 分别对应 $ck_1,\ldots,ck_k$. 然后, 随机选择 $s_1,s_2,\ldots,s_k\in\mathbb{Z}_p$, 对所有的层次节点, 计算 $\widetilde{C}_i=ck_ie(g,g)^{\alpha s_i}$ 和 $C'_i=g^{s_i}$.

多项式构造规则: 首先从根节点R开始, 为$\mathcal{T}$的每个节点x(包括叶子节点)以自上而下的方式选取一个多项式q_x. 对$\mathcal{T}$中的每个节点x, 多项式的次数d_x设为k_x-1, 其中k_x为门限值. 从根节点R开始, 数据所有者设置$q_R(0)=s_1$, 并选择d_R个剩余节点对其进行完全定义. 这些节点包括两种类型: 一种是根节点R的孩子节点中包含的层次节点, 另一种是随机选择的其他节点. 对于这些节点, 若节点x为层次节点, 设$q_x(0)=q_{L_i}(0)=s_i$; 否则, $q_x(0)=q_{parent(x)}(index(x))$. 此外, 多项式$q_x$剩余的$d_x$个节点由该节点的孩子节点包含的层次节点和随机选择的节点组成.

设Y是$\mathcal{T}$中的叶子节点集合, 令$\forall y\in Y$, 计算$C_y=h^{q_x(0)}$和$C'_y=H_1(att(y))^{q_x(0)}$.

在$\mathcal{T}$中, 令传输节点x的集合为X, 传输节点x的子节点的门限集合为$TN-CT(x)=\{ch_1,\ldots,ch_j,\ldots\}$. 为每个$TN-CT(x)$中的节点$ch_j$, 选择随机值$\phi_j\in\mathbb{Z}_p$. 然后计算$\widehat{C}_{x,j}=e(g,g)^{\alpha\left(q_{ch_j}(0)+H_2\left(e(g,g)^{\alpha q_x(0)}\right)\right)}\cdot e(g,g)^{\phi_j H_2\left(e(g,g)^{\alpha q_x(0)}\right)}$和$\breve{C}_{x,j}=g^{\alpha+\phi_j}$

数据所有者输出整合的密文$CT=\left\{\mathcal{T},\widetilde{C}_i,C'_i,C_y,C'_y,\widehat{C}_{x,j},\breve{C}_{x,j}\right\}$.

(4) 解密: $Decrypt(PK,CT,SK)$

该算法输入PK、分级用户访问树$\mathcal{T}$相关联的密文CT和用户属性集合S相关联的SK. 首先定义一个递归算法$DecryptNode(CT,SK,x)$.

如果x为叶子节点, 令$i=attr(x)$, 并定义$DecryptNode(CT,SK,x)$如下:

如果$i\notin S$, $DecryptNode(CT,SK,x)=\perp$.

如果$i\in S$, 计算$DecryptNode(CT,SK,x)$如下:

$$DecryptNode(CT,SK,x)=\frac{e(D_i,C_i)}{e\left(D'_i,C'_i\right)}=\frac{e\left(g^r\cdot H_1(i)^{r_i},g^{\beta q_x(0)}\right)}{e\left(g^{\beta r_i},H_1(att(i))^{q_x(0)}\right)}=e(g,g)^{r\beta q_x(0)}.$$

如果x为非叶子节点, $DecryptNode(CT,SK,x)$定义如下: 对于节点x中的所有孩子节点z, 其返回$DecryptNode(CT,SK,z)$并存储输出F_z. 设S_x为任意k_x-size个孩子节点z的集合.

如果集合S_x不存在, 则$DecryptNode(CT,SK,z)=\perp$.

否则, 计算F_x如下, 其中$S'_x=\{i=index(z),z\in S_x\}$.

$$F_x=\prod_{z\in S_x}F_z^{i,S'_x(0)}=\prod_{z\in S_x}\left(e(g,g)^{arq_z(0)}\right)^{i,S'_x(0)}=\prod_{z\in S_x}\left(e(g,g)^{arq_x(i)}\right)^{i,S'_x(0)}=e(g,g)^{r\beta q_x(0)}.$$

如果属性集合S满足全部或部分分级用户访问树$\mathcal{T}$, 即S满足全部或部分层次节点, 则$e(g,g)^{r\beta s_i}(i\in[1,k])$可通过如下公式递归计算得到:

$$A_i=DecryptNode(CT,SK,L_i)=e(g,g)^{r\beta q_x(0)}=e(g,g)^{r\beta s_i}.$$

通过如下公式进一步计算$e(g,g)^{\alpha s_i}$:

$$F_i=\frac{e\left(C'_i,D\right)}{A_i}=\frac{e(g^{s_i},g^{\alpha}\cdot h^r)}{e(g,g)^{r\beta s_i}}=e(g,g)^{\alpha s_i}.$$

基于层次节点, 如果属性集合S包含较低层次的授权节点, 则可利用传输节点相关的密文子项, 通过如下公式计算所有授权的层次节点的值.

$$F_{i+1,j}=\frac{\widehat{C}_{x,j}}{e\left(\breve{C}_{x,j},g\right)^{H_2(F_i)}}=\frac{e(g,g)^{\alpha\left(q_{ch_j}(0)+H_2\left(e(g,g)^{\alpha q_x(0)}\right)\right)}\cdot e(g,g)^{\phi_j H_2\left(e(g,g)^{\alpha q_x(0)}\right)}}{e(g^{\alpha+\phi_j},g)^{H_2(e(g,g)^{\alpha s_i})}}=e(g,g)^{\alpha q_{ch_j}(0)}.$$

因此可依次计算获得$F_{i+1,j},\ldots,F_{k,j}$, 即$e(g,g)^{\alpha s_i},\ldots,e(g,g)^{\alpha s_k}$. 然后, 可计算这些节点对应的内容密钥$\{ck_1,\ldots,ck_k\}$可通过以下公式计算:

$$\frac{\widetilde{C}_i}{F_i}=\frac{ck_i e(g,g)^{\alpha s_i}}{e(g,g)^{\alpha s_i}}=ck_i,\ i\in[1,k].$$

最后, 使用内容密钥$\{ck_1,\ldots,ck_k\}$, 利用对称解密算法依次对数据文件$\{m_i,\ldots,m_k\}$进行解密.

