小 型 微 型 计 算 机 系 统
Journal of Chinese Computer Systems

2016 年 10 月 第 10 期
Vol. 37 No. 10 2016

一种高效的 CP-ABE 云数据访问控制方案

严新成[1,3],陈 越[1,3],翟雨畅[2],兰巨龙[1,4],黄恺翔[1,3]

[1](解放军信息工程大学,郑州 450001)
[2](香港浸会大学,香港 九龙塘 999077)
[3](数学工程与先进计算国家重点实验室,郑州 450001)
[4](国家数字交换系统工程技术研究中心,郑州 450002)
E-mail:imtodshine@163.com

摘 要: 针对属性加解密过程中客户端计算开销大以及访问结构私密性未经保护等问题,提出一种基于密文策略的属性加密云数据高效访问控制方案.方案通过增加代理加解密服务器来减轻用户加解密计算开销并实现访问策略的隐藏;采用层次加密的思想来减轻数据加密的计算开销,即用对称密钥加密上传数据以保证效率,用属性密钥加密对称密钥以保证安全性;并通过设置授权用户集合来避免非授权用户的属性验证.实验分析表明,该方案与现有的访问控制方案相比,在加解密效率上有较大提升,有效缓解了客户机的加解密负担.

关 键 词: 云存储;访问控制;数据外包;属性加密;隐藏策略
中图分类号: TP393 **文献标识码:** A **文 章 编 号:** 1000-1220(2016)10-2155-07

An Efficient CP-ABE Based Access Control Scheme for Cloud Storage

YAN Xin-cheng[1,3],CHEN Yue[1,3],ZHAI Yu-chang[2],LAN Ju-long[1,4],HUANG Kai-xiang[1,3]
[1](PLA Information Engineering University,Zhengzhou 450001,China)
[2](Hong Kong Baptist University,Hong KongKowloon Tong 999077,China)
[3](State Key Laboratory of Mathematical Engineering and Advanced Computing,Zhengzhou 450001,China)
[4](National Digital Switching System Engineering & Technological R&D Center,Zhengzhou 450002,China)

Abstract: Aiming at the problems of the heavy client computing overhead for encryption and decryption and access structure privacy disclosure,an efficient CP-ABE (ciphertext policy-attribute based encryption) based access control scheme for cloud storage is proposed. By adding the proxy servers,the client computing overhead for encryption and decryption is greatly reduced and the access policies can be hided;Hierarchical encryption is introduced to reduce the computing overhead for data encryption,in which the symmetric key is used to encrypt upload data for efficient and the attribute key is used to encrypt symmetric key for security;By providing the authorized user set,the attribute set verification for unauthorized users can be avoided. The experiments show that,compared with the existing access control schemes,our scheme has higher encryption and decryption efficiency and clients' burden is effectively alleviated.

Key words: cloud storage;access control;data outsourcing;attribute encryption;policy hiding

1 引 言

云计算是当前信息技术领域的热点和关注焦点.按照美国国家标准与技术研究院(National Institute of Standards and Technology,NIST)的定义,云计算是一种模型,这种模型提供可用的、便捷的、按需的网络访问,进入可配置的计算资源池(如网络、服务器、存储设备、应用程序以及服务).这些资源可以以最小化管理成本或尽可能少的服务提供商交互的方式,被迅速提供并发布.

云计算发展面临许多关键性问题,其中数据存储的安全性访问是制约其发展的重要因素并引起广泛关注.云存储是在云计算基础上延伸和发展出来的概念.由于数据外包在云端,数据拥有者(Data Owner,DO)不能像管理本地文件那样来管理其存放在云端的数据,因此外包数据的安全性显得尤为重要.例如,云计算用户的数据存放在云端,云服务提供商(Cloud Service Provider,CSP)可能会因为商业利益而非法访问用户的数据,导致用户敏感信息泄露.尽管一些主流的 CSP 如 Amazon(S3)、Apple(iCloud)、Dropbox 等都在其隐私策略声明中表示在保有访问用户文件权力的情况下保护用户隐私[1-5],但由利益驱动、误操作、系统漏洞等原因造成的数据泄露事件却时有发生[6-8].因此,解决云存储数据机密性威胁较好的办法是对存储在云中的数据实施访问控制,比如通过密码机制使得只有具备相应密钥的授权人员才能解密密文,进而达到数据安全性访问的目的.

2 相关工作

文献[9]提出了基于角色的面向云环境的自适应访问控

收稿日期:2015-10-07 收修改稿日期:2016-01-19 基金项目:国家"九七三"重点基础研究发展计划项目(2012CB315901)资助. 作者简介:**严新成**,男,1991 年生,硕士研究生,研究方向为网络与信息安全;**陈 越**,男,1965 年生,博士,教授,博士生导师,研究方向为网络与信息安全;**翟雨畅**,女,1990 年生,硕士研究生,研究方向为数据挖掘、信息可视化;**兰巨龙**,男,1962 年生,博士,教授,博士生导师,研究方向为宽带信息网络;**黄恺翔**,男,1987 年生,博士研究生,研究方向为网络与信息安全.

制模型,可以根据资源的动态变化提供相应的安全服务并解决云计算环境变量的动态变化问题,但访问粒度较粗,并且针对的是明文数据;文献[10]在传统的基于角色的访问控制模型(Role-based Access Control,RBAC)基础上给出了一种满足计算及资源节点动态需求的访问控制管理模型(Cloud Administrate based on RBAC,CARBAC),该模型要求数据所有者具有很强的计算能力并时刻在线以维护角色的层次关系以及进行角色指派,因而不能发挥云强大的计算优势且同样只是针对明文数据.

