基于覆写验证的云数据确定性删除方案

杜瑞忠[1,2]，石朋亮[1,2]，何欣枫[1,2]
（1. 河北大学网络空间安全与计算机学院，河北 保定 071002；2. 河北省高可信信息系统重点实验室，河北 保定 071002）

摘 要：云存储中的数据在生命周期结束删除时，大多是采用删除密钥的逻辑删除方式进行处理，数据仍存在泄露风险，为此提出了一种基于密文重加密与覆写验证结合的云数据确定性删除方案（WV-CP-ABE）。当数据拥有者想删除外包数据时，通过重新加密密文改变密文对应的访问控制策略来实现数据细粒度删除操作；其次构建基于脏数据块覆写的可搜索路径散列二叉树（DSMHT），对要删除的数据进行覆写后正确性验证；最终采用更改密文访问控制策略和数据覆写双重机制保障数据确定性删除。实验分析证明，所提方案在数据确定性删除方面比以前的逻辑删除方法细粒度控制更好，安全性更可靠。

关键词：云存储；密文属性加密；确定性删除；散列二叉树；覆写验证
中图分类号：TP393.08
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2019012

Cloud data assured deletion scheme based on overwrite verification

DU Ruizhong[1,2], SHI Pengliang[1,2], HE Xinfeng[1,2]
1. Cyberspace Security and Computer College, Hebei University, Baoding 071002, China
2. Key Lab on High Trusted Information System in Hebei Province, Baoding 071002, China

Abstract: At the end of data life cycle, there is still a risk of data leakage, because mostly data which was stored in cloud is removed by logical deletion of the key. Therefore, a cloud data assured deletion scheme (WV-CP-ABE) based on ciphertext re-encrypt and overwrite verification was proposed. When data owner wants to delete the outsourced data, the data fine-grained deletion operation was realized by re-encrypting the ciphertext to change the access control policy. Secondly, a searchable path hash binary tree (DSMHT) based on dirty data block overwrite was built to verify the correctness of the data to be deletion. Finally, the dual mechanism of changing the ciphertext access control policy and data overwriting guarantees the data assured deletion. The experimental analysis proves that the fine-grained control is better and the security is more reliable than the previous logical delete method in the assured deletion of data.

Key words: cloud storage, CP-ABE, assured deletion, Hash binary tree, overwrite and verify

1 引言

作为新的服务模式，云计算[1]能够将数据存储和数据共享进行高效的结合，以便为用户提供更优质的服务。随着云计算技术的发展，越来越多的个人或者企业把数据上传到云端，一方面能节省本地硬件存储开销，另一方面可以随时随地享受云计算提供的服务。但 2018 年泰累兹数据威胁报告[2]指出，2017 年 36%的受访者表示遭遇过数据泄露，并预测这一比例还将上升，数据泄露已成为影响云计算发展和应用的重要问题之一。其中数据的不安全性删除[3]是导致数据泄露的一个重要原因。

收稿日期：2018−05−31；**修回日期**：2018−10−31
基金项目：国家自然科学基金资助项目（No.61572170）；河北省自然科学基金资助项目（No.F2018201153, No.2016205023）；河北省高等学校科学技术研究基金资助项目（No.ZD2016043）；河北省物联网监控工程技术研究中心基金资助项目（No.3142016020）
Foundation Items: The National Natural Science Foundation of China (No.61572170), The Natural Science Foundation of Hebei Province (No.F2018201153, No.216205023), The Science and Technology Research Project in Colleges and Universities of Hebei Province (No.ZD2016043), Hebei IoT Monitoring Engineering Technology Research Center(No.3142016020)

目前，用户大多是将数据加密后存储到云端，在数据删除命令发出后，通过删除加密密钥来保证数据的不可解密和恢复，以达到数据删除的目的。这样的删除机制存在很大的弊端，因为其只对数据进行逻辑删除，待删除的数据仍然存储在云端，一旦有非法分子获得云端的数据，就可能对所得数据进行暴力破解，从而导致敏感信息泄露。如果存在部分云服务提供商为了自身的利益，只对数据进行逻辑删除，那么用户在多租户模式下就面临着数据泄露危机。另外，云存储的模式和以往的存储模式有很大的区别。在云存储模式下，一方面数据拥有者将数据上传到云端存储，致使数据控制权被移交至云端。另一方面，现存的数据逻辑删除方式，一旦加密密钥被恢复，数据泄露几率增大。针对以上情况，在数据生命周期结束需要删除时，如何保证数据的确定性删除，使数据在云端永久性删除或者密钥删除后无法再次解密及恢复是现阶段云存储研究的一个重点和难点问题。