4.2 方案构造说明

如何在用户之间仅存在等级关系, 而属性之间不存在包含关系的情形下实现多文件加密是本文需要解决的一大难题. 本文方案首先引入控制节点作为一类特殊的传输节点, 其继承了传输节点的作用, 即高等级用户可通过传输节点向下解密低等级文件; 其次, 如图 4 中 $\mathcal{T}_1$ 所示, 若按照 FH-CP-ABE 的构造方式, 显然满足“财务部”和“经理”的用户仍然有向上访问高等级文件的可能性, 而通过控制节点和虚拟属性相结合, 构造分级用户访问树 $\mathcal{T}_2$, 则可消除低等级用户向上访问的可能性. $\mathcal{T}_2$ 中由于存在控制节点和虚拟属性, 有效地控制了不同等级用户访问分层文件的权限, 实现了 FH-CP-ABE 方案无法完成的分级用户访问.

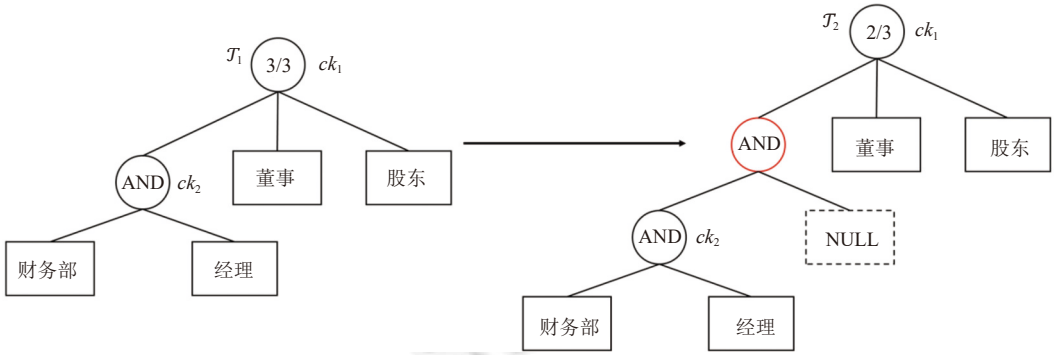


图 4 方案构造说明示例

5 安全性分析

5.1 安全性证明

定理 1. 假设 DBDH 困难问题成立, 那么不存在概率多项式时间敌手可选择性地攻破本文所提方案.

在本文方案的选择性安全游戏中, 假设存在敌手 $\mathcal{A}$ 有不可忽视的优势 ϵ 攻破本文方案的构造. 构造一个模拟器 $\mathcal{B}$, 它能够以 $\epsilon/2$ 的优势解决 DBDH 困难问题. $\mathbb{G}_0$ 的一个生成元为 g, 双线性映射为 $e:\mathbb{G}_0\times\mathbb{G}_0\to\mathbb{G}_T$.

证明: 挑战者 C 随机选择阶为 p 的群 $\mathbb{G}_0$ 和 $\mathbb{G}_T$, 双线性映射: $e:\mathbb{G}_0\times\mathbb{G}_0\to\mathbb{G}_T$. 随机选择 $a,b,c,c'\in\mathbb{Z}_p$, $b\in\{0,1\}$, $\langle A,B,C\rangle=\left\langle g^a,g^b,g^c\right\rangle$, 生成元 $g\in\mathbb{G}_0$ 以及一个随机参数 R, $R'\in\mathbb{G}_T$. 若 $b=0$, C 设 $T=e(g,g)^{abc}$, $T'=e(g,g)^{abc'}$, 否则, $T=R$, $T'=R'$. 为更为清晰地描述安全性证明, 以下假设只加密两个层次文件.

准备: 敌手 $\mathcal{A}$ 向模拟器 $\mathcal{B}$ 提交一个挑战访问结构 $\mathbb{T}^*$, 并将 $\mathbb{T}^*$ 发送给挑战者 C.

初始化: 模拟器 $\mathcal{B}$ 选择一个随机数 $\alpha'\in\mathbb{Z}_p$, 并记 $\alpha=\alpha'+ab$. 模拟器 $\mathcal{B}$ 计算 $e(g,g)^{\alpha}=e(g,g)^{\alpha'}\cdot e(g,g)^{ab}$. 同时, 设 $h=g^{\beta}=B=g^b$. 模拟器 $\mathcal{B}$ 将 PK 发送给敌手 $\mathcal{A}$.

查询阶段 1: 敌手 $\mathcal{A}$ 向模拟器 $\mathcal{B}$ 提交一个属性集合 $S_1,\ldots,S_u$ (任意集合 S_i 均不满足 $\mathbb{T}^*$), 模拟器 $\mathcal{B}$ 选择一个随机数 $r'\in\mathbb{Z}_p$, 并设 $r=r'-a$. 模拟器 $\mathcal{B}$ 可获得 $D=g^{\alpha}\cdot h^r=g^{\alpha}\cdot g^{\beta r}=g^{\alpha'+ab}\cdot g^{\beta r}=g^{\alpha'+ab}\cdot g^{\beta(r'-a)}=g^{(\alpha'+r'b)}$. 对每个属性 $j\in S_i$, 模拟器 $\mathcal{B}$ 选择随机数 $r_j\in\mathbb{Z}_p$. 计算私钥: $D_j=g^r\cdot H_1(j)^{r_j}=g^{r'-a}\cdot H_1(j)^{r_j}=g^{r'}/A\cdot H_1(j)^{r_j}, D'_j=h^{r_j}=g^{br_j}=B^{r_j}$. 最后, 模拟器 $\mathcal{B}$ 将产生的 SK 发送给敌手 $\mathcal{A}$.

