一种支持解密外包的 KP-ABE 方案

晋云霞[1],杨贺昆[1],冯朝胜[1,2],刘帅南[1],李 航[1],邹莉萍[1],万国根[3]
(1. 四川师范大学计算机科学学院,四川成都 610101;2. 电子科技大学网络与数据安全四川省重点实验室,四川成都 610054;
3. 成都信息工程大学网络空间安全学院,四川成都 610225)

摘 要: 针对现有密钥策略基于属性加密 KP-ABE(Key-Policy Attribute-Based Encryption)方案在解密时存在用户端计算开销大、解密时间长等问题,一些方案提出将解密外包给云服务器,但这些方案并未给出外包解密的并行化方法,存在解密效率低的问题. 本文提出一种支持解密外包的 KP-ABE 方案. 在该方案中,把大部分解密计算外包给 Spark 平台;并根据 KP-ABE 的解密特点设计并行化解密算法,完成对叶子节点和根节点的并行化解密. 性能分析表明,用户端仅需进行一次指数运算即可解密出共享数据,同时并行化设计能有效提高云端解密速率.

关键词: 密钥策略属性基加密; 解密外包; 快速解密; 共享访问树; 秘密共享; Spark
中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112 (2020)03-0561-07
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2020.03.020

A KP-ABE Scheme with Outsourced Decryption

JIN Yun-xia[1], YANG He-kun[1], FENG Chao-sheng[1,2], LIU Shuai-nan[1], LI Hang[1], ZOU Li-ping[1], WAN Guo-gen[3]
(1. *College of Computer Science, Sichuan Normal University, Chengdu, Sichuan* 610101, *China*;
2. *Network and Data Security Key Laboratory of Sichuan Province, University of Electronic Science and Technology of China, Chengdu, Sichuan* 610054, *China*;
3. *School of Cyber Security, Chengdu University of Information Technology, Chengdu, Sichuan* 610225, *China*)

Abstract: For most of the existing key-policy attribute-based encryption schemes, there are some problems in the decryption, such as the high cost of the client computing, long decryption time and high resource consumption. Some schemes propose outsourcing decryption to cloud servers. However, these schemes do not give the specific parallelization method of outsourcing decryption in cloud server, and there are problems of low efficiency of cloud decryption. To solve these problems, this paper presents a KP-ABE scheme for decryption outsourcing. In this scheme, most of decryption computation is outsourced to Spark platform; and according to the decryption characteristics of KP-ABE, a decryption parallelization algorithm is designed to complete the parallel decryption of leaf nodes and root nodes. The performance analysis shows that most of decryption computing is done by cloud servers and the client can decrypt the shared data by shared access tree with only once exponential operation, and the parallel design can effectively improve the cloud decryption rate.

Key words: key-policy attribute-based encryption; outsourced decryption; fast decryption; shared access tree; secret sharing; Spark

1 引言

云计算能有效解决用户端计算、存储能力不足等问题[1],允许用户将数据的处理和存储外包出去. 但数据外包使得数据脱离数据所有者掌控[2]难以确保数据机密性. 针对传统加密方法能保护数据隐私却无法在云环境下实现数据的高效共享和细粒度访问,基于属性的加密方法(Attribute-Based Encryption, ABE)[3]应运而生. 该方法通过一对多的加密模式实现了加密数据的高效共享;使用属性作为访问控制的最小单位实现了加密数据的细粒度访问. ABE 主要分为两种:密文策略基于属性加密 CP-ABE[4](Ciphertext-Policy Attribute-Based Encryption)和密钥策略基于属性加密 KP-ABE[5]. KP-ABE 采用密钥策略与访问属性相关联,CP-

收稿日期:2019-03-26;修回日期:2019-06-28;责任编辑:覃怀银
基金项目:国家自然科学基金(No. 61373163);国家科技支撑计划课题(No. 2014BAH11F02, No. 2014BAH11F01);网络与数据安全四川省重点实验室课题(No. NDS2019-1);四川省科技成果转化平台项目(No. 2018CC0060);四川省科技计划项目(No. 2017GZ0006)

ABE 采用密文策略与访问属性相关联.

在 KP-ABE 方案中,对于资源有限的用户端来说,无法快速解密. 解密外包的引入,能减轻用户端的计算开销,实现快速解密;但现有 KP-ABE 解密外包方案均未采用并行计算技术,云解密速度尚有较大提升空间.