2　相关工作

目前，研究者对云端数据确定性删除已经做了很多有益的尝试。熊金波等[4]从密码学的角度对已有的确定性删除方法进行综合性分析对比，给出了目前存在的三大类确定性删除机制的优缺点。文献[5]利用（S,T）门限密钥共享方法将加密密钥分成 n 份，发送到网络节点上的分布式散列表（DHT，distributed hash table）进行存储，但是 DHT 社交网络容易遭受非法分子的跳跃攻击。为解决此问题，文献[6]将加密密钥的长度进行扩展后再使用密钥共享方案发送到 DHT 网络上，虽然能抵御跳跃攻击但是加重了密钥存储开销。为减少密钥存储开销，李超零等[7]采用两级加密方式，其中控制密钥由密钥生成树生成，基于树结构减少密钥的存储，之后通过秘密共享方法发到 DHT 网络中。文献[8]提出一种基于混沌序列比特流变化的云多媒体文件确定性删除方案，数据加密上传前进行数据比特流变换，数据删除时仅仅删除对应数据的比特流，达到安全删除的目的，减少了对加密密钥和可信第三方的依赖，但是只适用于视频和图片类的多媒体文件。薛亮等[9]提出一种基于属性加密和属性撤销的数据删除方案，通过撤销属性实现数据的访问控制，从而达到数据删除的目的。但是以上方案仅仅是对数据加密密钥进行安全性处理，原数据依然在云端，存在非法用户获得数据后暴力破解的可能，为此张坤等[10]将加密密文进行分片抽样，把抽样后的剩余密文上传到云端，抽样密文交由可信第三方保管，使存在云端的数据不完整，来抵制暴力攻击。但是如果使用密文抽样技术使云端存储的密文不完整，会给云端的数据更新和密文检索带来不便。而针对多样性用户，如何实现文件数据的细粒度控制操作也成为数据确定性删除要考虑的一个关键问题。文献[11]提出一种基于策略的删除机制，使得密文对应一条或者几条策略，用策略加密密文，用户只有满足访问策略才能访问密文，删除时撤销策略。基于这种思想，熊金波等[12]提出一种基于身份加密的安全自销毁方案，根据用户身份不同提供数据的细粒度访问，可是容易暴露用户身份信息。文献[13]提出基于属性加密的数据文件删除机制，用属性加密文件，用户只有满足访问控制属性才能访问文件，减少了用户身份信息的暴露。禹勇等[14]提出一种在互联网和雾计算框架下，基于细粒度访问控制的确定性删除方案，通过改变数据的访问控制，来达到数据删除的目的，但不适应云端大数据存储。以上文献都未对云端数据删除后进行验证。综合分析现有文献，发现存在以下挑战。

1) 云存储环境下无法对数据文件进行有效的细粒度操作，致使数据泄露。

2) 没有实现即时删除，基于 DHT 社交网络的删除机制，只能依赖网络的更新周期，不能适用云存储环境下的即时删除要求。

3) 删除云存储中的数据时，大多数是采用删除密钥这样的逻辑删除方式，致使密钥删除后文件依旧存在泄露的可能。

4) 没有对云数据删除操作进行验证。

针对以上情况，提出一种基于密文重加密和数据覆写验证结合的云数据确定性删除方案(WV-CP-ABE，overwrite and verify ciphertext-policy attributed based encryption)，可以有效实现数据访问以及删除细粒度控制和数据删除验证。本方案的主要工作如下。

1) 采用基于密文策略属性基加密机制(CP-ABE，ciphertext-policy attribute-based encryption）加密数据，当数据拥有者想删除外包数据时，通过重新加密密文改变密文对应的属性访问控制

策略来实现数据细粒度操作和确定性删除。

2) 设计了一种基于脏数据覆写的可搜索路径散列二叉树（DSMHT，dirty data and search merkle hash tree），对云存储的数据覆写后进行验证。根据辅助的可信删除证据，判断是否对数据文件真正进行了覆写操作。

3) 对提出的云数据确定性删除方案进行了详细的敌手模拟安全性证明，表明本方案可以满足要求的数据细粒度操作和确定性删除目标。

3 预备知识

3.1 属性基加密

属性基加密[15] (ABE，attribute-based encryption)是由 Sahai 和 Watens 在 2005 年的欧密会上提出的模糊身份加密。目前常用的 2 种方案是：基于密钥策略属性基加密（KP-ABE，key-policy attribute-based encryption）和基于密文策略属性基加密(CP-ABE，ciphertext-policy attribute-based encryption)。这 2 种加密机制都用到属性访问控制策略(AC，access control)。

