

# 云环境中基于属性加密的安全接入控制

蔡炎伟, 白光伟, 沈 航, 朱德剑

(南京工业大学 计算机科学与技术学院, 江苏 南京 211816)

**摘 要:** 云存储技术受到越来越多的关注, 但云存储中的安全问题是目前云计算推广的最大障碍. 对此提出新的云环境下的接入控制方案, 根据用户的特点, 将用户分类为个人用户和公共用户, 对不同种类的用户采用不同的分享方式, 并实现用户的撤销. 对公共用户的分享采用多授权中心的方式, 降低授权中心的负担和信息泄露的风险. 最后分析方案的安全性并和其他方案的效率进行比较, 结果表明提出的方案可以有效地解决云存储中的安全问题.

**关键词:** 云存储; 安全; 多授权中心

**中图分类号:** TP393

**文献标识码:** A

**文章编号:** 1000-7180(2017)10-0091-06

## Attribute-Based Encryption Secure Access Control in Cloud

CAI Yan-wei, BAI Guang-wei, SHEN Hang, ZHU De-jian

(Department of Computer Science and Technology, Nanjing Tech University, Nanjing 211816, China)

**Abstract:** As cloud storage technology attracts more and more attention, Security in cloud storage is now the biggest obstacle of cloud computing promotion. This paper proposes a new access control scheme in cloud. According to the characteristics of the users, they are classified as private users and public users. Use different ways to share the files and achieve the revocation of users. The sharing of files to public users adopts a multi-authority solution to reduce the burden of authorities and the risk of leaking information. Finally, analyze the security of the scheme and compare the efficiency of this scheme with those of other schemes. The results show that the proposed scheme can effectively solve the security problem in cloud storage.

**Key words:** cloud storage; security; multi-authority

### 1 引言

云存储是在云计算概念上衍生发展出来的一个新的概念. 它是指通过集群应用、网格技术或分布式文件系统等功能, 将网络中大量各种不同类型的存储设备通过应用软件集合起来协同工作, 共同对外提供数据存储和业务访问功能的一个系统, 保证数据的安全性, 并节约存储空间.

然而, 将文件上传到云服务器存在着许多安全风险<sup>[1]</sup>. 一个可行的解决方案是在上传文件之前先进行加密, 由文件所有者来决定如何加密文件, 以及哪些用户有权限读取文件. 文件所有者还应该可以对用户进行撤销.

文献[2-3]将 KP-ABE 用于云环境下的数据安全存储. 数据所有者可以将数据加密后存储在云端, 并通过发送密钥的方式共享数据给多个用户, 不需要通过其他的授权中心. 数据所有者还可以通过将更新密文和用户私钥委托给云端服务器的方式, 十分有效地实现用户撤销. 由于个人用户一般是所有者认识的人, 可以直接由所有者自己来验证, 所以非常适合采用该方案, 由所有者自己直接来分配用户的权限.

本文将用户分为个人用户和公共用户, 实现对云环境下的安全存储. 对于公共用户, 我们采用多授权中心的方式来提高安全性, 每个授权中心只负责用户的部分属性, 并结合代理重加密实现对用户的

撤销. 对于个人用户, 采用文献[2-3]中提出的方案, 由文件所有者直接分配用户权限.

## 2 预备知识

为了更好地阐述本文所提出的方案, 对相关的密码学理论知识进行介绍, 具体如下.

(1) 双线性对映射. 设  $p$  是一个大素数,  $G_1$  和  $G_2$  是 2 个有着相同阶数  $p$  的乘法循环群.  $G_1$  和  $G_2$  的生成元分别为  $g_1$  和  $g_2$ .  $e$  是一个双线性对映射, 即  $G_1 \times G_2 \rightarrow G_T$ .

① 双线性: 对于所有的  $u \in G_1, v \in G_2$ , 以及所有的  $a, b \in \mathbb{Z}_p, e(u^a, v^b) = e(u, v)^{ab}$  均成立.