针对上述问题,将并行计算技术引入 KP-ABE 方案设计中,以 Goyal 等[5]方案(以下简称 GPSW 方案)和方案[6]为基础,设计了一种支持并行化解密外包的 KP-ABE 方案.

2005 年,Sahai 和 Waters 首先提出了基于属性加密算法 ABE[3]. 该算法用属性标识用户身份,当且仅当解密者的属性值与加密者的属性值匹配达到一定门限值时才能解密. 基于属性加密有两种扩展,分别是 2006 年 Goyal 等[5]提出的密钥策略基于属性加密算法 KP-ABE 和 2007 年 Bethencourt 等[4]提出的密文策略基于属性加密算法 CP-ABE. 2010 年, Yu 等[7]提出了一种基于云计算环境的 KP-ABE 方案. 该方案在保障数据安全的情况下充分利用了云计算的存储和计算能力,并利用代理重加密和懒惰法来提升加密效率. 2011 年,Green 等[6]基于文献[3]提出了支持解密外包的 ABE 方案. 该方案同时适用于 KP-ABE 和 CP-ABE. 此方案中,所有复杂的双线性配对运算和大多数指数运算都在外包云平台上完成,降低了用户端的解密消耗. 2012 年,Attrapaduang 等[8]提出了密文长度固定的 KP-ABE 方案,该方案对单调访问和非单调访问结构都适用,但用户密钥子项的数量是属性空间大小的平方. 2013 年,Hohenberger 等[9]分析了现有 KP-ABE 方案,提出一种仅需进行两次双线性配对的解密方案,该方案的用户私钥大小和生成计算量增加 $|T|$ 倍(T 表示构成访问结构的属性集合). 2014 年 Lai 等[10]提出了一种完全安全的 KP-ABE 快速解密方案. 该方案在解密时仅计算两次双线性对运算,但在密钥生成时的计算量、大小和解密时的计算量都会增加 n 倍(n 为属性个数). 2015 年,Lin 等[11]提出一种可验证的解密外包方案,该方案利用哈希映射的方式验证外包解密的正确性. 2016 年, Rahulamathavan 等[12]利用匿名发布协议提出了一种强隐私保护的去中心化的支持解密外包的 KP-ABE 方案. 2017 年, Liu 等[13]提出一种支持解密外包和属性撤销的 ABE 方案,但未给出具体云端解密操作. 迄今为止,已有不少通过将解密外包给云服务提供商来提升用户端解密效率的 ABE 方案被提出,但云服务器在解密时仍然采用效率较低的串行计算方式.

2 系统模型及定义

2.1 系统构成

支持解密外包的 KP-ABE 方案系统模型,包括授权机构、云服务器、数据拥有者、数据消费者,如图 1 所示.

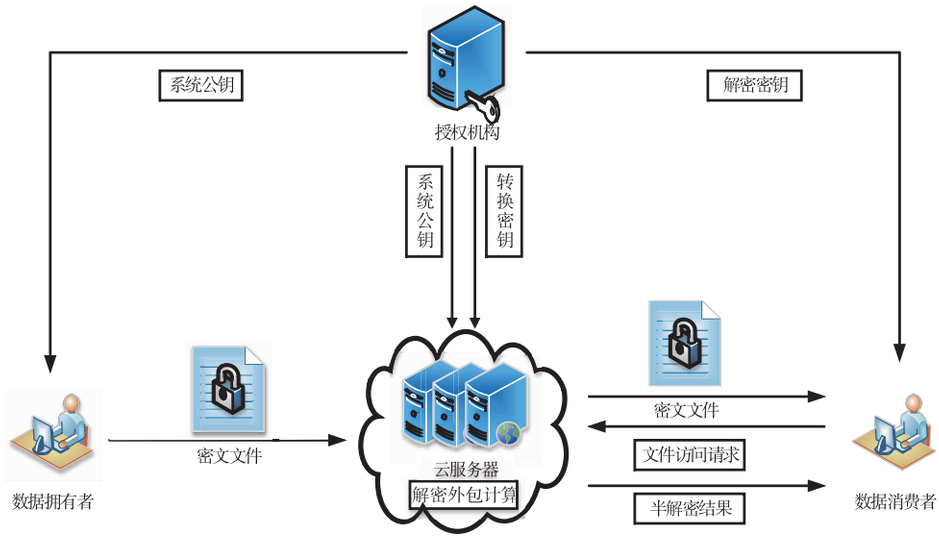

图1 支持解密外包的KP-ABE系统模型

授权机构 授权机构完全可信,它生成系统公钥 PK、系统主密钥 MK、转换密钥 TK、用户私钥 SK、用户解密密钥 DK.