随着云存储系统对数据访问控制在灵活性方面提出了更高的要求,传统的访问控制模型已经不能满足数据细粒度访问控制的需求.2005 年,文献[11]首先提出了基于属性的加密方案(Attribute-based Encryption,ABE),其基本思想是:当用户的私钥以及密文分别对应的属性集合的交集达到一个门限值时才能解密密文.常见的 ABE 访问控制结构如图 1 所示.文献[12]在模糊身份加密方案的基础上提出了密钥策略的基于属性的加密方案(Key-policy Attribute-based Encryption,KP-ABE),其使用访问控制结构加密密钥;2007 年,Bethencourt J 等[13]提出一种密文策略的基于属性的访问控制方法(Ciphertext-Policy Attribute-based Encryption,CP-ABE),该方法使用访问控制结构加密明文.在用户访问权限被撤销时,KP-ABE 和 CP-ABE 算法都要求 DO 对数据重新加密.然而 ABE 算法效率较低,重加密代价很大.如何有效地支持动态策略,已成为 ABE 面临的主要难题.为解决上述问题,文献[14]采用代理重加密方法使得重加密过程由云服务器来完成且保证数据不会泄露,懒惰加密方法则提高了重加密的效率.文献[15]将用户域分成私人域和公共域,私人域采用 CP-ABE 访问控制,公共领域采用等级多信任机构来管理属性和密钥,减少了管理复杂度.以上方案共同的优点[16]是可以进行细粒度的访问控制,不足则是要求用户对文件或数据加密,加重客户端的负担,且未能充分利用云端超强的计算能力.

一方面,属性加密的计算密集性是制约 ABE 访问控制灵活性高效性的因素,包括配对操作以及幂运算.文献[17]首次提出了外包解密 ABE 密文的方案.这类似于代理重加密的思想,把再加密密钥给不可信的代理,将用户 A 公钥加密消息的密文转化成用户 B 的私钥可以解密的密文.用户解密时,代理服务器首先将 ABE 密文转化成 ElGamal 类型密文并发送至用户,用户即可以一种高效计算的方式完成解密过程.然而,文献[17]并未针对 DO 的计算开销采取措施;2014 年,Muhammad Asim 等[26]提出一种属性加解密外包方案,用于减少客户端的计算开销,但并未给出云存储数据访问控制方案的完整构建,同时存在非授权用户的属性验证问题.

另一方面,以往提出的大多数 ABE 方案[18-19]都只有保护消息私密性的功能,而忽略了对访问结构的保护.2008 年,Nishide 等人[20]提出隐藏策略的加密方案,在保护消息的同时保证了访问结构私密性,但其私钥长度和解密运算双线性配对的次数均随着属性个数的增加而线性增大;2011 年 Lai 等[21]利用子群判定假设在合数阶群中提出了一个新的可以隐藏访问结构的加密方案,并证明是完全安全的.但是为了达到一定的安全级别,合数群的阶相对取的比较大.例如,在基于椭圆曲线的密码方法中,相同的安全级别,合数群的阶至少为 1024bit,而素数群的阶只需要 160bit 即可.因此双线性配对的计算效率在合数群中会比素数群中低很多,差别大约为 50 倍,并且当安全级别提高时,这个差距会更大[22].文献[23]利用双系统密码技术[24]首次在素数群中提出了一个可以隐藏访问结构的 ABE 方案,并且依赖于 D-Linear 假设和 DBDH(Decisional Bilinear Diffie-Hellman)假设[25],在标准模型下证明是完全安全的,并且方案中用户私钥长度和解密过程中双线性对的运算量都为固定值,但由于加解密过程都是在客户端执行,计算开销较大.

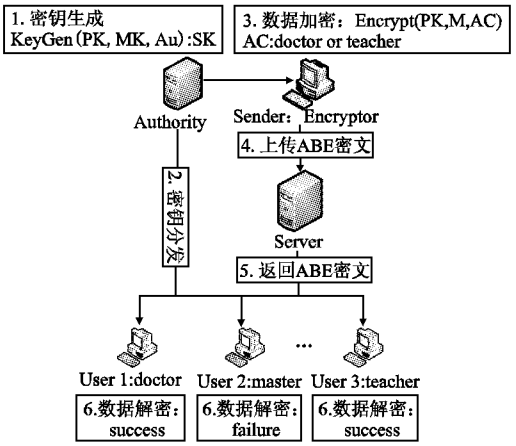

图 1 常见的 ABE 访问结构

Fig. 1 Common access structure for ABE scheme

针对上述 ABE 方案中客户端加解密计算开销大、访问策略的私密性未加保护以及现有方案[26]存在的非授权用户随意进行属性判定进而发起对系统的拒绝服务攻击等问题,本文提出一种基于密文策略的属性加密云数据高效访问控制方案.该方案在 Green 等[17]的 CP-ABE 外包解密方案以及文献[26]的基础上,对加解密外包方案的正确性以及访问策略的隐藏进行了分析,并通过设置授权解密集合来避免非授权用户的属性验证.方案通过部署加解密服务器减轻客户机的加解密计算开销,并通过代理解密服务器的属性验证达到隐藏访问结构的目的.同时结合混合加密技术,给出了完整的更适用于云存储数据的访问控制方案,即隐藏访问结构的外包加解密方案.

3 预备知识

3.1 双线性配对

假设 G_0、G_1 是两个阶为大素数 p 的乘法循环群,定义 e:$G_0 \times G_0 \rightarrow G_1$ 是一个具有下列性质的映射:

1) 双线性:对于所有的 $g_1, g_2 \in G_0$ 以及 $a, b \in Z_p$,都有 $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$;2) 非退化性:存在 $g_0 \in G_0$,使得 $e(g_0, g_0) \neq 1$;3) 可计算性:e 可以被高效计算出来.

其中,$e(*, *)$ 为对称操作,即 $e(g_1^a, g_2^b) = e(g_1^b, g_2^a)$.

3.2 复杂性假设

定义 1. 双线性 Diffie-Hellman(Bilinear Diffie-Hellman,BDH)参数生成器.若一个以安全参数 k(k>0)为输入的随机算法 Γ,在以 k 的多项式时间内运行,输出关于两个群 G_0 和 G_1 的描述、它们共同的素数阶 p 以及一个可有效计算的双

线性映射 $e:G_0\times G_0\to G_1$,则称 Γ 为一个 BDH 参数生成器.

定义 2. 可计算 Diffie-Hellman(Computational Diffie-Hellman,CDH)问题. 随机选择 $a,b\in Z_p^*$,给定三元组(g,g^a,g^b),计算 g^{ab}.

定义 3. 判定双线性 Diffie-Hellman(Decisional Bilinear Diffie-Hellman,DBDH)问题. 令 g 为 G_0的生成元,G_0、G_1和 e 为定义 1 中参数生成器 Γ 的输出. DBDH 问题定义如下:随机选择 $a,b,c\in Z_p^*$. 给定元组(g,g^a,g^b,g^c,R),判断等式 $e(g,g)^{abc}=R$ 是否成立.