挑战: 敌手 $\mathcal{A}$ 向模拟器 $\mathcal{B}$ 提交两个等长的消息 $\mathcal{M}_0$ 和 $\mathcal{M}_1$. 模拟器 $\mathcal{B}$ 将 $\mathcal{M}_0$, $\mathcal{M}_1$ 提交给挑战者 C. 挑战者 C 选择一个随机值 $b\in\{0,1\}$, 使用挑战的访问结构加密 $\mathcal{M}_b$, 产生的密文子项为: $C'_i=g^{s_i}=g^c=C$, $\widetilde{C}_i=\mathcal{M}_b e(g,g)^{\alpha s_i}=\mathcal{M}_b e(g,g)^{\alpha c}=\mathcal{M}_b\cdot Te(g,g)^{\alpha' c}$, $\forall y\in Leaf_{\mathbb{T}^*}: C_y, C'_y$. 对于 $\forall ch_j\in TN-CT(x)$, 模拟器 $\mathcal{B}$ 随机选择 $\phi_j\in\mathbb{Z}_p$, 计算 $\overline{C}=\dfrac{\widetilde{C}_1}{\mathcal{M}_0}$, $\widehat{C}_{x,j}=e(g,g)^{\alpha c'}\cdot e(g,g)^{\alpha H_2(\overline{C})}\cdot e(g,g)^{\phi_j H_2(\overline{C})}=e(g,g)^{\alpha' c'}\cdot T'\cdot e(g,g)^{(\alpha'+ab+\phi_j)H_2(\overline{C})}, \breve{C}_{x,j}=g^{\alpha'+ab+\phi_j}$. 模拟器 $\mathcal{B}$ 将 $CT^*=\left\{\mathbb{T}^*,\widetilde{C}_i,C'_i,C_y,C'_y,\widehat{C}_{x,j},\breve{C}_{x,j}\right\}$ 发送给敌手 $\mathcal{A}$.

查询阶段 2: 敌手 $\mathcal{A}$ 重复阶段 1 的查询, 但是敌手 $\mathcal{A}$ 获取的私钥 SK 不满足挑战访问结构 $\mathbb{T}^*$.

猜测: 敌手$\mathcal{A}$输出b的猜想为b', 那么模拟器$\mathcal{B}$输出的猜测也是b'. 当$b=0$时, 模拟器$\mathcal{B}$给出了一个准确的模拟, 即$T=e(g,g)^{abc}$, $T'=e(g,g)^{abc'}$, 则可计算$\overline{C}=\dfrac{\widetilde{C}_i}{\mathcal{M}_0}=Te(g,g)^{\alpha' c}$, $\widehat{C}_{x,j}=e(g,g)^{\alpha c'}\cdot e(g,g)^{\alpha H_2(\overline{C})}\cdot e(g,g)^{\phi_j H_2(\overline{C})}=e(g,g)^{\alpha' c'}\cdot T'\cdot e(g,g)^{(\alpha'+ab+\phi_j)H_2\left(Te(g,g)^{\alpha' c}\right)}$. 此时, 输出即为本文方案的完整密文. 当$b=1$时, 表示$T$, T'为$\mathbb{G}_T$群中的随机元素, 则消息$\mathcal{M}_b$对敌手$\mathcal{A}$完全隐藏, 因此无法计算出$\overline{C}$和$\widehat{C}_{x,j}$. 敌手$\mathcal{A}$在此安全游戏中的优势为: $\frac{1}{2}\Pr[b'=b|b=0]+\frac{1}{2}\Pr[b'=b|b=1]-\frac{1}{2}=\frac{1}{2}\left(\frac{1}{2}+\epsilon\right)+\frac{1}{2}\cdot\frac{1}{2}-\frac{1}{2}=\frac{1}{2}\epsilon$.

通过以上描述, 不存在概率多项式时间敌手$\mathcal{A}$以不可忽略的优势攻破本文方案的构造, 因此本文方案在DBDH假设下能达到sCPA安全.

5.2 安全性分析

为保证方案的安全性, 层次节点的秘密值$q_{child_j}(0)$(在解密过程中以$e(g,g)^{\alpha q_{child_j}(0)}$的形式被计算)只允许用户通过高层次节点的秘密值和该节点的孩子节点的密文子项计算得到, 而在FH-CP-ABE方案中, 密文子项$\widehat{C}_{(x,y),j}=e(g,g)^{\alpha\left(q_{(x,y)}(0)+q_{child_j}(0)\right)}\cdot H_2\left(e(g,g)^{\alpha q_{(x,y)}(0)}\right)$的构造由于将哈希运算后的值直接作为乘数, 导致$q_{child_j}(0)$可计算, 进而解密出所有层次节点, 显然这一密文子项的构造存在极大的安全风险. 为避免该问题, 本文方案的解决思路为构造相关的密文子项仅允许用户通过方案中的方式从高向低进行解密, 而无法通过获取的密文子项进行计算. 首先采用随机化方法, 在构造密文子项$\widehat{C}_{x,j}$时, 为$TN-CT(x)$中的每一个节点选择一个随机值ϕ_j, 利用其与哈希运算$H_2\left(e(g,g)^{\alpha q_x(0)}\right)$的值相乘进行随机化, 并将乘积隐藏在双线性映射的指数上, 使得$TN-CT(x)$中的每一个节点相关的密文子项$\widehat{C}_{x,j}$都无法消去哈希运算后的值, 从而无法获取层次节点的秘密值. 其次, 即使同一层次的用户进行合谋, 在访问结构中存在控制节点和虚拟属性, 仍可控制其无法向上进行解密. 通过上述方法, 用户既无法使用密文子项通过除法运算获取同一层次的节点的秘密值, 同时也无法通过合谋向上访问, 因此消除了越权访问的可能性.

6 性能分析

6.1 特性分析

本文方案实现了分级用户访问的多文件加密, 且解决了越权访问的安全问题, 表1为本文方案和文献[3]、文献[19]的特性对比. 由表1可以看出: 虽然本文方案和文献[19]与文献[3]对比, 均可实现多文件加密, 然而文献[19]无法实现用户分级访问, 且存在越权访问的安全问题, 而本文方案既满足用户分级访问, 又避免了越权访问的风险.