云服务器 云服务器半可信,用于存储密文数据并对密文数据 CT 进行半解密.

数据拥有者 数据拥有者是系统模型中原始数据的所有者,加密原始数据,并将加密后的密文数据 CT 上传至云服务器.

数据消费者 数据消费者是系统模型中共享密文的使用者. 数据消费者从云服务器处下载半解密密文 CT′,并使用从授权机构获取的解密密钥 DK 对半解密密文 CT′解密,得到明文数据.

2.2 算法定义

定义 一种支持解密外包的 KP-ABE 方案由以下五个算法构成.

(1) Setup(γ,U) $\rightarrow$(PK,MK):该算法由授权机构执行,输入安全参数 γ 和属性空间 U,输出系统公钥 PK 和系统主密钥 MK.

(2) Encryption(S,PK,M) $\rightarrow$(CT):该算法由数据拥有者执行,输入属性集合 S、系统公钥 PK、明文 M,输出密文 CT.

(3) KenGen(T,MK)$\rightarrow$(TK,DK,SK):该算法由授权机构执行,输入访问策略 T 和系统主密钥 MK,输出转换密钥 TK、用户解密密钥 DK、用户私钥 SK.

(4) Transformation(TK,CT)$\rightarrow$(CT′):该算法由云服务器执行,输入转换密钥 TK 和密文 CT,输出半解密密文 CT′.

(5) Decryption(CT′,DK)$\rightarrow$(M):该算法由数据消费者执行,输入半解密密文 CT′、用户解密密钥 DK,输出明文 M.

2.3 安全模型定义

sCPA(Selectively CPA)安全模型定义如下.

●Init:敌手 A 声明想要挑战的属性集合 S.

●Setup:挑战者 B 运行 Setup 算法,并把系统公钥 PK 发送给敌手 A.

●Phase 1:允许敌手 A 查询属性集 S 不满足访问结构 T 的密钥.

●Challenge:敌手 A 提交两个长度相等的消息 M_0,M_1. 挑战者 B 从 M_0和 M_1中随机选择一条明文消息 M_b,并基于属性集 S 和系统公钥 PK 加密 M_b得到密文 CT*;然后把密文 CT*传递给敌手 A.

●Phase 2:重复 Phase 1 操作.

●Guess:敌手 A 输出 b 的猜想 b'.

在该游戏中敌手 A 的优势如式(1)所示

$$\varepsilon = \Pr[b' = b] - \frac{1}{2} \tag{1}$$

在上述游戏中,若敌手 A 在任何多项式时间内的攻击优势可忽略,则安全.

3 支持并行解密外包的 KP-ABE 方案

3.1 共享访问树

在共享访问树的构建上,密文由属性集标记. 私钥由树访问结构标识,其中树的每个非叶子节点表示一个门限值,每个叶子节点与属性相关联. num_x 表示叶子节点个数,k_x 表示门限,当 $k_x = 1$ 时表示“OR”门,$k_x = \text{num}_x$ 时表示“AND”门.

为方便使用共享访问树,定义 parent(*)表示树中节点 * 的父节点,att(*)表示与属性相关联的叶子节点 *,index(*)表示节点 * 的兄弟节点序号,序号按照共享访问树结构从左往右依次编号.

令共享访问树 T 的根节点为 R,T_x 表示以 x 为根的子树. 每个节点 x 的孩子数用 k_x 表示,随机选择一个一元 $k_x - 1$ 次多项式 q_x 给 x,秘密共享数从访问树 T 的根节点 R 开始自上而下分发. $\forall s \in Z_p$,根节点 R 秘密共享数 $s = q_R(0)$;其它节点的秘密共享数表示为 $q_{\text{parent}(x)}(\text{index}(x))$.

若属性集 S 满足共享访问树 T_x,则 $T_x(S) = 1$ 接下来计算 $T_x(s)$. 如果 x 是非叶子节点,计算 x 的孩子节点 x'的 $T'_x(s)$;至少有 k_x 个孩子返回 1 时,$T_x(s) = 1$. 若 x 是叶子节点,$\text{att}(x) \in S$,则 $T_x(s) = 1$.

3.2 加密共享数

图 2 是一个共享访问树,假设 S_X表示节点 X 对应的秘密共享数,C_X表示节点 X 上的加密共享数,$C_X = e(g,g)^{\alpha \cdot S_X}(\alpha \in Z_p)$,$L_{xy}$表示节点 x 的第 y 个孩子的拉格朗日系数.