3.3 访问树

令 τ 是和访问策略相关的访问树,τ 中叶子节点 K 代表属性集 $\omega\in\Omega$ 的一个属性,Ω 为属性全集;τ 中的非叶子节点 k 代表一个阈门,由其子节点以及门限值来描述;num_k代表节点 k 的子节点数目,T_k为其门限值,有 $0<T_k\leqslant num_k$. 如果 $T_k=1$,则 k 对应一个或门;如果 $T_k=num_k$,则节点 k 对应一个与门. 对于所有的叶子节点,$T_k=1$. 定义函数 parent(k)返回一个节点 k 的父节点,att(K)返回叶子节点 K 相关的属性. 此外,定义 τ 中非叶子节点的子节点之间的序列如下:将节点 k 的子节点从 1 到 num_k进行标号,函数 index(κ)返回子节点 κ 的序列值(即标号).

3.4 Shamir 秘密共享方案

Shamir 的秘密共享方案是信息理论安全的. 待共享的秘密 S 被分成 n 份,即 $S_1,S_2,\cdots,S_{n-1},S_n$. 只有获得大于或者等于 d 份才能重构 S. 令 $y=f(x)$是一个次数为 $d-1$ 的多项式. 方案由两部分组成:1) 共享分配. 秘密 $c_0=S$ 在用户之间进行分发,通过给每个用户 U_i分发随机多项式 $f(x)$的 d 个共享中的一个,该多项式使用 c_0以及 $d-1$ 个随机选择的系数 c_1,$c_2,\cdots,c_{d-1}$;2) 秘密重构. 任意 d 个或者更多的用户通过结合他们各自不同的共享来构建多项式 $f(x)$(可通过多项式插值法实现,比如使用拉格朗日插值法)即可重构 S. 秘密通过 $f(0)=c_0$给出.

4 方案设计

4.1 系统框架

ABE 加解密过程的计算复杂度加大了客户端的计算开销. 为此,本文在文献[26]的基础上提出改进的加解密外包的属性加密高效访问控制方案,其系统框架如图 2 所示.

方案在常见的 ABE 访问控制框架的基础上增加了两个半可信的代理服务器. 其中,用于加密外包的服务器代理记为 Proxy A,用于解密外包的代理服务器记为 Proxy B. 在数据加密阶段,DO 将半加密的密文发送给 Proxy A,同时指定访问控制结构,然后由 Proxy A 生成相应的访问控制策略,并形成最终的密文以减轻客户端加密过程的计算开销;在数据解密阶段,Proxy B 首先将密文解密成 ElGamal 类型密文以减轻客户端解密过程的计算开销.

本文基于 CP-ABE 的加解密外包方案给出完整的云数据访问控制方案. 其中,用对称密钥加密文件以提高效率,用属性密钥加密对称密钥实现访问控制. DO 首先根据逻辑关系,将需要上传的信息分类,然后用不同的对称密钥分别进行加密. 同时为每个文件设定访问策略,按照该策略对相应的对称密钥进行属性加密. 对于解密用户,属性授权机构(Attribute Authority ,AA)根据其身份,对每个用户属性集进行认证,并

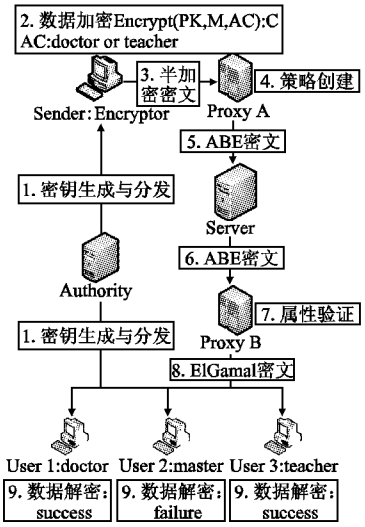

图 2 加解密外包的 ABE 访问结构

Fig. 2 Encryption and decryption outsourcing access structure for ABE scheme

产生相应的属性私钥. 只有持有该私钥的用户所具有的属性满足访问策略时才能正确解密密文,得到对称密钥 Ψ. 方案的高效性体现在属性加解密外包上,同时支持访问策略的隐藏:解密用户无论能否解开密文,均对该密文的访问结构一无所知,即策略对于用户而言是透明的. 另外,针对现有方案中存在的因大量非授权用户的属性判定而带来的计算开销以及发起对系统的拒绝服务攻击等问题,本文采用设置授权用户集合的方法,只有用户的身份编号在授权范围内才能执行属性判定过程.

4.2 隐藏策略的 ABE 加解密外包方案

在该方案中,主机或者用户将加解密的部分功能外包给两个相互独立的半可信的实体,即代理加密服务器 Proxy A 和代理解密服务器 Proxy B. 接下来定义一个双线性映射 $e:G_0\times G_0\to G_1$以及两个抗同谋攻击的哈希函数:

$$H_1:\{0,1\}^*\to G_0,H_2:G_1\to\{0,1\}^n.$$

其中,安全参数 λ 决定群组的大小,$\{0,1\}^*$代表一个任意长度的二进制串. 对 $x,v\in Z_p$,我们定义拉格朗日系数 $\Delta_{v,\Lambda}(x)=\prod_{j\in\Lambda,j\neq v}\frac{x-j}{v-j}$,其中 Λ 是 Z_p中的一个元素集合.

1) Setup(λ):初始化过程. 这个算法由可信的属性授权机构 AA 运行来产生系统变量,即公共密钥 PK 和主私钥 MK. 算法选择一个随机的生成元 $g\in G_0$,阶为素数 p,以及随机的变量 $\alpha,\beta,\gamma,\theta,\delta\in Z_p$. 另外,其选择加密的哈希函数 H_1以及 H_2,并设置 $A=e(g,g)^{\alpha\beta}$. 公共密钥 PK 和主私钥 MK 分别为:

$$PK=(G_0,G_1,g,g^\beta,g^\delta,A,H_1,H_2)$$

$$MK=(g^\alpha,\beta,\gamma,\theta,\delta)$$

2) KeyGeneration$_d(MK,PK,\omega)$:代理解密密钥以及用户解密密钥产生阶段. 该算法同样由可信的授权机构运行,其用来为属性集为 $\omega\in\Omega$ 的用户产生私钥. 算法选择随机的变量

$r_u, z \in Z_p$并设置 $SK_\omega = (SK_{pdec}, SK_{udec})$. 其中,

$$SK_{pdec} = (D^{(1)} = (g^\alpha g^{\gamma\theta r_u})^{\frac{1}{z}} = g^{\frac{\alpha}{z}} g^{\gamma\theta t},$$

$$D^{(2)} = (g^{\gamma\beta r_u})^{\frac{1}{z}} = g^{\gamma\beta t},$$

$$\forall a_j \in \omega: D_j^{(3)} = (H_1(a_j)^{\gamma r_u})^{\frac{1}{z}} = H_1(a_j)^{\gamma t})$$

令 $t = \frac{r_u}{z}$,则有 $SK_{udec} = z$.