表1 特性分析

文献	多文件加密	用户分级访问	越权访问
文献[3]	×	×	×
文献[19]	√	×	√
本文	√	√	×

6.2 理论分析

本文方案将与文献[3]、文献[19]进行计算开销和存储开销的对比. 首先定义理论分析中使用的符号, 如表2所示. 假设群中的指数运算为E(其中在群$\mathbb{G}_0$和$\mathbb{G}_T$进行一次指数运算所需要的时间分别表示为$E_{\mathbb{G}_0}$和$E_{\mathbb{G}_T}$); 双线性配对运算为B_e; 用户的属性为$|\mathbb{A}_u|$; 密文CT中的属性为$\left|\mathbb{A}_{c_i}\right|$; 传输节点的集合为$|\mathbb{N}_T|$; $|S_i|$为满足访问结构的最小内部节点(包括根节点); $\mathbb{Z}_p,\mathbb{G}_0,\mathbb{G}_T$中元素的长度分别为$L_{\mathbb{Z}_p},L_{\mathbb{G}_0},L_{\mathbb{G}_T}$. 另外, 对于每个传输节点$x$, 设$TN-CT(x)$包含$j$个节点, 即$TN-CT(x)=\{ch_1,\ldots,ch_j\}$. 假设数据所有者有$k$个文件$M=\{m_1,\ldots,m_k\}$需要加密, 其中$k$个文件按照访问规则被分为$k$个依次递减的访问等级, 与密文相关的属性表示为$\mathbb{A}_{c_i}=\{\mathbb{A}_{c_1},\mathbb{A}_{c_2},\ldots,\mathbb{A}_{c_k}\}(\mathbb{A}_{c_1},$

$\mathbb{A}_{c_2},\ldots,\mathbb{A}_{c_k}$ 表示安全等级依次递减的与密文相关的属性), 且 $\exists i,j\in k\ (i\neq j)$, $\mathbb{A}_{c_i}\not\subset\mathbb{A}_{c_j}$. 同样地, 满足解密条件的最少内部节点表示为 $S_i=\{S_1,S_2,\ldots,S_k\}$.

表 2 符号定义

符号	定义	符号	定义
$E_{\mathbb{G}_0}$	群 $\mathbb{G}_0$ 上的指数运算	$\mathbb{N}_T$	传输节点集合
$E_{\mathbb{G}_T}$	群 $\mathbb{G}_T$ 上的指数运算	S_i	满足访问结构的最小内部节点
B_e	双线性配对运算	$L_{\mathbb{Z}_p}$	$\mathbb{Z}_p$ 上元素的长度
$\mathbb{A}_u$	用户 u 的属性集合	$L_{\mathbb{G}_0}$	$\mathbb{G}_0$ 上元素的长度
$\mathbb{A}_{c_i}$	密文 CT 中的属性	$L_{\mathbb{G}_T}$	$\mathbb{G}_T$ 上元素的长度

本文方案和文献 [3]、文献 [19] 的性能对比如表 3 所示, 其中不包括访问结构 $\mathcal{T}$、哈希函数计算以及简单的乘法计算的分析.

表 3 性能对比

指标	文献[3]	文献[19]	本文方案
加密时间	$\left[2\left(\lvert\mathbb{A}_{c_1}\rvert+\ldots+\lvert\mathbb{A}_{c_k}\rvert\right)+k\right]E_{\mathbb{G}_0}+2kE_{\mathbb{G}_T}$	$\left(2\lvert\mathbb{A}_{c_1}\rvert+k\right)E_{\mathbb{G}_0}+(j\lvert\mathbb{N}_T\rvert+k)E_{\mathbb{G}_T}$	$\left(2\lvert\mathbb{A}_{c_1}\rvert+j\lvert\mathbb{N}_T\rvert+k\right)E_{\mathbb{G}_0}+(2j\lvert\mathbb{N}_T\rvert+k)E_{\mathbb{G}_T}$
解密时间	$2k(\lvert\mathbb{A}_u\rvert+1)B_e+2(\lvert S_1\rvert+\ldots+\lvert S_k\rvert)E_{\mathbb{G}_T}$	$(2\lvert\mathbb{A}_u\rvert+1)B_e+(\lvert S_1\rvert+j\lvert\mathbb{N}_T\rvert+k)E_{\mathbb{G}_T}$	$(2\lvert\mathbb{A}_u\rvert+j\lvert\mathbb{N}_T\rvert+1)B_e+(\lvert S_1\rvert+j\lvert\mathbb{N}_T\rvert+k)E_{\mathbb{G}_T}$
PK长度	$3L_{\mathbb{G}_0}+L_{\mathbb{G}_T}$	$3L_{\mathbb{G}_0}+L_{\mathbb{G}_T}$	$3L_{\mathbb{G}_0}+L_{\mathbb{G}_T}$
MSK长度	$L_{\mathbb{Z}_p}+L_{\mathbb{G}_0}$	$L_{\mathbb{Z}_p}+L_{\mathbb{G}_0}$	$L_{\mathbb{Z}_p}+L_{\mathbb{G}_0}$
SK长度	$(2\lvert\mathbb{A}_u\rvert+1)L_{\mathbb{G}_0}$	$(2\lvert\mathbb{A}_u\rvert+1)L_{\mathbb{G}_0}$	$(2\lvert\mathbb{A}_u\rvert+1)L_{\mathbb{G}_0}$
CT长度	$\left[2\left(\lvert\mathbb{A}_{c_1}\rvert+\ldots+\lvert\mathbb{A}_{c_k}\rvert\right)+k\right]L_{\mathbb{G}_0}+kL_{\mathbb{G}_T}$	$\left(2\lvert\mathbb{A}_{c_1}\rvert+k\right)L_{\mathbb{G}_0}+(j\lvert\mathbb{N}_T\rvert+k)L_{\mathbb{G}_T}$	$\left(2\lvert\mathbb{A}_{c_1}\rvert+j\lvert\mathbb{N}_T\rvert+k\right)L_{\mathbb{G}_0}+(2j\lvert\mathbb{N}_T\rvert+k)L_{\mathbb{G}_T}$