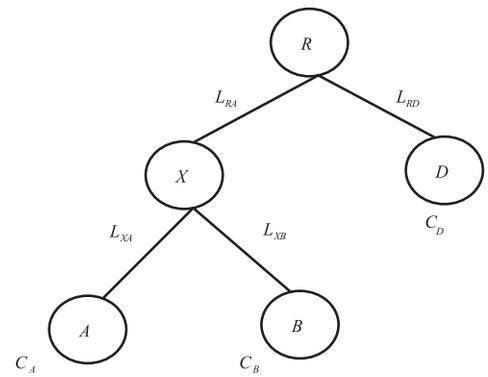

图2　共享访问树(所有非叶节点均表示AND门)

根节点的加密共享数 C_R计算过程见式(2)~(4).

$$C_X = C_A^{L_{XA}} \cdot C_B^{L_{XB}} = e(g,g)^{\alpha S_A L_{XA}} \cdot e(g,g)^{\alpha S_B L_{XB}} \tag{2}$$

$$C_D = C_D^{L_{RD}} = e(g,g)^{\alpha S_D} \tag{3}$$

$$C_R = C_X^{L_{RA}} \cdot C_D^{L_{RD}} = e(g,g)^{\alpha S_A L_{XA} L_{RA}} \cdot e(g,g)^{\alpha S_B L_{XB} L_{RA}} \cdot e(g,g)^{\alpha S_D L_{RD}} \tag{4}$$

基于以上观察,可得定理 1.

定理 1 共享访问树的根节点的加密共享数等于以每个叶子节点的加密共享数为底、以从叶子节点到根节点路径上所有节点的拉格朗日积为指数幂的乘积.

3.3 方案设计

(1)授权机构执行 Setup:输入属性空间 $U = \{1,2,\cdots,n\}$,安全参数 γ;选择以大素数 p 为阶,$g \in G_1$ 为生成元的双线性群 G_1、G_2,定义双线性映射 $e: G_1 \times G_1 \rightarrow G_2$. 定义拉格朗日系数:$\Delta_{i,S(x)} = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$(其中 $i \in Z_p$,

$S \in Z_p$). $\forall i \in U$,选择随机数 $\alpha \in Z_p$、$t_i \in Z_p$,输出系统主密钥 MK = $(t_1, \cdots, t_{|U|}, \alpha)$ 和系统公钥 PK,如式(5)所示.

$$\mathrm{PK} = (G_1, g, T_1 = g^{t_1}, \cdots, T_{|U|}) = g^{t_{|U|}}, Y = e(g,g)^{\alpha}) \tag{5}$$

(2)数据拥有者执行 Encryption:输入公钥 PK、明文 $M \in G_2$、属性集 S,选择一个随机数 $s \in Z_p$,得到密文 CT 见式(6)并上传至云服务器:

$$\mathrm{CT} = (S, C' = M \cdot Y^s, \{C_i = T_i^s\}_{i \in S}) \tag{6}$$

(3)授权机构执行 KenGen:输入访问共享树 T、主密钥 MK,输出转换密钥 TK 和用户解密密钥 DK.

按照共享访问树进行秘密值的分发. 从根节点 R 开始给共享访问树中的每一个非叶子节点 x 选择一个一元 $k_x - 1$ 次多项式 q_x. 秘密分享从根节点 R 开始自上而下进行分发,设置根节点 $q_R(0) = \alpha$,设置其他非叶子节点 x 为 $q_x(0) = q_{\mathrm{parent}(x)}(\mathrm{index}(x))$. 叶子节点: $D_x = (g^{\frac{q_x(0)}{t_i}}, i = \mathrm{att}(x))$,集合 SK = $\{D_x = g^{\frac{q_x(0)}{t_i}}, i = \mathrm{att}(x)\}$ 为用户私钥.

选择一个随机数 $n \in Z_p$,对 SK 进行盲化,并得到转换密钥,如式(7)所示

$$\mathrm{TK} = \{D_x^n = (g^{\frac{q_x(0)}{t_i}})^n, i = \mathrm{att}(x)\} \tag{7}$$

利用安全信道把用户解密密钥 DK = $\{n\}$ 发给数据消费者.