其中,SK_{udec}是由解密用户持有的 ElGamal 类型的私钥,而 SK_{pdec}是一个转换密钥,由半可信的 Proxy B 共享. 在用户申请数据阶段,系统将计算开销较大的策略估算过程(验证解密用户属性是否满足访问结构)转移到 Proxy B 上执行,并将半解密密文(即 ElGamal 类型密文)返回给用户.

3) $\text{KeyGeneration}_e(MK, PK)$:加密密钥的产生. 该算法为 Proxy A 产生私钥. 该代理将使用这个私钥来创建由数据属主指定的访问策略:

$$SK_{penc} = \frac{\theta}{\delta}$$

Proxy A 使用这个密钥来恢复在访问结构 τ 中共享的分量进而计算加密访问策略. 就实际部署而言,若系统需要有多个代理,那么该算法的结果是,每个代理获得一个不同的密钥 SK_{penc}(也就是不同的 δ 将会被使用,即 $SK_{p_i enc} = \frac{\theta}{\delta_i}$).

4) $\text{Encryption}(PK, \Psi, \tau)$:该算法由客户端运行(数据加密者). 为了减轻主机计算负担,该算法仅产生半加密的密文,其包含加密的对称密钥 $\Psi \in G_1$,加密私钥以及授权的属性集. Proxy A 随即使用这些分量计算产生加密策略. 算法选择随机值 $s \in Z_p$,并指定访问授权集合 *List*. 计算一个部分加密密文如下:

$$\widehat{CT} = (C = \Psi \oplus H_2(A^s), List, \overline{C} = g^{\beta s},$$

$$\widetilde{C} = g^{\delta s}, \forall a_j \in \tau: \widetilde{C}_j = H_1(a_j)^{-s})$$

其中,a_j是指属性集合的所有属性.

5) $\text{PolicyCreation}(\widehat{CT}, SK_{penc})$:访问策略创建过程. 该算法由 Proxy A 运行来产生和访问结构 τ 相关的访问策略 $F(\omega)$,其必须和密文 C 相关. Proxy A 执行如下步骤:

a) 解密 $\widetilde{C}$:$C' = \widetilde{C}^{SK_{penc}} = (g^{\delta s})^{\frac{\theta}{\delta}} = g^{\theta s} = g^{\hat{s}}$,其中 $\hat{s} = \theta s$;

b) 创建访问策略 $F(\omega)$:

在该阶段,Proxy A 创建和访问策略 τ 相关的加密分量. 它使用 Shamir 的秘密共享方案在 τ 的叶子节点中来分发 $g^{\hat{s}}$. 算法从根节点 R 开始,自上而下地为每一个在 τ 中的节点 k 选择一个多项式 $q_k(\cdot)$. 首先,对于树中的每个节点 k,其设置该节点的多项式的次数 d_k 比其门限值 T_k 小 1,即 $d_k = T_k - 1$. 然后,从访问树根节点 R 开始,该算法设置 $q_R(0) = C'$(其中 $C' = \widetilde{C}^{SK_{penc}} = g^{\theta s} = g^{\hat{s}}$). 并随机选择多项式 $q_R(\cdot)$的 d_R个其他的点来完整地定义这个多项式. 对于任意一个其他节点 k,算法设置 $q_k(0) = q_{parent(k)}(index(k))$. 并随机选择剩余的 d_k个点来完整地定义 $q_k(\cdot)$.

c) 创建密文. 最终的密文 CT 组成如下:

$$\text{CT} = (C = \Psi \oplus H_2(A^s), List,$$

$$\overline{C} = g^{\beta s}, \forall a_j \in \tau: C_j = g^{\hat{s}_j} H_1(a_j)^{-s})$$

这里,$g^{\hat{s}_j}$即为步骤 b)中 τ 的叶子节点所分享的 $g^{\hat{s}}$的分量.

6) $\text{PolicyVerification}(SK_{pdec}, CT)$:该算法由 Proxy B 执行,用于验证用户的属性集 ω 是否满足密文中定义的访问树 τ. 它以解密用户属性集 ω 相关的转换密钥 SK_{pdec}以及密文 CT 为输入. 若满足则计算并输出一个部分解密的 ElGamal 类型的密文$\overline{CT}$. 但在属性验证之前,算法首先判断用户身份是否在授权集合内,若满足,则进行下述过程,否则不能进行属性验证.

该算法使用递归函数 $\text{DecryptNode}(CT, D_{j,\omega}^{(3)}, k)$,我们分别为叶子节点 K(见下文 a)过程)以及 τ 的内部节点 k(见下文 b)过程)定义了函数.

a) $\text{DecryptNode}(CT, SK_{pdec}, K)$

如前所述,访问树的每个叶子节点都和一个实值的属性相关联,即可令 $j = \text{att}(K)$. 如果 $j \in \omega$,我们有:

$$\begin{aligned}
&\text{DecryptNode}(CT, SK_{pdec}, K)\\
&= e(C_j, D^{(2)}) \cdot e(\breve{C}, D_j^{(3)})\\
&= e(g^{\hat{s}_j} H_1(a_j)^{-s}, g^{\gamma\beta t}) \cdot e(g^{\beta s}, H_1(a_j)^{\gamma t})\\
&= e(g^{\hat{s}_j}, g^{\gamma\beta t}) \cdot e(H_1(a_j)^{-s}, g^{\gamma\beta t}) \cdot e(g, H_1(a_j))^{s\gamma\beta t}\\
&= e(g,g)^{\hat{s}_j \gamma\beta t}
\end{aligned}$$