② 可计算性: 对于所有的  $u \in G_1, v \in G_2$ , 都存在一个有效的算法计算出映射  $e(u, v)$ .

③ 非退化性: 存在  $a, b \in \mathbb{Z}_p$ , 使得  $e(g_1^a, g_2^b) \neq 1$ .

(2) 代理重加密. 在云计算中, 云计算服务提供商作为代理人, 用户 A 无法完全相信, 因此需要将自己存储的数据在本地用自己的公钥  $P_a$  加密后再进行上传, 这样云计算服务商就无法得到其数据的明文信息, 而该数据只有用户 A 使用自己的私钥  $S_a$  才能解密. 当用户 A 需要把该数据与用户 B 共享时, 他可以根据自己的一些信息(如私钥)及用户 B 的公钥  $P_b$  计算一个转换密钥  $Rk$ , 由云计算服务商使用转换密钥  $Rk$ , 将针对用户 A 的密文重加密得到针对用户 B 的密文, 这样, 用户 B 可以容易地从云中下载该密文数据, 使用自己的私钥  $S_b$  即可解密<sup>[4]</sup>.

(3) KP-ABE. 访问结构是安全系统研究的术语, 系统的访问结构是指被授权的集合的结构<sup>[5]</sup>. KP-ABE(key policy attribute based encryption, 密钥策略属性基加密系统)是指, 密钥对应于一个访问控制, 而密文对应于一个属性集合, 当且仅当属性集合中的属性能够满足此访问结构才能解密密文<sup>[6]</sup>.

## 3 系统模型

系统由多个部分组成: 个人用户、公共用户、文件所有者、多个授权中心、云端服务器.

文件所有者将自己的文件加密之后上传到云端服务器, 文件分为个人文件和公共文件. 个人用户一般是所有者熟悉的人, 比如亲人或朋友等. 个人用户从云端服务器下载加密之后的个人文件, 并向所有者发送请求, 如果所有者同意, 将会发送解密用的私钥给对方. 公共用户一般是单位的工作人员, 比如公司的员工或者是医院的医生等. 公共用户从云端服

务器下载加密之后的公共文件, 只有符合所有者添加的用户属性条件的用户才可以正确解密文件. 授权中心一般有多个, 每个授权中心验证用户的一部分属性, 比如验证用户所属单位的授权中心, 验证用户职位的授权中心等. 多个授权中心可以减小单个授权中心的负担, 每个授权中心只负责验证一部分信息也可以提高系统的安全性.

### 3.1 个人文件的上传和获取

加密后的文件在逻辑上采取树状方式存储在云端, 类似于个人电脑中的文件夹的形式. 个人文件的存储结构可以由用户自己设置, 也可以采用系统推荐的设置. 文件存储的位置位于存储结构树的叶子结点. 个人用户向所有者请求以某一个结点为根节点的子树的文件, 所有者同意之后发送包含访问权限结构的密钥给对方, 个人用户可以解密以请求的结点为根节点的所有叶子结点存储的文件.

### 3.2 公共文件的上传和获取

授权中心负责用户属性的验证, 每个授权中心负责验证用户的一部分属性. 比如第  $k$  个授权中心  $A_k$  负责验证用户的职位, 第  $j$  个授权中心  $A_j$  负责验证用户所属的单位, 用  $N$  来表示授权中心的个数. 所有者通过添加用户属性的方式来决定可以解密文件的用户, 这里以公司财务文件举例, 所有者添加用户属性公司为 A, 部门为财务科, 职位为财务总监或财务会计, 表示只有 A 公司财务科的财务总监或会计才可以查看文件的内容. 公司财务文件存储结构如图 1 所示, 存储结构可以由文件所有者自己设置, 也可以采用系统推荐的设置. 加密后的公共文件也只可以存储在叶子结点中.