(4)云服务器执行 Transformation:输入密文 CT 和转换密钥 TK,根据定理 1,计算半解密密文 CT′. 在本方案中,整个数据转换算法采用 Spark 技术中的 Map 和 Reduce 操作完成共享访问树 T 根节点秘密值的恢复. 为共享访问树的每个节点启动一个 Map;为共享访问树中每个叶子节点启动一个 Reduce.

Map 操作 该模块按照叶子节点和非叶子节点分为如下两种情况:

(1)对于叶子节点 A,令 $i = \mathrm{att}(A)$,$\mathrm{NodeLeaf}_A$ 为节点标识符,Map 输入的键值对为($\mathrm{NodeLeaf}_A$, $g^{s \cdot t_i}$). 该节点的部分秘密值计算过程见式(8).

$$\begin{aligned} C'_A &= \mathrm{DecNode}(\mathrm{CT}, \mathrm{TK}, A) \\ &= e(\mathrm{TK}_A, C_i) \\ &= e((g^{\frac{q_x(0)}{t_i}})^n, g^{s \cdot t_i}) \\ &= e(g,g)^{n \cdot s \cdot q_x(0)} (i \in S) \end{aligned} \tag{8}$$

否则输出 $\perp$.

将计算结果输出到叶子节点 A 对应的 Reduce,即 Map 操作输出的键值对为($\mathrm{NodeLeaf}_A$, C'_A).

(2)对于非叶子节点 X,用 x 表示 X 的所有孩子节点,则 Map 的输入键值对为(Node_X, $\mathrm{NodeLeaf}_x$),并计算非叶子节点 X 的每个孩子节点对应拉格朗日系数值 $\Delta_{i,S'_X(0)}$,其中 $i = \mathrm{index}(x)$,$S'_X(0) = \{\mathrm{index}(x) : x \in S_X\}$ 将计算结果输出到叶子节点对应的 Reduce,即 Map 操作输出的键值对为(Node_X, $\Delta_{i,S'_X(0)}$).

Reduce 操作 输入叶子节点 A 的 Map 输出键值对($\mathrm{NodeLeaf}_A$, C'_A)和该叶子节点到根节点路径上所有非叶子节点的 Map 输出键值对(Node_X, $\Delta_{i,S'_X(0)}$),计算叶子节点到根节点的拉格朗日系数乘积: $\prod \Delta_{i,S'_X(0)}$. 则叶子节点 A 在 Reduce 操作输出的键值对为($\mathrm{NodeLeaf}_A$, $C'^{\Delta_{i,S'_R(0)}}_A$).

得到根节点秘密值 C'_R,从而得到半解密密文 CT′ 计算过程如式(9):

$$\begin{aligned} C'_R &= \mathrm{DecNode}(\mathrm{CT}, \mathrm{TK}, R) \\ &= e(g,g)^{n \cdot s \cdot q_r(0)} \end{aligned} \tag{9}$$

最后得到半解密密文 CT′ = $\{(g,g)^{n \cdot s \cdot \alpha}\}$. 数据转换算法的并行化流程如图 3 所示.

(5)数据消费者执行 Decryption:输入半解密密文 CT′和用户解密密钥 DK,通过计算可求得明文 M.

使用用户解密密钥 DK,对从云服务器回传的部分解密密文 CT′进行计算:$(\mathrm{CT}')^{1/\mathrm{DK}}$ 得 Y^s,且 $C' = MY^s$,故 $M = \dfrac{C'}{Y^s}$.

4 安全性证明与性能分析

4.1 安全性证明

定理 2 本文方案和 GPSW 方案一样,达到 sCPA 安全.

假设敌手 A 在标准模型下有不可忽略的优势攻破本文方案,那么可以构建挑战者 B,使其在相同安全模型下攻破 GPSW 方案的优势也不可忽略,与 GPSW 方案达到 sCPA 安全相矛盾.

●Init:敌手 A 申明要挑战的属性集合 S.

●Setup:挑战者 B 获取 GPSW 方案的公钥 PK = $(G_1, g, T_i = g^{t_i}, Y = e(g,g)^{\alpha})$并发送给 A.

●Phase 1:假设敌手 A 向 B 提交属性集合 S 不满足的访问结构 T 对应的用户私钥请求. B 将请求发送给 GPSW 方案的挑战者,由 GPSW 方案的挑战者利用密钥生成算法生成用户私钥 SK′并返回给 B. 然后 B 选择一个随机数 $n \in Z_p$,生成转换密钥 TK = $(\mathrm{SK}')^n$. 将(n, TK)返回给 A.