如果 $j \notin \omega$,那么 $\text{DecryptNode}(CT, SK_{i,\omega}, K) = \perp$. 其中 $\perp$ 代表失败.

b) $\text{DecryptNode}(CT, SK_{pdec}, k)$

对于访问树 τ 的所有内部节点 k 的子节点 κ,该算法称作 $\text{DecryptNode}(CT, SK_{pdec}, \kappa)$,输出为 F_κ,用于决定用户是否有足够的属性来满足策略. 为了满足策略,应当有足够的数据点(也就是满足条件的子节点)在内部节点 k 中重构多项式进而重构 $q_k(0)$. 令 Ω_k为子节点 κ 的任意 T_k大小的集合,即从节点 k 的子节点 $\kappa_1, \kappa_1, \cdots, \kappa_{num_k}$中任意选出 T_k个,其中 $F_\kappa \neq \perp, \forall \kappa \in \Omega_k$. 如果不存在集合 Ω_k,那么节点 k 是不满足的,函数返回 $\perp$. 否则,算法使用多项式插值法,计算下列函数的值:

$$\begin{aligned}
F_k &= \prod_{\kappa \in \Omega_k} F_\kappa^{\Delta_{v,\Omega_k'(0)}}, \text{其中} v = index(\kappa), \Omega_k' = \{index(\kappa), \kappa \in \Omega_k\}\\
&= \prod_{\kappa \in \Omega_k} (e(g,g)^{\hat{s}_j \gamma\beta t})^{\Delta_{v,\Omega_k'(0)}}\\
&= e(g,g)^{\hat{s}_k \gamma\beta t}
\end{aligned}$$

Proxy B 部分解密过程如下:

Proxy B 首先计算访问树 τ 的根节点 R 的 $\text{DecryptNode}(\cdot)$ 函数. 如果函数 $\text{DecryptNode}(CT, SK_{pdec}, R)$返回 $\perp$,那么和代理解密密钥 SK_{pdec}以及用户私钥 SK_{udec}相关联的解密用户的属性集 ω 不满足访问策略 τ,解密失败,算法返回 $\perp$. 否则,则解密算法执行如下:

$$Z^{(1)} = \text{DecryptNode}(CT, SK_{pdec}, R) = e(g,g)^{\hat{s}\gamma\beta t} = e(g,g)^{\theta s\gamma\beta t},$$

$$Z^{(2)} = e(\overline{C}, D^{(1)}) = e(g^{\beta s}, g^{\frac{\alpha}{z}} g^{\gamma\theta t}) = e(g,g)^{\frac{\alpha\beta s}{z}} \cdot e(g,g)^{\theta s\gamma\beta t},$$

$$Z^{(3)} = \frac{Z^{(2)}}{Z^{(1)}} = \frac{e(g,g)^{\frac{\alpha\beta s}{z}} \cdot e(g,g)^{\theta s\gamma\beta t}}{e(g,g)^{\theta s\gamma\beta t}} = e(g,g)^{\frac{\alpha\beta s}{z}}$$

算法最终输出半解密密文$\overline{CT} = (C, Z^{(3)})$,同时 CSP 将相应的加密数据返回给请求用户.

7) $\text{Decryption}(SK_{udec}, \overline{CT})$:该算法由用户执行,即用户解密得到加密数据对称密钥 Ψ 的过程. 算法输入为用户私钥

SK_{udec}以及半解密密文$\overline{CT}$. 对称密钥 Ψ 可恢复如下:

$$\Psi = C \oplus H_2((Z^{(3)})^{SK_{udec}})$$

4.3 混合加密方案

由于数据对称加密和非对称加密计算开销上的差异性,本文在文献[26]的基础上引入层次访问控制方案,即结合了属性加密(非对称加密)和对称加密的混合加密方案. 具体实现如下:

假设每一个用户的解密私钥为 SK_{u_jdec}.

数据加密过程:U_i选择 Ψ 作为对称密钥,用于数据加密. 加密数据项 M 为 $C_M = \{M\}_\Psi$. 然后 U_i利用属性加密方案加密 Ψ,C_Ψ和 C_M一起存档.

数据解密过程:用户 U_j申请访问 U_i的数据,首先向 CSP 发出数据访问请求. 鉴于文献[26]中并未进行数据请求过滤,即用户无论满足解密条件与否都可以进行属性验证过程. 对于非授权用户的数据请求,大量的属性验证过程一方面带来不必要的计算开销,另一方面可能对系统造成拒绝服务攻击. 因此本文首先对用户身份编号进行判定,只有在授权集合内才能够进行属性验证,另外通过设置多台代理服务器解决潜在的拒绝服务攻击问题. 接下来数据请求用户验证自己的属性是否满足用户 U_i设定的访问控制策略. 若不满足,则无法获取 C_Ψ;若满足,则系统返回 C_Ψ和 C_M. 最后 U_j使用其私钥 SK_{u_jdec}解密 C_Ψ得到 Ψ,然后用 Ψ 解密 C_M得到明文信息 M. 其中,密文 C_Ψ是 ElGamal 类型,SK_{u_jdec}解密 C_Ψ过程较为高效.

5 安全性及性能分析

5.1 安全性分析

ABE 本身具有很强的安全性:1) ABE 算法基于椭圆曲线上的双线性对,从密码学理论上来讲,破译密码是困难的;2) ABE 的密文被附上了一个访问结构,这个访问结构的复杂性使得在安全性证明的模拟过程中,模拟者难以将其"嵌入"一个普通简单的困难性假设(如 Diffie-Hellman 假设),这导致了挑战密文的困难;3) ABE 私钥对应一定的属性,不同的私钥属性集合可能具有相交的属性,这样的私钥相关性也给模拟私钥提取问询造成了困难.

现在定义 ABE 外包加解密方案的敌手 $\mathcal{A}$ 和挑战者 C 之间的两个安全游戏. 在第一个游戏中,敌手 $\mathcal{A}$ 向 Proxy A 询问私钥,即 SK_{penc}. 在第二个游戏中,敌手 $\mathcal{A}$ 询问用户的私钥,即 SK_ω. 上述游戏都是基于这样一个假设:模型中 Proxy A 和 Proxy B 不会进行合谋. 基于该假设的目的就是实现较好的职责分离.