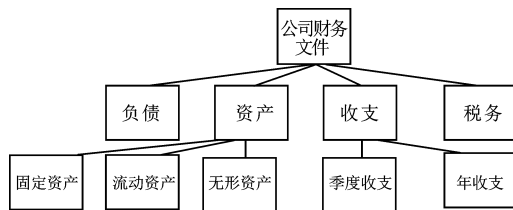


图 1 公司财务文件存储结构

## 4 算法设计

本节介绍公共文件的上传和获取的算法设计.

### 4.1 初始设置

系统初始化阶段, 由一个授权中心选择两个乘法循环群  $G_1$  和  $G_2$ , 分别将  $g_1$  和  $g_2$  作为底数, 并选择一个素数  $p$  作为阶数, 选择之后在授权中心之间公布. 各个授权中心随机选择一个  $v_k \in \mathbb{Z}_p (k \in \{1,$

$\dots, N\}$ ), 并计算出  $Y_k = e(g_1, g_2)^{v_k}$ , 然后发送给所有其他的授权中心, 最后相乘得到结果为

$$Y = \prod_{k \in \{1, \dots, N\}} Y_k \quad (1)$$

任意两个授权中心  $A_k (k \in \{1, \dots, N\})$  和  $A_j (j \in \{1, \dots, N\} \setminus \{k\})$  任意选择一个  $s_{kj} \in Z_p$ ,  $s_{kj}$  由授权中心  $A_k$  和  $A_j$  共享, 并且对其他授权中心保密. 我们定义  $s_{kj} = s_{jk}$ .

每个授权中心随机选择一个  $x_k \in Z_p (k \in \{1, \dots, N\})$ , 并计算出  $y_k = g_1^{x_k}$ .  $u$  为用户的唯一全球身份标识, 我们定义  $PRF_{kj}(u)$  为

$$PRF_{kj}(u) = g_1^{x_k x_j / (s_{kj} + u)} \quad (2)$$

所有者可以选择对某些授权中心所管理的属性不作要求. 例如只对公司和部门作限制, 公司为  $A$ , 部门为  $IT$  部, 对职位并无要求. 负责验证的授权中心往往包含很多属性, 对职位来说, 有工程师、会计、部门经理等等. 按照一般的方式, 只能采取工程师或会计或部门经理或……的结构, 极大地增加了系统的开销. 针对该问题, 我们向每个授权中心加入一个虚拟属性  $A_k^* (k \in \{1, \dots, N\})$ , 用该属性表示所属授权中心任意一个属性都可以. 用  $O_k$  表示第  $k$  个授权中心所管理的属性的集合, 对于属性  $i \in O_k$ , 任意选择一个  $t_{k,i} \in Z_p$ , 并且计算出  $T_{k,i} = g_2^{t_{k,i}}$ .

每一个授权中心定义一个版本号  $\mathcal{ver}$ , 初始值为 1. 定义  $MK_k$  为

$$\langle x_k, \{s_{kj}\}_{j \in \{1, \dots, N\} \setminus \{k\}}, \{t_{k,i}\}_{i \in O_k} \rangle \quad (3)$$

$A_k$  得到属于自己的私钥  $(\mathcal{ver}, MK_k)$ . 定义  $PK$  为

$$PK = \langle Y = \prod_{k \in \{1, \dots, N\}} Y_k, \{y_k, \{T_{k,i} = g_2^{t_{k,i}}\}_{i \in O_k}\}_{k \in \{1, \dots, N\}} \rangle \quad (4)$$

系统发布公钥为  $(\mathcal{ver}, PK)$ .