6.2.1 计算性能分析

数据所有者加密 k 个层次文件时, 关于层次节点的密文子项的计算开销为 $kE_{\mathbb{G}_0}+kE_{\mathbb{G}_T}$; 对于分级用户访问树的每个叶子节点 (包括 $k-1$ 个虚拟属性), 需要进行两次指数运算; 关于传输节点的密文子项的计算开销为 $j|\mathbb{N}_T|E_{\mathbb{G}_0}+2j|\mathbb{N}_T|E_{\mathbb{G}_T}$, 因此加密数据的计算开销为 $\left(2\lvert\mathbb{A}_{c_1}\rvert+j|\mathbb{N}_T|+k\right)E_{\mathbb{G}_0}+(2j|\mathbb{N}_T|+k)E_{\mathbb{G}_T}$. 用户在解密时, 解密叶子节点需要进行两次双线性配对运算, 故计算开销为 $2|\mathbb{A}_u|B_e$; 解密非叶子节点的计算开销为 $(|S_1|+k)E_{\mathbb{G}_T}$; 计算 F_i 的开销为 B_e; 解密层次节点需 $j|\mathbb{N}_T|B_e+j|\mathbb{N}_T|E_{\mathbb{G}_T}$, 因此本文方案解密数据文件的计算开销为 $(2|\mathbb{A}_u|+j|\mathbb{N}_T|+1)B_e+(|S_1|+j|\mathbb{N}_T|)E_{\mathbb{G}_T}$. 由表 3 可观察到在本文方案中, 当 $TN-CT(x)$ 中节点的个数 j 固定不变时, 随着文件的数目 k 的增加, 加密的计算开销增长率为 $E_{\mathbb{G}_0}+E_{\mathbb{G}_T}$, 解密的计算开销的增长率为 $E_{\mathbb{G}_T}$; 当文件的数目 k 固定不变时, 随着 $TN-CT(x)$ 中的节点个数 j 的增加, 加密的计算开销增长率为 $|\mathbb{N}_T|E_{\mathbb{G}_0}+2|\mathbb{N}_T|E_{\mathbb{G}_T}$, 解密的计算开销增长率为 $|\mathbb{N}_T|E_{\mathbb{G}_T}$. 由此可看出, 由于本文方案使用一个整合的用户分级访问树, 因此在加密和解密的计算开销上都远远小于文献 [3]. 为实现用户分级访问, 同时避免越权访问的风险, 本文在访问树中引入了虚拟属性, 并重构了关于传输节点的密文子项. 因此本文方案是在牺牲一部分计算效率的情况下, 保证了本文方案的分级访问和安全性. 在相同条件下, 本文方案的与文献 [19] 相比, 加密的计算开销增加了约 $[j|\mathbb{N}_T|+2(k-1)]E_{\mathbb{G}_0}+j|\mathbb{N}_T|E_{\mathbb{G}_T}$, 解密的计算开销增加了约 $j|\mathbb{N}_T|B_e$. 从整体上看, 在保证了本文方案分级访问的特性及安全性的前提下, 稍微牺牲部分计算效率仍是值得的.

6.2.2 存储性能分析

在存储开销方面, 本文方案与文献 [3]、文献 [19] 的公钥 PK, 主私钥 MSK 和用户私钥 SK 的大小相同, 分别

为 $3L_{\mathbb{G}_0}+L_{\mathbb{G}_T}, L_{\mathbb{Z}_p}+L_{\mathbb{G}_0}, (2|\mathbb{A}_u|+1)L_{\mathbb{G}_0}$. 同时, 在本文方案中, 对于 k 个层次文件, 密文子项的存储开销为 $kL_{\mathbb{G}_0}+kL_{\mathbb{G}_T}$; 对于分级用户访问树的每个叶子节点 (包括 $k-1$ 个虚拟属性), 密文子项的存储开销为 $2\left|\mathbb{A}_{c_1}\right|L_{\mathbb{G}_0}$; 对于传输节点, 密文子项的存储开销为 $j|\mathbb{N}_T|L_{\mathbb{G}_0}+2j|\mathbb{N}_T|L_{\mathbb{G}_T}$, 因此本文方案密文 CT 的存储开销为 $\left(2\left|\mathbb{A}_{c_1}\right|+j|\mathbb{N}_T|+k\right)L_{\mathbb{G}_0}+(2j|\mathbb{N}_T|+k)L_{\mathbb{G}_T}$. 从表 3 中可看出, 当 j 固定不变时, 随着文件的数目 k 的增加, 加密的存储开销的增长率为 $L_{\mathbb{G}_T}+L_{\mathbb{G}_0}$; 当文件的数目 k 固定不变时, 随着 $TN-CT(x)$ 中的节点个数 j 的增加, 加密的存储开销的增长率为 $|\mathbb{N}_T|L_{\mathbb{G}_0}+2|\mathbb{N}_T|L_{\mathbb{G}_T}$. 在同样的条件下, 本文方案的密文大小远远小于文献 [3]. 同样地, 相同条件下与文献 [19] 相比, 本文方案由于虚拟节点的存在以及重构的传输节点的密文子项, 存储开销将增加 $j|\mathbb{N}_T|L_{\mathbb{G}_T}+j|\mathbb{N}_T|L_{\mathbb{G}_0}$.

6.3 实验分析

本文实验基于 CP-ABE 工具包和 JPBC 库, 在 512 bit 的 A 类超奇曲线 $y^2=x^3+x$ 构造的 160 位的椭圆双曲线群上实现本文方案的实验. 实验的硬件配置为: Windows 10 64 位操作系统、CPU 为 Intel(R)Core(TM) i5-3470 (3.20 GHz), 64 bit, 内存为 8 GB. 所有实验结果均为 10 次实验的平均值.