Game 1:

Setup:挑战者 C 运行上述 Setup 算法并将公开参数 PK 传给敌手 $\mathcal{A}$,主私钥 MK 自己保留.

Phase 1:敌手 $\mathcal{A}$ 向 Proxy A 进行多项式有限次的私钥问询,即 SK_{penc_i},其中 $\forall i \in \{1,2,\cdots,Q\}$. 挑战者 C 将这些私钥发送给敌手 $\mathcal{A}$.

Challenge:在这个阶段敌手 $\mathcal{A}$ 从希望被挑战的消息空间中选择两个等长的明文消息 M_0、M_1 并提交. 挑战者随机掷硬币 $b \in \{0,1\}$ 并返回给敌手 $\mathcal{A}$ 部分加密的密文 M_b(即该密文并不包含加密策略).

Phase 2:重复 Phase 1 的私钥询问过程,其中问询的内容是 Phase 1 中不曾问到的.

Guess:在这个阶段中,敌手 $\mathcal{A}$ 输出一个猜测 $b' \in \{0,1\}$,如果 $b' = b$ 则敌手 $\mathcal{A}$ 获胜. 在这个攻击游戏中敌手 $\mathcal{A}$ 获胜的概率为 $\left|\Pr[b' = b] - \frac{1}{2}\right|$,即在多项式时间内没有敌手 $\mathcal{A}$ 能够以不可忽略的优势赢得上述游戏.

Game 2:

Setup:挑战者 C 运行上述 Setup 算法并将公开参数 PK 传给敌手 $\mathcal{A}$,主私钥 MK 自己保留.

Phase 1:根据属性集 $\omega_1,\omega_2,\cdots,\omega_Q$,敌手 $\mathcal{A}$ 向用户进行多项式有限次的私钥问询. 挑战者将这些私钥 SK_{ω_i}发送给敌手 $\mathcal{A}$. 其中对 $\mathcal{A}$ 而言,$\forall i \in \{1,2,\cdots,Q\}$.

Challenge:在这个阶段敌手 $\mathcal{A}$ 从希望被挑战的消息空间中选择两个等长的明文消息 M_0,M_1 并提交. 此外,敌手 $\mathcal{A}$ 提交给挑战者 C 一个访问树 τ^*,使得 Phase 1 中被询问的私钥并不满足访问树 τ^*. 挑战者随机掷硬币 $b \in \{0,1\}$ 并返回给敌手 $\mathcal{A}$ 在访问策略 τ^* 下加密的密文 M_b.

Phase 2:重复 Phase 1 的并不满足访问树 τ^* 的私钥询问过程,其中问询的内容是 Phase 1 中不曾问到的.

Guess:在这个阶段中,敌手 $\mathcal{A}$ 输出一个猜测 $b' \in \{0,1\}$,如果 $b' = b$ 则敌手 $\mathcal{A}$ 获胜. 在这个攻击游戏中敌手 $\mathcal{A}$ 获胜的概率为 $\left|\Pr[b' = b] - \frac{1}{2}\right|$,即在多项式时间内没有敌手 $\mathcal{A}$ 能够以不可忽略的优势赢得上述游戏.

定义 4. 一个加解密外包的 ABE-EDO 方案是安全的,如果没有敌手能够在多项式时间内以不可忽略的优势赢得上述 ABE-EDO 安全游戏. 优胜概率定义为 $\left|\Pr[b' = b] - \frac{1}{2}\right|$.

定理 1. 令 q 是一个敌手 $\mathcal{A}$ 从对挑战者 C 的询问中可以接收到的群元素数量的上界,其来自哈希函数 $H(\cdot)$,群 G_0、G_1,双线性映射 $e(\cdot,\cdot)$以及在 ABE-EDO 安全游戏中的互动. 敌手 $\mathcal{A}$ 在安全游戏中的优势为 $O(q^2/p)$.

此外,ABE-EDO 方案的安全性可以使用类似于在 Shoup(1997)、Bethencourt 等(2007)或者是 Boneh 等(2005)中的论据进行证明.

5.2 正确性验证

代理解密服务器 Proxy B 首先计算访问结构 τ 中根节点 R 的 DecryptNode(·)函数. 函数 DecryptNode(CT,SK_{pdec},R)如果返回 $\perp$,那么和密钥 SK_{pdec}以及用户私钥 SK_ω相关联的属性集 ω 不满足 τ. 否则计算得 $Z^{(3)} = \frac{Z^{(2)}}{Z^{(1)}} = \frac{e(g,g)^{\frac{\alpha\beta s}{z}} \cdot e(g,g)^{\theta s\gamma\beta t}}{e(g,g)^{\theta s\gamma\beta t}}$

$= e(g,g)^{\frac{\alpha\beta s}{z}}$,并输出$\overline{CT} = (C, Z^{(3)})$.

用户可用其私钥 SK_{udec}解密如下:

$$\begin{aligned}\Psi &= C \oplus H_2((Z^{(3)})^{SK_{udec}}) \\ &= \Psi \oplus H_2(A^s) \oplus H_2(e(g,g)^{\frac{\alpha\beta s}{z} \cdot z}) \\ &= \Psi \oplus H_2(A^s) \oplus H_2(A^s) \\ &= \Psi\end{aligned}$$

正确性得证.

5.3 性能分析

该部分从两方面对方案的性能进行分析:

设$|m|$为待上传文件对称加密后的大小,数据加密涉及到的属性个数为$|A_c|$,系统属性个数为$|A_u|$,某用户属性个数为$|A_w|$.

5.3.1 客户机对文件进行加解密时的计算量分析

加密时,除了对需要上传的明文数据进行对称加密之外,还需要对用于加密明文数据的对称密钥进行公钥加密.设t_p为1次配对操作(Pairing Operations)所用的时间,t_m为一次求幂操作(Exponentiation Operations)所用的时间.其中计算C时需要1次配对操作和1次幂操作,求$\overline{C}$时需要进行1次幂运算,求$\widetilde{C}$时需要进行1次幂运算,在对$\widetilde{C}_j$运算时需要进行1次幂运算,因此加密时DO共需进行$(3|A_c|+3)$次幂运算,1次配对操作.用户解密时,只需进行1次幂运算就行.最终*DO*加密时间开销为$(3|A_c|+3)t_m+t_p$,用户解密时间开销为t_m.