#### 4.2 私钥生成

用户与每个授权中心交互, 组成自己的私钥. 用户与  $A_k$  交互, 对于  $j \in \{1, \dots, N\} \setminus \{k\}$ ,  $A_k$  随机选择一个  $R_{kj} \in Z_p$ , 我们定义  $D_{kj}$  为:

$$D_{kj} = \begin{cases} g_1^{R_{kj}} PRF_{kj}(u), & \text{如果 } k > j \\ g_1^{R_{kj}} / PRF_{kj}(u), & \text{其他} \end{cases} \quad (5)$$

$A_k$  随机选择一个多项式  $p_k$ , 该多项式满足以下条件:

$$p_k(0) = v_k - \sum_{j \in \{1, \dots, N\} \setminus \{k\}} R_{kj} \quad (6)$$

对于用户的有效属性  $i$ ,  $A_k$  计算出参数  $S_{k,i} = g_1^{p_k(i)/t_{k,i}}$ . 定义  $R_u$  为

$$R_u = \sum_{(k,j) \in \{1, \dots, N\} \times (\{1, \dots, N\} \setminus \{k\})} R_{kj} \quad (7)$$

定义  $D_u$  为

$$D_u = \prod_{(k,j) \in \{1, \dots, N\} \times (\{1, \dots, N\} \setminus \{k\})} D_{kj} = g_1^{R_u} \quad (8)$$

用  $A_u^k$  表示身份标识为  $u$  的用户的属性集合与  $O_k$  的交集, 定义  $SK_u$  为

$$SK_u = \langle D_u, \{S_{k,i}\}_{i \in A_u^k, k \in \{1, \dots, N\}} \rangle \quad (9)$$

用户得到的私钥为  $(\mathcal{ver}, SK_u)$ .

#### 4.3 文件加密

文件所有者选择加密密钥  $M$ , 并为所有授权中心选择属性. 如果为一个授权中心选择多个属性, 他们之间的关系是或. 不同授权中心之间的关系是与. 比如为验证公司的授权中心选择公司  $A$ , 为验证职位的授权中心选择财务总监和会计, 为验证部门的授权中心选择财务部, 其他授权中心属性任意, 即选择虚拟属性, 那么可以解密密文的用户是  $(\text{公司} = A) \vee (\text{部门} = \text{财务部}) \vee ((\text{职位} = \text{会计}) \wedge (\text{职位} = \text{财务总监}))$ . 选择  $s \in Z_p$ , 得到密文  $CT$  为

$$CT = \langle \mathcal{ver}, E_0 = M \cdot Y^s, E_1 = g_2^s, \{C_{k,i} = T_{k,i}^{s_{k,i}}\}_{i \in A_k^c, k \in \{1, \dots, N\}} \rangle \quad (10)$$

用  $M$  将文件加密, 并将加密后的文件和  $CT$  一起上传到云服务器

#### 4.4 文件解密

公共用户从云端下载加密后的文件和密文  $CT$ , 用  $A_k^c$  表示文件所有者在  $A_k$  中选择的属性的集合, 对于属性  $i \in A_k^c \cap A_k^c (k \in \{1, \dots, N\})$ , 根据  $S_{k,i}$  和  $C_{k,i}$  计算得到:

$$e(S_{k,i}, C_{k,i}) = e(g_1, g_2)^{\mathcal{P}_k(i)} \quad (11)$$

根据式(6)中  $p_k(0)$  所满足的条件, 并通过式(11)进行插值运算, 我们可以得出:

$$P_k = e(g_1, g_2)^{\mathcal{P}_k(0)} = e(g_1, g_2)^{s(v_k - \sum_{j \neq k} R_{kj})} \quad (12)$$

将式(12)中的  $P_k$  全部相乘, 我们可以得到  $Q$  为

$$Q = e(g_1, g_2)^{s(\sum \{v_k\} - R_u)} \quad (13)$$

将式(13)得到的  $Q$  和  $e(g_1^{R_u}, g_2^s)$  相乘, 我们可得到  $Y^s$  为

$$Y^s = e(g_1^{R_u}, g_2^s) \cdot Q \quad (14)$$

通过  $E_0/Y^s$  得到加密密钥  $M$ , 从而解密文件.