假设初始化的系统属性个数为 n，则得系统属性集合为 $\Omega=\{\text{att}_1,\text{att}_2,\cdots,\text{att}_n\}$，$A_i=\{v_{i,1},v_{i,2},\cdots,v_{i,n_i}\}$，其中 $n_i=|A_i|$，A_i 表示第 i 个属性 att_i 的取值。用户属性集合 $\text{Au}=\{\text{Au}_1,\text{Au}_2,\cdots,\text{Au}_m\}$，其中 $m\in[1,n]$。Au_i 表示用户属性集合 Au 中属性的取值。$\text{AC}=[\text{AC}_1,\text{AC}_2,\cdots,\text{AC}_k]$，$k\in[1,n]$，为定义的一个密文的访问控制策略。访问控制策略是借助树结构，采用与门、或门和门限控制方法对属性进行管理。数据拥有者首先将访问控制策略 AC 转换为一棵访问控制树，其中非叶子节点表示属性控制判断条件，叶子节点表示属性值。图 1 所示的是访问控制策略 AC= [hebie, cs, man, pro, is]转化为树形式的一种表达。

3.2 双线性映射

设 p 是素数，G_T 是阶为 p 的乘法循环群，Gv 是阶为 p 的乘法循环群，通常称映射 $e:G_T\times G_T\to G_V$ 为一个双线性对，e 满足以下的 3 个性质。

1) 双线性：对于任意 $\delta,\xi\in Z_p$ 和 $\chi,\gamma\in G_T$，都有 $e(\chi^{\delta},\gamma^{\xi})=\text{e}(\chi,\gamma)^{\delta\xi}$。

2) 非退化性：存在 $\chi,\gamma\in G_T$，使 $e(\chi,\gamma)\neq 1_{G_v}$。

3) 可计算性：对任意的 $\chi\in G_T,\gamma\in G_v$，存在有效的算法计算 $e(\chi,\gamma)$ 的值。

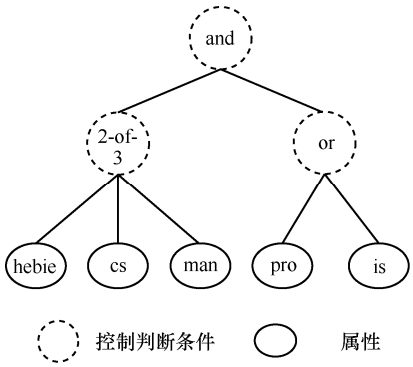


图 1 访问控制策略 AC

4 方案体设计

4.1 符号意义

符号及其含义如表 1 所示。

表 1 符号及其含义

符号	含义
PK	系统的公钥
MSK	系统的主密钥
SK_u	授权用户私钥
M	原始密文
C	加密密文
NC	重新加密后的密文
DC	脏数据覆写后的密文
Ω	系统属性集合
A_u	授权用户所拥有的属性集合
AC	基于策略属性的访问控制策略
a_j	系统属性集合里的属性元素
G_T	阶为 P 的乘法循环群
G_V	阶为 p 的乘法循环群
g	群 G_0 的生成元
n	系统属性集合中属性的个数
Z_p	阶为 P 的整数域
D_o	用户私钥的公共数值
D_j	用户私钥的属性值
$H(\cdot)$	单向散列函数

4.2 系统模型

本文提出的云数据确定性删除方案（WV-CP-ABE）共包括四部分，分别是数据拥有者（DO，data owner）、可信授权机构（TA，trusted authority）、云服务提供商（CSP，cloud server provider）和用户（user）。整体结构如图 2 所示，具体各部分的功能如下。

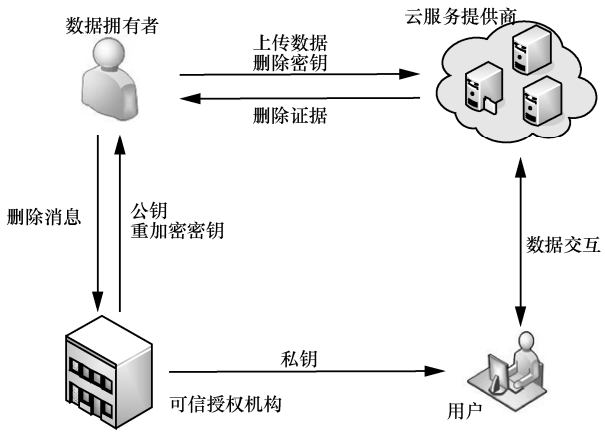


图 2　系统整体结构

数据拥有者(DO)：创建数据文件，上传云端前对数据文件进行加密处理。数据拥有者虽然上传了数据到云端，但是怀疑云服务提供商是否按照约定对数据进行处理，担心数据有泄露的危险。