5.3.2 传输过程的通信消耗分析

DO将密文加密后,需要传给Proxy A.这一阶段通信开销为半加密密文$\widehat{CT}$的大小.令C的大小为$|G_T|$,$\overline{C}$和$\widetilde{C}$的大小均为$|G|$,每个$\widetilde{C}_j$的大小为$|G|$,$|\mathrm{List}|$为访问列表的大小.半加密密文$\widehat{CT}$大小即为$(|A_c|+2)|G|+|G_T|$.

CSP将密文CT发送给Proxy B,留待对解密用户进行访问申请的判定.该阶段通信开销如下:C的大小为$|G_T|$,$\widetilde{C}$的大小为$|G|$,每个$\widetilde{C}_j$的大小为$|G|$.密文CT大小即为$(|A_c|+1)|G|+|G_T|$.

最后,当解密用户属性满足访问结构时,Proxy B将半解密密文$\overline{CT}$发送给解密用户.该阶段通信开销如下:C的大小为$|G_T|$,$Z^{(3)}$大小为$|G_T|$.半解密密文$\overline{CT}$大小即为$2|G_T|$.

综上,整个传输过程的通信消耗为$(2|A_c|+3)|G|+4|G_T|$.但是考虑到实际传输(如用户到CSP的物理距离因素)影响,且从用户A到Proxy A再到服务器传输文件大小几乎没变,故取$(|A_c|+3)|G|+2|G_T|$.

本文的通信消耗与计算消耗同之前的属性加密方案相比较如表1所示.

表1 通信开销和计算开销对比

Table 1 Comparison for communication and computation overhead

方案	通信开销	用户加密时间开销	用户解密时间开销
文献[17]	$(3l+1)\|G\|+2\|G_T\|$	$(3l+2)t_m+t_p$	t_m
文献[19]	$(2\|A_c\|\cdot\|A_u\|+2)\|G\|+2\|G_T\|$	$(\|A_c\|\cdot\|A_u\|+2)t_m+t_p$	$(\|A_c\|+2)t_m+t_p$
文献[26]	$(\|A_c\|+3)\|G\|+2\|G_T\|$	$(\|A_c\|+3)t_m+t_p$	t_m
文献[27]	$(4\|A_c\|+2)\|G\|+2\|G_T\|$	$(3\|A_C\|+2)t_m+t_p$	$\|A_w\|\cdot t_m+t_p$
本文	$(\|A_c\|+3)\|G\|+2\|G_T\|+\|List\|$	$(\|A_c\|+3)t_m+t_p$	t_m

为测试本文方法的功能和性能,搭建了如下仿真实验环境.其中,Proxy A、Proxy B、数据存储服务器及AA配置:Intel(R) Core(TM) i7-4770 CPU,16GB内存,32GB+2TB 7200转硬盘,运行Windows 8.1 64bit操作系统;客户端配置:Intel酷睿i5 CPU,4GB内存,1TB 7200转硬盘,运行Windows 8.1 64bit操作系统.实验中数据对称加密算法采用128bit AES密钥,且主要考虑数据加解密过程的计算开销随着属性数目增长变化的情况以及对系统性能的影响,忽略了数据在分布式网络中的传输延迟.通过仿真实验综合对比了本方案与文献[19,27]在客户端数据加密以及数据解密2个方面的性能,实验结果分别如图3、图4所示.

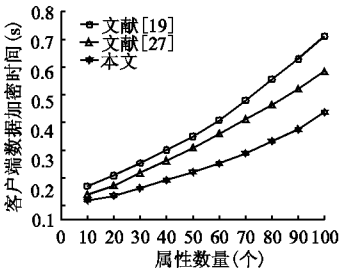

图3 数据加密计算开销与属性数量的变化关系

Fig. 3 Relationship between computation overhead in data encryption and the number of attributes

从理论分析及实验结果可以看出,本文中用户加解密时间开销跟其他几个方案相比有所降低,可以有效地提高用户的加解密效率;从表1可以看出,通信开销相比其他几个方案相比也有一定程度减小.并且随着互联网的发展,数据传输的通信开销也将逐步弱化.另外,在保护数据上传用户的访问结构私密性上,方案[19,27]均未涉及.和文献[26]相比,虽然通信开销略微增加(多了一个访问列表),但针对其存在的非授权用户的属性验证请求问题,本文给出了有效的处理方法.因此,本方案更适合于云环境下的数据访问控制.

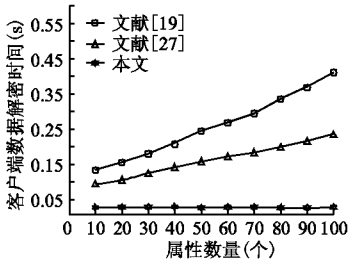

图4 数据解密计算开销与属性数量的变化关系

Fig. 4 Relationship between computation overhead in data decryption and the number of attributes

5.4 隐藏策略分析

就访问策略而言,无论是基本的基于门限的ABE方案还是基于访问树(访问控制结构)的ABE方案,均为明文信息,这在一定程度上损害了DO的数据隐私,并可能因此引发同谋攻击.虽然有文献在保护用户访问结构隐私方面做了很多工作,比如文[23]利用双系统密码技术[24]首次在素数群中提

出了一个可以隐藏访问结构的基于属性加密方案,并且依赖于 D-Linear 假设和 DBDH 假设[25],且在标准模型下证明是完全安全的,但在属性加解密效率上没有提供更高效的解决方法. 本文在采用加解密外包提高效率的同时也实现了访问策略的隐藏,相应分析如下.

在用户解密之前,首先由代理解密服务器 Proxy B 运行 PolicyVerification(SK_{pdec}, CT)算法,详见具体方案部分描述. 其使用递归函数 DecryptNode(CT, $D_{j,\omega}^{(3)}$, k),验证用户的属性集 ω 是否满足访问树 τ,若满足则计算并输出一个部分解密的 ElGamal 类型的密文 $\overline{CT}$. 否则返回 $\perp$,用户无法完成解密过程.

方案通过部署代理解密服务器实现访问策略的隐藏. 和先前的 ABE 方案相比,解密过程不再是通过客户端获取访问策略的明文信息进行属性验证,而是由代理解密服务器完成. 另外,方案通过设置授权用户集合来避免对非法用户的属性验证. 这种情况下,非授权用户无法获取密文,对访问策略信息也一无所知;合法用户只能判断自己是否可以解密,但无法判断还有哪些其他用户可以解密,进而达到了较强的匿名性.