#### 4.5 用户撤销

该步骤实现当用户属性发生变化时, 用户权限的撤销. 具体包含以下几个步骤.

##### 4.5.1 重新产生授权中心密钥

选择一个最小属性集, 若不包含该属性集中的

属性,财务信息所有者指定的结构无法满足.例如指定的结构为(公司 = A)  $\vee$  (部门 = 财务部)  $\vee$  ((职位 = 会计)  $\wedge$  (职位 = 财务总监)),那么最小属性集为{A,财务部,会计,财务总监},用  $\gamma$  表示这个最小属性集.

对属性  $i \in \gamma$ ,选择  $t'_{k,i} \in Z_p$ ,定义  $rk_{k,i \leftrightarrow i'} = t'_{k,i}/t_{k,i}$ ,并记录到  $AHL_i$  中, $AHL_i$  为属性  $i$  的版本记录表.

4.5.2 重加密密文

定义  $UpdateFile(i, X, AHL_i)$ . 如果  $i$  是最新的版本,则退出. 否则,寻找  $AHL_i$ ,并且找出  $i$  的最新版  $t_{i(n)}$ . 计算

$$\begin{aligned} rk_{k,i \leftrightarrow i(n)} &= rk_{k,i \leftrightarrow i'} \cdot rk_{k,i' \leftrightarrow i''} \cdots rk_{k,i(n-1) \leftrightarrow i(n)} \\ &= t_{k,i}(n)/t_{k,i} \end{aligned} \tag{15}$$

$$X^{(n)} = (X)^{rk_{k,i \leftrightarrow i(n)}} \tag{16}$$

输出  $X^{(n)}$ .

对于  $k \in \{1, \cdots, N\}, i \in A_k^C$ , 计算

$$C'_{k,i} = UpdataFile(i, C_{k,i}, AHL_i) \tag{17}$$

定义  $CT'$  为

$$CT' = \langle \varphi r + 1, E_0, E_1, \{C'_{k,i}\}_{i \in A_k^C, k \in \{1, \cdots, N\}} \rangle \tag{18}$$

4.5.3 用户私钥更新

用户将  $SK_u$  发送到服务器端.  $UpdateSK(i, X, AHL_i)$  的过程与  $UpdateFile(i, X, AHL_i)$  类似,只是最后一步将  $rk_{k,i \leftrightarrow i(n)}$  替换成  $(rk_{k,i \leftrightarrow i(n)})^{-1}$ . 对于  $k \in \{1, \cdots, N\}, i \in A_k^u$ , 计算

$$S'_{k,i} = UpdataFile(i, S_{k,i}, AHL_i) \tag{19}$$

得到私钥  $Pr$  为

$$pr = \langle \varphi r + 1, D_u, \{S'_{k,i}\}_{i \in A_k^u, k \in \{1, \cdots, N\}} \rangle \tag{20}$$

5 安全性和效率分析

本节将从2个方面对方案进行分析:(1)分析本文方案的安全性;(2)将本文方案的空间开销与其他方案进行对比.

表1所示为效率分析中用到的一些符号以及相关参数.

5.1 安全性分析

(1) 在用户撤销之前,当且仅当用户满足文件所有者提出的结构时,用户方可解密文件. 当用户满足信息所有者提出的结构时,用户的属性集合和信息所有者在每个授权中心选择的属性集合的交集必然不为空,那么我们可以根据式(11)、(12) 计算出所有授权中心的  $P_k$ . 将所有授权中心的  $P_k$  连乘得到  $Q$ .  $Q$  满足式(21):