Challenge:A 向挑战者 B 提交长度相等的明文消息 M_0、M_1,然后 B 向 GPSW 方案挑战者发送消息 M_0、M_1和属性集合 S,从 GPSW 方案挑战者获得密文 CT = $(S, C' = M_b \cdot Y^s, \{C_i = T_i^s\}_{i \in S})$,$B$ 把密文 CT 发送给 A.

●Phase 2:S 不满足访问结构,重复 Phase 1. 当 S 满足访问结构,无法查询对应的私钥,B 随机选择 $d \in Z_p$,GPSW 挑战者运行密钥生成算法生成伪私钥 SK′,进而生成伪转换密钥 TK′ = $(\mathrm{SK}')^d$.

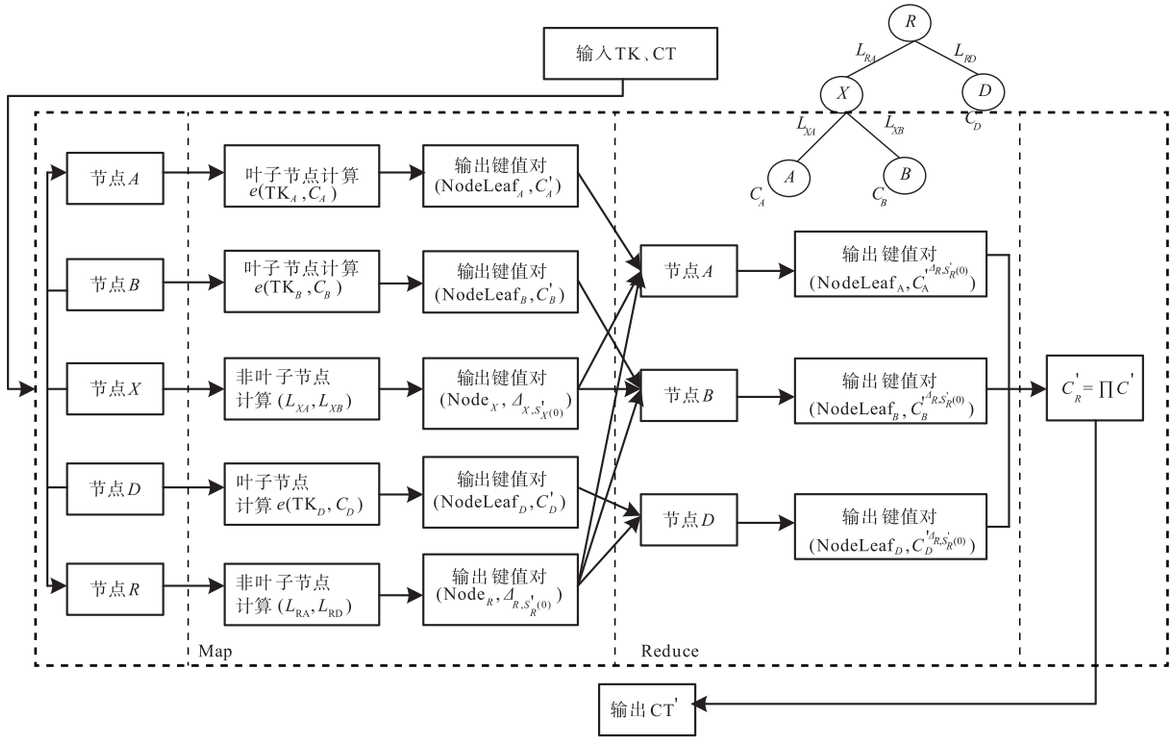

图3　Spark集群半解密模型

●Guess:若敌手 A 输出对 b 的猜想 b', 则 B 输出对 b 的猜想也为 b'. 因此,如果敌手 A 能以不可忽略的优势攻破本文所提出的方案,那么 B 也能以不可忽略的优势攻破 GPSW 方案.

4.2　性能分析

解密开销主要来源于双线性对运算和指数运算,故本文用这两项指标衡量计算性能. 令 $|T_L|$ 表示 $|T|$ 的叶子节点个数,P 表示双线性配对运算时间,E_G 表示一次 G_1 域指数运算,E_{G_2} 表示一次 G_2 域指数运算. 表 1 为本方案与 GPSW 方案、方案[6]、方案[12]的解密开销对比:本方案客户端解密时间为 $1E_G$;云服务器解密所需时间为 $|T_L|P+(|T|-1)E_G$,故本方案的总解密时间为:$|T_L|P+|T|E_G$.