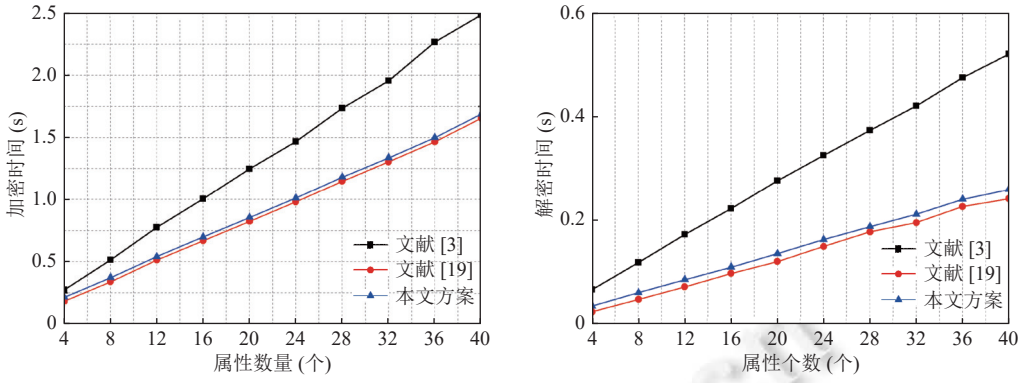
为了更为清楚地描述, 仍以图 1 给出的实际应用为例. 假设某公司将公司信息的访问策略按照规定的安全等级划分为 k 个访问层次 (对应 k 个数据文件 $M=\{m_1,\ldots,m_k\}$). 本文方案的访问策略应为: $\{[[[(a_1,\ldots,a_i,i \text{ of } i) \text{ AND } (\text{NULL})], b_1,\ldots,b_i,i \text{ of } (i+1)] \text{ AND } (\text{NULL}),\ldots, \text{ AND } (\text{NULL})], n_1,\ldots,n_i,i \text{ of } (i+1)\}$ (即策略最坏情况), 其中 a_i,b_i,n_i 表示属性. 同时, 本文方案规定属性的个数同步增长, 即 a_i 由 10 个增长到 20 个, 那么 b_i 也由 10 个增长到 20 个. 基于上述构造的访问策略, 根据文献 [3], 该公司需构造对应的 k 个访问策略. 例如, 该公司需要共享 3 份文件, $M=\{m_1,m_2,m_3\}$, 这 3 份文件依据安全等级划分为 3 个层次. 本文方案的访问策略构造为 $\{[[[(c_1,\cdots,c_i,i \text{ of } i) \text{ AND } (\text{NULL})], b_1,\ldots,b_i, i \text{ of } (i+1)] \text{ AND } (\text{NULL})], a_1,\ldots,a_i,i \text{ of } (i+1)\}$, 文献 [3] 的访问策略构造从高到低为: $\{(c_1,\ldots,c_i,i \text{ of } i) \text{ AND } (b_1,\ldots, b_i,i \text{ of } i) \text{ AND } (a_1,\ldots,a_i,i \text{ of } i)\}$, $\{(c_1,\ldots,c_i,i \text{ of } i) \text{ AND } (a_1,\ldots,a_i,i \text{ of } i)\}$, $\{c_1,\cdots,c_i,i \text{ of } i\}$. 文献 [19] 的访问策略构造为 $\{(c_1,\ldots,c_i,i \text{ of } i) \text{ AND } (b_1,\ldots,b_i,i \text{ of } i) \text{ AND } (a_1,\ldots,a_i,i \text{ of } i)\}$. 在本文实验中, 使用的属性数量设置为 $N=\{4,8,12,16, 20,24,28,32,36,40\}$, 文件数量设置为 $k=\{2,4,6,8\}$.

6.3.1 计算开销

图 5(a) 和图 5(b) 分别为共享 2 个层次文件时的加密和解密的计算开销. 当共享 2 份数据文件时, 本文方案在加解密效率方面远远高于文献 [3], 且加解密时间随着属性数量的增加呈线性增长, 当属性数量越多时, 本文方案的加解密计算开销将节省得越多. 如图 5(a) 所示, 当 $N=20$ 时, 文献 [3] 和本文方案的加密时间分别约为 1.248 s 和 0.856 s, 本文方案节省的时间约为 0.4 s, 加密的效率约提高了约 31.4%; 当 $N=40$ 时, 文献 [3] 和本文方案的加密时间分别约为 2.486 s 和 1.687 s, 本文方案节省的时间约为 0.8 s, 加密的效率约提高了约 32.1%. 如图 5(b) 所示, 当 $N=20$ 时, 文献 [3] 和本文方案的解密时间分别约为 0.277 s 和 0.136 s, 本文方案节省的时间约为 0.14 s, 解密的效率约提高了约 50.9%; 当 $N=40$ 时, 文献 [3] 和本文方案的解密时间分别约为 0.521 s 和 0.26 s, 本文方案节省的时间约为 0.26 s, 解密的效率约提高了约 50.1%. 显然, 当分层文件数量保持不变时, 两个方案的加解密时间开销随着属性数量的增加呈线性增长, 且本文方案明显优于文献 [3].

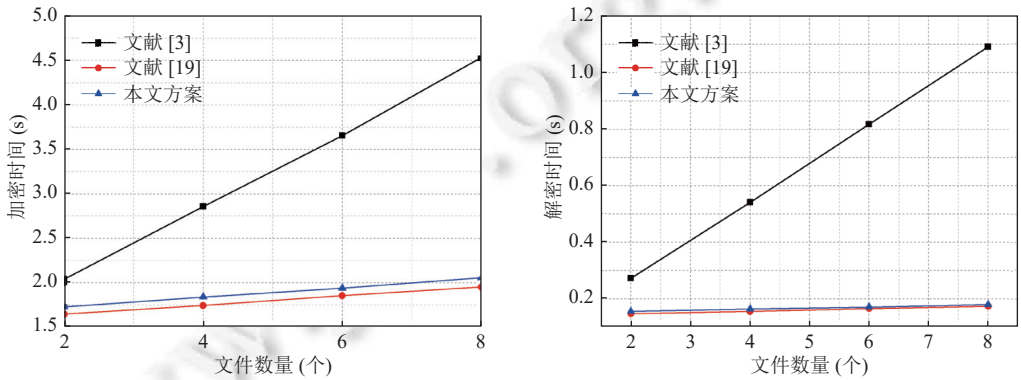
图 6(a) 和图 6(b) 分别为共享 2 份数据文件和属性为 40 个时的加密和解密的计算开销. 当属性数量固定不变时, 本文方案在加解密效率方面远远高于文献 [3], 加解密时间随文件数量的增加呈线性增长, 且本文方案的增长率相对较小. 如图 6(a) 所示, 当 k 从 2 变到 8 时, 文献 [3] 的加密时间分别为 2.04 s 和 4.533 s, 本文方案的加密时间分别约为 1.73 s 和 2.05 s, 差值从 0.31 s 增长到 2.476 s, 加密的效率分别提高了 15.2% 和 54.6%; 如图 6(b) 所示, 文献 [3] 的解密时间分别为 0.273 s 和 1.091 s, 本文方案的解密时间分别为 0.156 s 和 0.18 s, 差值从 0.117 s 增长到 0.911 s, 解密的效率分别提高了 42.9% 和 83.5%. 由此可得, 本文方案在加密和解密效率方面的优势更加明显. 因此, 本文方案更适用于共享安全分级的分层文件的环境中.