表 1 效率分析中符号以及相关参数

| 符号    | 说明                   |
|-------|----------------------|
| $S_k$ | 文件加密密钥的字节长度          |
| $S_1$ | $G_1/G_2$ 中一个元素的字节长度 |
| $S_T$ | $G_T$ 中一个元素的字节长度     |
| $S_Z$ | $Z_p$ 中一个元素的字节长度     |
| $S_P$ | CT 中访问策略的字节长度        |
| $N$   | 授权中心的个数              |
| $N_O$ | 系统中文件所有者的个数          |
| $N_u$ | 系统中用户的个数             |
| $N_R$ | 一个文件中撤销用户的个数         |
| $N_G$ | 一个属性组中用户的个数          |
| $m$   | 授权中心中属性种类的个数         |
| $t_c$ | CT 中出现的属性总数          |
| $t_u$ | $SK_u$ 中出现的属性总数      |
| $l$   | 所有者文件的层数             |
| $U$   | 授权中心所有属性的集合          |

$$\begin{aligned} Q &= e(g_1, g_2)^{s(\sum \{v_k\} - R_u)} \\ &= e(g_1, g_2)^{s(\sum \{v_k\}) - s \cdot R_u} \\ &= e(g_1, g_2)^{s(\sum \{v_k\})} / e(g_1, g_2)^{s \cdot R_u} \\ &= e(g_1^{\sum \{v_k\}}, g_2)^s / e(g_1^{R_u}, g_2^s) \\ &= Y^s / e(g_1^{R_u}, g_2^s) \end{aligned} \tag{21}$$

我们可以看出只要将  $Q$  乘以  $e(g_1^{R_u}, g_2^s)$  即可得到  $Y^s$ . 在发给用户的私钥中包含  $D_u$ , 用户下载的  $CT$  包含  $E_1$ . 根据式(8) 和  $E_1 = g_2^s$ . 可以得到式(22):

$$\begin{aligned} &e(g_1^{R_u}, g_2^s) \\ &= e(D_u, g_2^s) \\ &= e(D_u, E_1) \end{aligned} \tag{22}$$

将  $Q$  乘以  $e(D_u, E_1)$  即可得到  $Y^s$ .

再通过  $E_0/Y^s$  得到加密密钥  $M$ , 从而解密文件.

当用户不满足文件所有者提出的结构时,用户的属性集合和信息所有者在每个授权中心选择的属性集合的交集必然有一个为空,那么不可以得到所有授权中心的  $P_k$ ,就无法计算出正确的  $Q$ ,无法继续以上的计算过程,得不到正确的解密密钥.

综上所述,在用户撤销之前,当且仅当用户满足文件所有者提出的结构时,用户方可解密文件.

(2) 在用户撤销之后,当且仅当用户满足文件所有者提出的结构时,用户方可解密文件.

进行重加密时,对用户不作要求的授权中心并没有影响,因为无论如何,用户都能通过这些没作要求的授权中心的验证. 更新的只是作要求的授权中心的部分,这样减少了系统的开销. 我们从用户撤销

的重新产生授权中心密钥步骤中可以看出,我们的算法只对用户选择的属性做了参数更新. 根据式(15)和式(16),可以得出式(23):

$$\begin{aligned} &C'_{k,i} \\ &= (C_{k,i})^{r_{k,i}^{t(n)}} \\ &= (C_{k,i})^{t_{k,i}^{(n)}/t_{k,i}} \\ &= (T_{k,i}^s)^{t_{k,i}^{(n)}/t_{k,i}} \\ &= (g_2^{s \cdot t_{k,i}})^{t_{k,i}^{(n)}/t_{k,i}} \\ &= g_2^{s \cdot t(n)_{k,i}} \end{aligned} \tag{23}$$

根据式(10)和 $T_{k,i} = g_2^{t_{k,i}}$ ,我们可得出 $C_{k,i} = g_2^{s \cdot t_{k,i}}$ .  $t(n)_{k,i}$ 是在用户撤销的重新产生授权中心密钥步骤中随机选择的,我们可以看出新的密文的形式与原来的形式相同,相当于用新的参数生成了新的密文. 同理可证密钥更新的过程也是用新的参数产生了新的用户密钥. 又因为解密过程与撤销前的解密过程算法相同,由此可以得出,当用户满足信息所有者提出的结构时,可以得到最新的解密密钥,当用户不满足信息所有者提出的结构时,无法得到最新的解密密钥.