表 1　解密开销对比

方案	客户端解密时间	云服务器解密时间	总解密时间
GPSW 方案	$(\|T_L\|P+(\|T\|-1)E_G)$	–	$(\|T_L\|P+(\|T\|-1)E_G)$
文献[6]方案	$1E_G$	$(\|T_L\|+1)P+(2\|T\|-1)E_G$	$(\|T_L\|+1)P+(2\|T\|)E_G$
文献[12]方案	$\|T\|E_G$	$(\|T_L\|+\|T\|+1)(P+E_{G_2})$	$(\|T_L\|+\|T\|+1)(P+E_{G_2})+\|T\|E_G$
本文方案	$1E_G$	$\|T_L\|P+(\|T\|-1)E_G$	$\|T_L\|P+\|T\|E_G$

4.3　实验分析

基于本方案实现了一个 KP-ABE 系统,所实现的系统采用 160 位椭圆曲线群,椭圆曲线为 512 位有限域上的超奇异椭圆曲线 $y^2=x^3+x$. 实验使用的虚拟 PC 机用户客户端配置为:CentOS6. 5 64 位操作系统,Intel(R) CPU(E5-2620 2. 0GHz) 1GB 内存;虚拟云服务器配置为:CentOS6. 5 64 位操作系统,4 个 Intel(R) CPU(E5-2620 2. 0GHz) 8GB 内存. 实验使用虚拟云服务器的 Spark2. 1. 1 集群配置为:5 台虚拟机,Cent-OS6. 5 64 位操作系统,2 个 Intel(R) CPU(E5-2620 2. 0GHz)8GB 内存,其中 1 台为 master 节点,4 台 slave 节点.

实验评估了客户端解密时间、云服务器解密时间和总解密时间. 对于客户端解密时间和总解密时间,本方案分别与 GPSW 方案、文献[6]方案、文献[12]方案进行对比;对于云服务器解密时间,本方案分别与文献[6]方案、文献[12]方案进行对比.

实验使用虚拟 PC 机作为客户端进行对比,实验数据和访问共享策略属性数量均相同且逻辑关系均为“AND”.

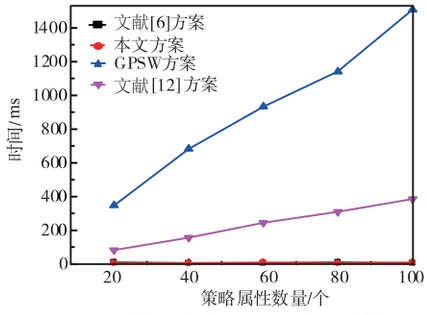

图4 客户端解密时间对比图

如图4 所示本方案用户客户端解密时间保持在5ms 以内,满足用户端响应时间要求.

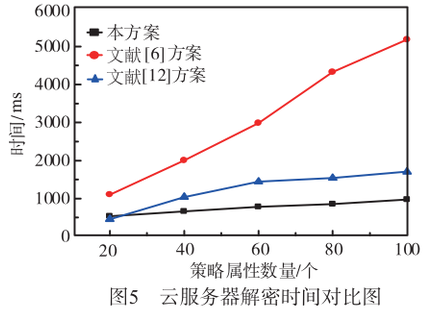

图5 云服务器解密时间对比图

如图5 所示云服务器解密时间随着策略属性数量的增加而增加,但本方案的云服务器解密时间增长速率低于文献[6]方案.

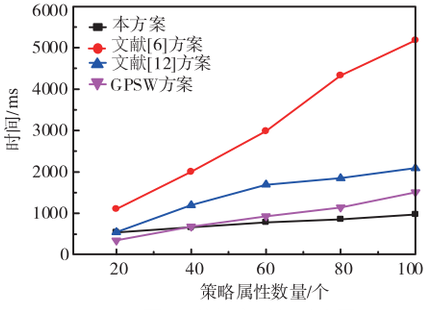

图6 总解密时间对比图

如图6 显所示本方案总解密时间增长速率最低;总解密时间在1s 以内,能实现快速解密.

5 结语

目前 KP-ABE 方案在解密时仍然存在计算时间较长、计算效率较低等问题,导致计算资源有限的用户端难以快速解密. ABE 解密外包方案的提出能有效降低用户端的计算消耗,但外包解密计算仍采用低效的串行方式. 针对该问题,把 Spark 集群和并行计算方法引入到 KP-ABE 方案设计中,提出一种支持解密外包的 KP-ABE 方案. 在该方案中,将解密过程中求解共享访问树根节点加密共享数的任务分解成可并行执行的 Map 任务和 Reduce 任务. 安全性分析表明,该方案达到 sCPA 安全.