6 结论及下一步工作

本文提出一种改进的数据外包的属性加密方案,并结合混合加密技术使之适用于云存储数据访问控制. 方案依赖于两个半可信、但非共谋的代理服务器,分别用于代理计算开销较大的加密运算和解密运算. 在加密过程中,主机和加密代理服务器相关联,并为指定的属性集创建加密策略分量,加密代理无法恢复原始信息且强制使用给定的属性;在解密过程中,解密代理服务器用于进行策略评判. 当解密用户拥有授权的属性集,该代理将原始密文转换成 ElGamal 类型的密文,方便用户高效解密. 同时,该方案支持访问控制策略的隐藏,对于用户指定的访问结构私密性起到很好的保护作用. 另外,针对云存储用户的变动以及属性撤销带来的大量密文重加密问题,如何构建高效的支持属性撤销的云数据访问控制方案将是下一步的研究重点.

References:

[1] CNET. Who owns your files on Google drive? [EB/OL]. http://news. cnet. com/8301-10233-57420551-93/who-owns-your-files-on-google-drive/,2013.

[2] Dropbox. Dropbox privacy policy [EB/OL]. https://www. dropbox. com/privacy,2013.

[3] Google. Google terms of service [EB/OL]. http://www. google. com/policies/terms/,2013.

[4] Apple Inc. Apple privacy policy (Covering iCloud) [EB/OL]. http://www. apple. com/privacy/,2013.

[5] Microsoft. Microsoft services agreement [EB/OL]. http://widows. microsoft. com/en-US/windows-live/microsoft-service-agreement,2012.

[6] wired. com. Dropbox left user accounts unlocked for 4 hours sunday [EB/OL]. http://www. wired. com/threatlevel/2011/06/dropbox/, 2013.

[7] Twitter. Tweetdeck [EB/OL]. http://money. cnn. com/2012/03/30/technology/tweetdeck-bug-twitter/,2013.

[8] Verizon:2015 Data Breach Investigations Report [EB/OL]. http://www. freebuf. com/news/64183. html,2015.

[9] Jung Y, Chung M. Adaptive security management model in the cloud computing environment [C]. Proceedings of the 12th International Conference on Advanced Communication Technology, Washington DC: IEEE Press,2010:1664-1669.

[10] Yang Liu, Tang Zhuo, Li Ren-fa, et al. Roles query algorithm in cloud computing environment based on user require [J]. Journal on Communications,2011,32 (7):169-175.

[11] Sahai A, Waters B. Fuzzy identity-based encryption [C]. Proceedings of Eurocrypt 2005, Berlin,2005:457-473.

[12] Goyal V, Pandey O, Sahai A, et al. Attribute based encryption for fine-grained access control of encryption security data [C]. Proceedings of the 2006 ACM Conference on Computer and Communications Security, Alexandria, Virginia, USA: ACM Press,2006:89-98.

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

[14] Yu S, Wang C, Ren K, et al. Achieving secure, scalable, and fine-grained data access control in cloud computing [C]. Proceedings of IEEE INFORCOM 2010, San Diego, CA: IEEE Press,2010:1-9.

[15] Chen Dan-wei, Shao Ju, Fan Xiao-wei, et al. MAH-ABE based privacy access control in cloud computing [J]. Acta Electronica Sinica,2014,42(4):821-827.

[16] Feng Chao-sheng, Qin Zhi-guang, Yuan Ding, et al. Key techniques of access control for cloud computing [J]. Acta Electronica Sinica, 2015,2(2):312-319.

[17] Green M, Hohenberger S, Waters B. Outsourcing the decryption of ABE ciphertexts [C]. In USENIX Security,2011:523-538.

[18] Herranz J, Laguillaumie F, Rafols C. Constant size ciphertexts in threshold attribute-based encryption [C]. In PKC 2010, LNCS 6065,2010:19-34.

[19] Attrapadung N, Libert B, Panafieu E. Expressive key-policy attribute-based encryption with constant-sizeciphertexts [C]. In PKC 2011, LNCS 6571,2011:90-108.

[20] Nishide T, Yoneyama K, Ohta K. Attribute-based encryption with partially hidden encryptor-specified access structures [C]. In ACNS 2008, LNCS 5037,2008, :111-129.

[21] Lai J, Deng R H, Li Y. Fully secure cipertext-policy hiding CP-ABE. [J]. Lecture Notes in Computer Science,2011,6672(2):24-39.

[22] Freeman D M. Converting pairing-based cryptosystems from composite-order groups to prime-order groups [C]. In EUROCRYPT 2010, LNCS 6110,2010:44-61.

[23] Wang Hai-bin, Chen Shao-zhen, Attribute-based encryption with hidden access structures [J]. Journal of Electronics & Information Technology,2012,34(2):457-461.

[24] Lewko A, Waters B. New techniques for dual system encryption and fully secure HIBE with short ciphertexts [EB/OL]. http://eprint. Iracr. org/2009/482,2009.

[25] Waters B. Dual system encryption: realizing fully secure IBE and HIBE under simple assumptions [C]. In CRYPTO 2009, LNCS 5677,2009:619-636.

[26] Muhammad Asim, Milan Petkovic, Tanya Ignatenko. Attribute-based encryption with encryption and decryption outsourcing [C]. 2th Australian Information Security Management Conference,2014:21-28.

[27] Waters B. Ciphertext-policy attribute-based encryption: an expressive, efficient, and provably secure realization [C]. International Association for Cryptologic Research, Springer Berlin Heidelberg, 2011:53-70.

附中文参考文献:

[10] 杨 柳,唐 卓,李仁发,等. 云计算环境中基于用户访问需求的角色查找算法[J]. 通信学报,2011,32(7):169-175.

[15] 陈丹伟,邵 菊,樊晓唯,等. 基于 MAH-ABE 的云计算隐私保护访问控制[J]. 电子学报,2014,42(4):821-827.

[16] 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,2015,2(2):312-319.

[23] 王海斌,陈少真. 隐藏访问结构的基于属性加密方案[J]. 电子与信息学报,2012,34(2):457-461.