可信授权机构(TA)：产生密钥中心，负责给用户分发私钥，根据用户属性分发不同的用户私钥，而只有满足数据文件访问控制策略的用户才能下载文件解密出明文。

云服务提供商(CSP)：自身拥有强大的计算能力和存储资源，对租户提供长时间的存储服务。但是本身是诚实且好奇的，在商业利益和自己名声的驱使下，会有泄露租户信息的不法行为。

用户(user)：数据文件的使用者，通过自身拥有的属性在可信授权机构获取私钥，然后在云端下载数据，如果满足数据的访问控制策略，就能成功解密文件。

4.3　系统结构流程

云数据确定性删除方案(WV-CP-ABE)总共包括以下几个步骤：系统初始化 setup、用户私钥产生 KeyGen、数据加密 encrypt、数据解密 dncrypt、删除信息生成 DelRequest、删除密钥生成 ReKeyGen、访问控制策略重加密 ReEncrypt 和数据覆写验证 Verify，流程如图 3 所示，具体如下。

步骤 1　系统初始化 setup(1^k)：TA 执行初始化算法，依据相应的参数 K，产生一对密钥，即公共密钥 PK 和主要密钥 MSK。

步骤 2　用户私钥产生 KeyGen(PK, MSK, A_u)：TA 对 PK、MSK 以及用户属性集合 A_u 进行计算，生成用户私钥 SK_u；并给数据拥有者生成私钥 ssk，该私钥用于密文签名。