参考文献

[1] 冯朝胜,秦志光,袁丁. 云数据安全存储技术[J]. 计算机学报,2015,38(1):150 – 163.
FENG Chao-sheng,QIN Zhi-guang,et al. Technigues of secure storage for cloud data[J]. Chinese Journal of Computers,2015,38(1):150 – 163. (in Chinese)

[2] 冯朝胜,秦志光,袁丁,等. 云计算环境下访问控制关键技术[J]. 电子学报,2015,43(2):312 – 319
FENG Chao-sheng, QIN Zhi-guang, et al. Key techniques of access control for cloud computing[J]. Acta Electronica Sinica,2015,43(2):312 – 319. (in Chinese)

[3] Sahai A,Waters B R. Fuzzy identity-based encryption[A]. Ronald Cramer. Proceedings of the 24th Annual International Conference on Theory and Applications of Cryptographic Techniques[C]. Berlin:Springer,2004. 457 – 473.

[4] Bethencourt J,Sahai A,Waters B. Ciphertext-policy attribute-based encryption[A]. IEEE Symposium on Security and Privacy[C]. Washington: IEEE Computer Society, 2007. 321 – 334.

[5] Goyal V,Pandey O,Sahai A,et al. Attribute-based encryption for fine-grained access control of encrypted data[A]. Computer and Communications Security[C]. New York: ACM,2006. 89 – 98.

[6] Green M,Hohenberger S,Waters B,et al. Outsourcing the decryption of ABE ciphertexts[A]. Usenix Security Symposium[C]. Berkeley:Usenix Association,2011. 34 – 34.

[7] Yu S,Wang C,Ren K,et al. Achieving secure,scalable,and fine-grained data access control in cloud computing [A]. International Conference on Computer Communications[C]. Berlin,Heidelberg:Springer,2010. 534 – 542.

[8] Attrapadung N,Herranz J,Laguillaumie F,et al. Attribute-based encryption schemes with constant-size ciphertexts [J]. Theoretical Computer Science,2012,422:15 – 38.

[9] Hohenberger S,Waters B. Attribute-based Encryption with Fast Decryption[M]. Berlin, Heidelberg: Springer, 2013. 162 – 173.

[10] Lai J,Deng R H,Li Y,et al. Fully secure key-policy attribute-based encryption with constant-size ciphertexts and fast decryption[A]. ACM Symposium on Information [C]. New York:ACM,2014. 239 – 248.

[11] Lin S,Zhang R,Ma H,et al. Revisiting attribute-based encryption with verifiable outsourced decryption[J]. IEEE Transactions on Information Forensics and Security,2015,

10(10):2119 - 2130.

[12] Rahulamathavan Y, Veluru S, Han J, et al. User collusion avoidance scheme for privacy-preserving decentralized key-policy attribute-based encryption[J]. IEEE Transactions on Computers, 2016, 65(9):2939 - 2946.

[13] Liu Z, Jiang Z L, Wang X, et al. Offline/online attribute-based encryption with verifiable outsourced decryption [J]. Concurrency and Computation: Practice and Experience, 2017, 29(7):1532 - 0626

作者简介

晋云霞　女,1993 年生于四川成都. 四川师范大学在读研究生. 研究方向信息安全.
E-mail:jinyunnvshi@ foxmail. com

杨贺昆　男,1992 年生于河南驻马店. 四川师范大学在读研究生. 研究方向大数据安全.
E-mail:yangpotatoes@ gmail. com

冯朝胜(通信作者)　男,1971 年 1 月生于四川广元. 教授、硕士生导师. 研究方向云计算安全.
E-mail:csfenggy@ 163. com

刘帅南　男,1997 年生于安徽宿州,四川师范大学在读研究生. 研究方向为云计算与信息安全.
E-mail:liushuainan9721@ 163. com

李　航　男,1997 年出生于四川巴中. 四川师范大学在读研究生. 研究方向为云计算与信息安全.
E-mail:hanghanglh@ foxmail. com

邹莉萍　女,1994 年生于四川乐山. 四川师范大学硕士研究生. 研究方向为信息安全.

万国根　男,1963 年生,教授、博士. 研究方向:网络空间安全.
E-mail:wanguogen@ cuit. edu. cn