同时由图 5 和图 6 均可以看出, 由于本文方案中引入了虚拟节点, 并重新构造了传输节点相关的密文子项, 因此本文方案的加解密效率与文献 [19] 相比较略低一些. 但是在保证本文方案的分级访问的特性及更高的安全性的前提下, 牺牲的这部分效率显然是值得的.



(a) 加密时间随属性数量变化　(b) 解密时间随属性数量变化

图 5　加解密时间随属性数量变化对比

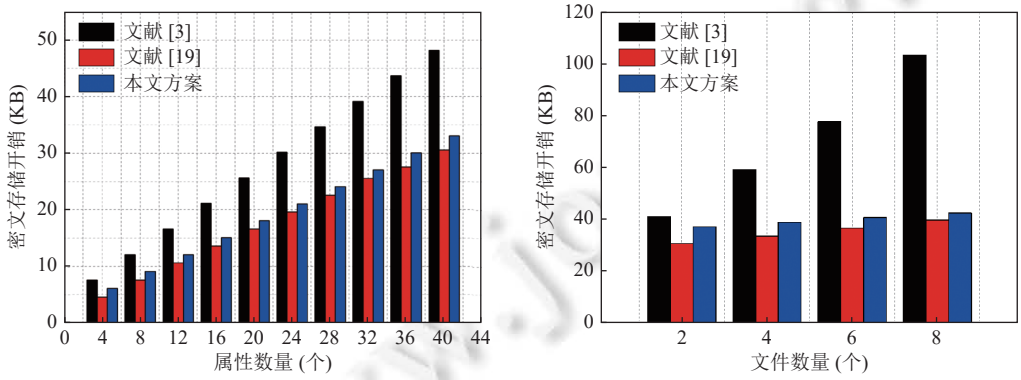


(a) 加密时间随文件数量变化　(b) 解密时间随文件数量变化

图 6　加解密时间随文件数量变化对比

6.3.2　存储开销

图 7(a) 和图 7(b) 分别共享 2 份数据文件和属性为 40 个时存储开销.



(a) 存储开销属性数量变化　(b) 存储开销随文件数量变化

图 7　存储开销对比

由图 7(a) 可见: 当共享的层次文件数量相同时, 随着属性数量增加, 两个方案的密文存储开销均有增长. 但是在相同条件下, 本文方案的存储开销明显小于文献 [3], 且本文方案的存储开销的增长率相对较小. 例如, 当 $N = 20$ 时, 文献 [3] 与本文方案的存储开销分别为 25.7 KB和 18.1 KB; 当 $N = 40$ 时, 两个方案的存储开销分别为 48.2 KB

和 33.1 KB. 当N从 20 变到 40 时, 两个方案的密文存储开销的差值从 22.5 KB 降低到 15 KB. 当属性数量固定不变时, 随着共享的层次文件的增加, 本文方案的存储开销也随之增加, 但是增长率远远小于文献 [3] 的增长率. 例如, 当k从 4 变到 8 时, CP-ABE 方案存储开销分别为 59.1 KB 和 103.1 KB, 本文方案的存储开销分别为 38.6 KB 和 48.8 KB. 两个方案的密文存储开销的差值从 44 KB 降低到 10.2 KB. 由此可看出本文方案节省了大量的存储空间.

与计算开销方类似地, 由于本文方案改变了访问结构树的构造和相关密文的构造, 因此存储开销也略高于文献 [19] 的存储开销, 当然牺牲的这部分开销也是值得的.

7 结 论

针对现有的文件分层 CP-ABE 方案普遍不支持分级用户访问, 且构造方面存在用户越权访问的问题, 本文提出了支持分级用户访问的文件分层 CP-ABE 方案. 该方案通过引入控制节点和虚拟属性, 构造了分级用户访问树, 并在加密阶段重新构造与传输节点相关的密文子项以避免用户越权访问, 解决了分级用户的层次文件共享问题. 本文方案加密和解密的时间开销和密文存储开销相比于传统的 CP-ABE 方案均有大幅度降低. 此外, 本文方案证明了在 DBDH 假设下选择明文攻击 CPA 是安全的.

References:

[1] Sahai A, Waters B. Fuzzy identity-based encryption. In: Proc. of the 24th Annual Int'l Conf. on the Theory Raphic Techniques. Aarhus: Springer, 2004. 457–473. [doi: 10.1007%2F11426639_27]

[2] Goyal V, Pandey O, Sahai A, Waters B. Attribute-based encryption for fine-grained access control of encrypted data. In: Proc. of the 13th ACM Conf. on Computer and Communications Security. Alexandria: ACM, 2006. 89–98. [doi: 10.1145/1180405.1180418]

[3] Bethencourt J, Sahai A, Waters B. Ciphertext-policy attribute-based encryption. In: Proc. of IEEE Symp. on Security and Privacy. Berkeley: IEEE Computer Society, 2007. 321–334. [doi: 10.1109/SP.2007.11]

[4] Horwitz J, Lynn B. Toward hierarchical identity-based encryption. In: Proc. of the Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Amsterdam: Springer, 2002. 466–481. [doi: 10.1007/3-540-46035-7_31]

[5] Gentry C, Silverberg A. Hierarchical ID-based cryptography. In: Proc. of the 8th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Queenstown: Springer, 2002. 548–566. [doi: 10.1007/3-540-36178-2_34]

[6] Boneh D, Boyen X. Efficient selective-ID secure identity-based encryption without random oracles. In: Proc. of Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Interlaken: Springer, 2004. 223–238. [doi: 10.1007/978-3-540-24676-3_14]

[7] Boneh D, Boyen X, Goh EJ. Hierarchical identity based encryption with constant size ciphertext. In: Proc. of the 24th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Aarhus: Springer, 2005. 440–456. [doi: 10.1007/11426639_26]

[8] Tsai TT, Tseng YM, Wu TY. RHIBE: Constructing revocable hierarchical ID-based encryption from HIBE. Informatica, 2014, 25(2): 299–326. [doi: 10.15388/Informatica.2014.16]

[9] Li J, Wang Q, Wang C, Ren K. Enhancing attribute-based encryption with attribute hierarchy. In: Proc. of the 4th Int'l Conf. on Communications and Networking. Xi'an: IEEE, 2009. 1–5.