用户撤销之后,原来的密文作废并产生新的密文,用户原有的密钥因为与新的密文中的 $\varpi$ 不一致,已经不能用来解密新的密文. 如果想解密密文,只可以请求新的用户私钥,即进行用户密钥的更新.

表 3 空间开销比较

| 方式                   | 密文长度                              | 用户私钥长度                          | 公钥长度                                         | 重加密的消息长度                                   |
|----------------------|-----------------------------------|---------------------------------|----------------------------------------------|--------------------------------------------|
| NGS <sup>[7]</sup>   | $(t_c + 2N_R)S_1 + S_T$           | $(t_u + 4)S_1$                  | $(\overline{m} + \overline{l} + 6)S_1 + S_T$ | 0                                          |
| RNS <sup>[8]</sup>   | $t_c(2S_1 + S_T) + S_P$           | $t_uS_1$                        | $ U (S_1 + S_T)$                             | $O((t_u + 1) * S_T(N_u - N_R))$            |
| BCH <sup>[9]</sup>   | $l * S_1 + S_k$                   | $l * N_o * S_1$                 | $2S_1 * N_o$                                 | N/A                                        |
| VFPS <sup>[10]</sup> | $S_k$                             | $N_oS_k$                        | $O(N_oN_u)$                                  | $O(N_u)$                                   |
| HN <sup>[11]</sup>   | $(2t_c + 1)S_1 + S_T + S_P$       | $(2t_u + 1)S_1 + 2S_k \log N_u$ | $2(S_1 + S_T)$                               | $(N_u - N_G)S_Z * \log(N_u / (N_u - N_G))$ |
| 本文方案                 | $(t_c + N + 1) * S_1 + S_P + S_k$ | $(t_u + 1) * S_1$               | $( U  + N)S_1$                               | $t_uS_Z$                                   |

从表 3 中可以看出,我们方案的用户私钥长度比 BCH 和 VFPS 的用户私钥长度更小,重加密的消息长度比 RNS、VFPS 和 HN 重加密的消息长度更小,密文长度比 NGS 的密文长度更小,与 HN 和 RNS 的密文长度差不多. 公钥的长度比 VFPS 和 BCH 的公钥长度更小,与 RNS 的公钥长度差不多. 总体来说,我们的方案效率比先前的方案在密钥管理的效率上要高,重加密的消息长度要小.

综上所述,在用户撤销之后,当且仅当用户满足文件所有者提出的结构时,用户方可解密文件.

5.2 效率分析

5.2.1 算法特征比较

将本方案的算法特征和其他方案的算法特征进行比较,结果如表 2 所示.

表 2 算法特征比较

| 方式                   | 是否多授权中心 | 用户域     |
|----------------------|---------|---------|
| NGS <sup>[7]</sup>   | 否       | 公共用户    |
| RNS <sup>[8]</sup>   | 是       | 公共用户    |
| BCH <sup>[9]</sup>   | 是       | 所有      |
| VFPS <sup>[10]</sup> | 是       | 所有      |
| HN <sup>[11]</sup>   | 否       | 公共用户    |
| 我们的方案                | 是       | 公共和私人用户 |

我们的方案和 BCH、VFPS 以及 RNS 都是多授权中心的方式,用户域上比其他方式更加灵活,既有公共用户,也有个人用户.

5.2.2 空间开销比较

我们从密文长度、用户密钥长度、公钥长度和重加密的消息长度这几个方面和其他的一些方案作比较. 结果如表 3 所示.