[10] Wang GJ, Liu Q, Wu J. Hierarchical attribute-based encryption for fine-grained access control in cloud storage services. In: Proc. of the 17th ACM Conf. on Computer and Communications Security. Chicago Illinois: ACM, 2010. 735–737. [doi: 10.1145/1866307.1866414]

[11] Wan ZG, Liu JE, Deng RH. HASBE: A hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Trans. on Information Forensics and Security, 2012, 7(2): 743–754. [doi: 10.1109/TIFS.2011.2172209]

[12] Deng H, Wu QH, Qin B, Domingo-Ferrer J, Zhang L, Liu JW, Shi WC. Ciphertext-policy hierarchical attribute-based encryption with short ciphertexts. Information Sciences, 2014, 275: 370–384. [doi: 10.1016/j.ins.2014.01.035]

[13] Wang ZY, Wang J. A provably secure ciphertext-policy hierarchical attribute-based encryption. In: Proc. of the 2015 Int'l Conf. on Cloud Computing and Security. Nanjing: Springer, 2015. 38–48. [doi: 10.1007/978-3-319-27051-7_4]

[14] Chandar PP, Mutkuraman D, Rathinrai M. Hierarchical attribute based proxy re-encryption access control in cloud computing. In: Proc. of the 2014 Int'l Conf. on Circuits, Power and Computing Technologies. Nagercoil: IEEE, 2014. 1565–1570. [doi: 10.1109/ICCPCT.2014.7055015]

[15] Huang QL, Yang YX, Shen MS. Secure and efficient data collaboration with hierarchical attribute-based encryption in cloud computing. Future Generation Computer Systems, 2017, 72: 239–249. [doi: 10.1016/j.future.2016.09.021]

[16] Lin X, Han YL. Location hierarchical access control scheme based on attribute encryption. In: Proc. of the 36th Chinese Control Conf. (CCC). Dalian: IEEE, 2017. 9010–9014. [doi: 10.23919/ChiCC.2017.8028791]

[17] Ali M, Mohajeri J, Sadeghi MR, Liu XM. A fully distributed hierarchical attribute-based encryption scheme. Theoretical Computer Science, 2020, 815: 25–46. [doi: 10.1016/j.tcs.2020.02.030]

[18] Wang SL, Yu JP, Zhang P, Wang P. A novel file hierarchy access control scheme using attribute-based encryption. Applied Mechanics and Materials, 2015, 701–702: 911–918. [doi: 10.4028/www.scientific.net/AMM.701-702.911]

[19] Wang SL, Zhou JW, Liu JK, Yu JP, Chen JY, Xie WX. An efficient file hierarchy attribute-based encryption scheme in cloud computing. IEEE Trans. on Information Forensics and Security, 2016, 11(6): 1265–1277. [doi: 10.1109/TIFS.2016.2523941]

[20] Jiang SC, Guo WB, Fan GS. Hierarchy attribute-based encryption scheme to support direct revocation in cloud storage. In: Proc. of the 16th IEEE/ACIS Int'l Conf. on Computer and Information Science. Wuhan: IEEE, 2017. 869–874. [doi: 10.1109/ICIS.2017.7960114]

[21] Sandhia GK, Raja SVK, Jansi KR. Multi-authority-based file hierarchy hidden CP-ABE scheme for cloud security. Service Oriented Computing and Applications, 2018, 12(3–4): 295–303. [doi: 10.1007/s11761-018-0240-6]

[22] Guo R, Li X, Zheng D, Zhang YH. An attribute-based encryption scheme with multiple authorities on hierarchical personal health record in cloud. The Journal of Supercomputing, 2020, 76(7): 4884–4903. [doi: 10.1007/s11227-018-2644-7]

[23] Kang L, Zhang LY. Improving file hierarchy attribute-based encryption scheme with multi-authority in cloud. In: Proc. of the 2nd Int'l Conf. on Frontiers in Cyber Security. Xi'an: Springer, 2019. 3–18. [doi: 10.1007%2F978-981-15-0818-9_1]

[24] Chandrasekaran B, Nogami Y, Balakrishnan R. An efficient file hierarchy attribute based encryption using optimized tate pairing construction in cloud environment. Journal of Applied Security Research, 2020, 15(2): 270–278. [doi: 10.1080/19361610.2019.1649534]

[25] He H, Zheng LH, Li P, Deng L, Huang L, Chen X. An efficient attribute-based hierarchical data access control scheme in cloud computing. Human-centric Computing and Information Sciences, 2020, 10(1): 49. [doi: 10.1186/s13673-020-00255-5]

[26] Challagidad PS, Birje MN. Efficient multi-authority access control using attribute-based encryption in cloud storage. Procedia Computer Science, 2020, 167: 840–849. [doi: 10.1016/j.procs.2020.03.423]

[27] Li GJ, Chen NY, Zhang YC. Extended file hierarchy access control scheme with attribute-based encryption in cloud computing. IEEE Trans. on Emerging Topics in Computing, 2021, 9(2): 983–993. [doi: 10.1109/TETC.2019.2904637]

刘帅南(1997－), 男, 硕士, 主要研究领域为云计算, 信息安全.

冯朝胜(1971－), 男, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为网络与信息安全, 云计算, 大数据安全.

刘彬(1996－), 男, 硕士, 主要研究领域为信息安全, 区块链, 联邦学习.

秦志光(1956－), 男, 博士, 教授, 博士生导师, CCF 杰出会员, 主要研究领域为信息安全, 分布式计算.

郭真(1997－), 女, 硕士, 主要研究领域为云计算, 信息安全.

卿昱(1970－), 女, 研究员, 主要研究领域为网络与信息安全.