6 结束语

云存储中共享文件的安全隐患一直是研究的热点,而利用以基于属性加密为代表的加密访问控制技术是解决该问题的一个有效途径. 本文用基于属性加密方法提出了一个新的云环境下的安全接入控制方案,将用户分类为个人用户和公共用户,根据用户类型灵活地分享文件,并实现用户的撤销. 最后对方案的安全性和效率进行分析. 下一步工作将重点研究访问策略的动态修改,使分享方式更加灵活.

## 参考文献:

- [1] Jung T, Li X Y, Wan Z, et al. Privacy preserving cloud data access with multi-authorities[C]// Proc. of IEEE INFOCOM, IEEE, 2013: 2625-2633.
- [2] Yu S, Wang C, Ren K, et al. Achieving secure, scalable, and fine-grained data access control in cloud computing[C]// Proc. of IEEE Conference on Information Communications, IEEE, 2010: 534-542.
- [3] Yu S, Wang C, Ren K, et al. Attribute based data sharing with attribute revocation[C]// Proc. of ACM Symposium on Information, Computer and Communications Security, ASIACCS 2010, Beijing, China, 2010: 261-270.
- [4] Matsuda T, Nishimaki R, Tanaka K. CCA proxy re-encryption without bilinear maps in the standard model [C]// Proc. of International Conference on Practice and Theory in Public Key Cryptography. Springer—Verlag, 2010: 261-278.
- [5] Frikken K, Atallah M, Li J. Attribute-based access control with hidden policies and hidden credentials[J]. IEEE Transactions on Computers, 2012, 55(10): 1259-1270.
- [6] Zhang Y, Chen X, Li J, et al. Anonymous attribute-based encryption supporting efficient decryption test [C]// Proc. of ACM Sigsac Symposium on Information, Computer and Communications Security. ACM, 2013: 511-516.
- [7] Narayan S, Safavi-Naini R. Privacy preserving EHR system using attribute-based infrastructure [C] // Proc. of ACM Cloud Computing Security Workshop, Csw 2010, Chicago, 2010: 47-52.
- [8] Ruj S, Nayak A, Stojmenovic I. DACC: Distributed access control in clouds[C]// Proc. of IEEE International Conference on Trust, Security and Privacy in Computing and Communications, IEEE, 2011: 91-98.
- [9] Benaloh J, Chase M, Horvitz E, et al. Patient controlled encryption: ensuring privacy of electronic medical records[C]// Proc. of ACM Cloud Computing Security Workshop, Csw 2009, Chicago, IL, USA, November. DBLP, 2009: 103-114.
- [10] Vimercati S D C D, Foresti S, Jajodia S, et al. Over-encryption: management of access control evolution on outsourced data[C]// Proc. of International Conference on Very Large Data Bases. VLDB Endowment, 2007: 123-134.
- [11] Hur J, Noh D K. Attribute-based access control with efficient revocation in data outsourcing systems[J]. IEEE Transactions on Parallel & Distributed Systems, 2011, 22(7): 1214-1221.

## 作者简介:

蔡炎伟 男, (1991-), 硕士研究生. 研究方向为云计算安全. E-mail: 1228338405@qq.com.

白光伟 男, (1961-), 博士, 教授, 博士生导师. 研究方向为无线传感器网络、移动互联网、网络体系结构和协议、网络系统性能分析和评价、多媒体网络服务质量等.

沈航 男, (1984-), 博士, 讲师. 研究方向为云计算、移动互联网、无线多媒体通信协议等.

朱德剑 男, (1991-), 硕士研究生. 研究方向为移动云计算、云计算节能技术.

(上接第 90 页)

- [8] Roy T Fielding, Richard N Taylor. Principled design of the modern Web architecture[D]. Irvine: Information and Computer Science University of California, 2000.

## 作者简介:

邓玉龙 男, (1990-), 硕士研究生. 研究方向为物联网技术、大数据分析. E-mail: 379879419@qq.com.

张岩 男, (1962-), 研究员. 研究方向为自动控制.

马玉鹏 男, (1979-), 副研究员. 研究方向为物联网应用技术、大数据分析.