步骤 3　数据加密 encrypt(PK, AC, M)：数据拥有者将明文 M、公钥 PK 和访问控制策略 AC 作为参

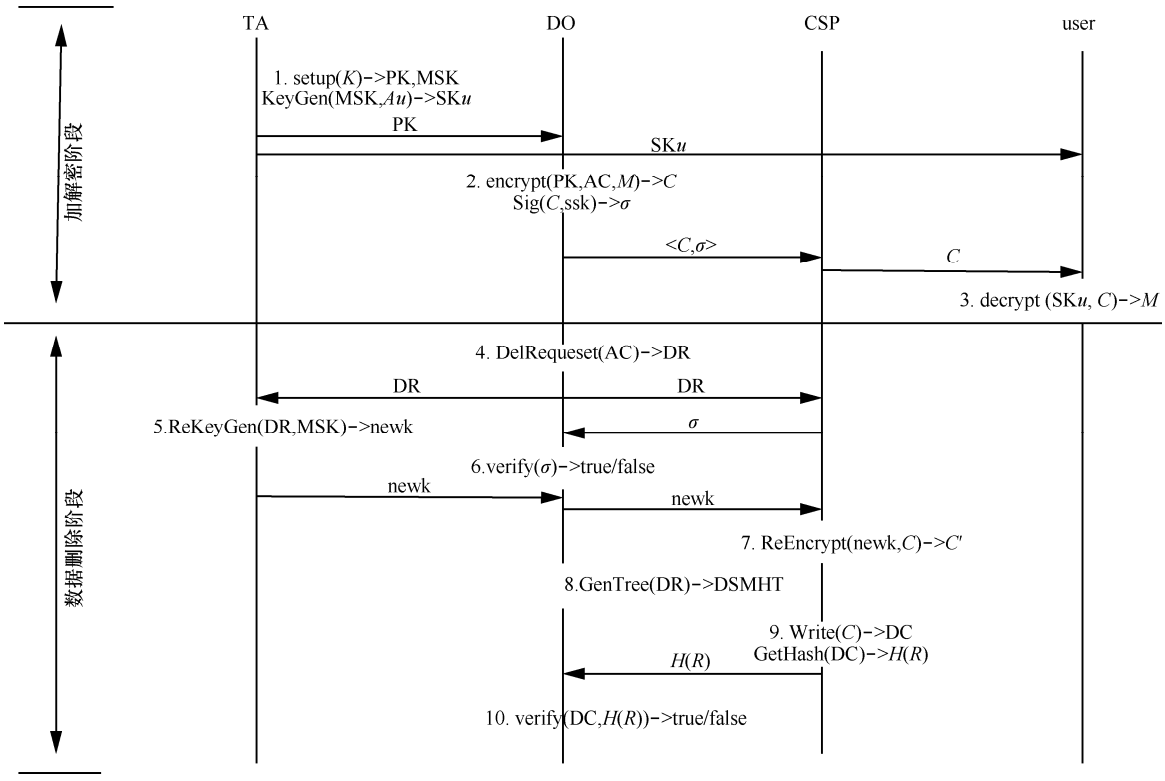


图 3　方案整体流程

数，生成密文 C；并对密文访问控制策略进行签名。

步骤 4 数据解密 decrypt(SK_u,C)：用户首先需要通过数据文件的访问控制策略 AC，如果通过，则利用获得的私钥 SK_u 将密文 C 解密出明文 M。

步骤 5 删除信息生成 DelRequest(AC)：数据拥有者输入要删除数据的访问控制策略 AC，输出删除信息 DR，分别发送给授权机构和云服务提供商，云服务商返回删除数据的签名，数据拥有者对其返回的签名进行验证。

步骤 6 删除密钥生成 ReKeyGen(DR,MSK)：可信授权机构输入删除信息 DR 和系统主密钥 MSK 生成新加密密钥 newk。

步骤 7 访问控制策略重加密 ReEncrypt (newk，C)：云服务提供商输入 newk、密文 C 输出重新加密的密文 NC。

步骤 8 数据覆写验证 verify(NC,$H(R)$)：数据拥有者构建基于脏数据块覆写的可搜索路径散列二叉树对云数据进行覆写操作，并验证覆写结果的正确性。

4.4 安全模型

假设存在敌手 A、挑战者和模拟器 S，为本方案构建以下敌手攻击游戏，具体如下。

1) 敌手 A 尝试构建访问控制策略 AC。

2) 挑战者运行 setup 初始化算法，输出 PK 给敌手。

3) 敌手为得到用户属性集合 A_U，向模拟器 S 请求多个私钥。

4) 敌手 A 给挑战者发送 2 条等长的数据明文 M_0 和 M_1，挑战者随机选择其中一条 $\phi=\{0,1\}$，挑战者选择访问控制策略加密 AC 加密数据，将加密明文 C_ϕ 发送给敌手。

5) 敌手多次尝试 3)。

6) 敌手根据得到的信息对 ϕ 进行猜测得到 ϕ'，如果敌手 A 猜测的 $\phi'=\phi$，则敌手在游戏中获胜；反之，敌手 A 失败。在敌手攻击游戏中，敌手 A 的优势为 $\left|\Pr[\phi'=\phi]-\frac{1}{2}\right|$。

5 方案详细设计

本节给出云数据确定性删除方案（WV-CP-ABE）中数据加密解密阶段和数据删除阶段的具体流程设计方案。数据加密解密阶段包括系统初始化、用户私钥产生、数据加密和数据解密 4 个步骤。数据确定性删除阶段包括删除信息生成、删除密钥产生、访问控制策略重加密和数据覆写验证 4 个步骤。

5.1 加密解密阶段

1) 系统初始化 setup(1^K)

这是在可信授权机构 TA 运行的一个随机算法。首先可信授权机构 TA 选择 2 个阶为 p 的乘法循环群 G_T，G_V，满足 $e:G_\mathrm{T}\times G_\mathrm{T}\to G_\mathrm{V}$，其中 g 为 G_T 的生成元，TA 随机选择 $y\in Z_p$，计算

$$Y=e(g,g)^y \tag{1}$$

然后选择 $t_j\in Z_{p(j\in[1,n])}$，计算

$$T_j=g^{t_j},j\in[1,n] \tag{2}$$

则公钥为 PK=$(e,g,Y,\{T_j\}_{j\in[1,n]})$，主密钥为 $\mathrm{MSK}=(y,\{t_j\}_{j\in[1,n]})$，其中公钥 PK 公开，主密钥 MSK 可信授权机构保存不公开。

2) 用户私钥产生 KeyGen(PK,MSK,A_u)

首先随机选择计算用户私钥的公共基 $r\in Z_p$，计算用户私钥基 D_0

$$D_0=g^{y+r} \tag{3}$$

对于每个属性 a_j，都有 $r_j\in Z_p$，然后基于用户属性计算属性值 D_j

$$D_j=\left\{g^{\frac{r_j}{t_j}}\right\}_{a_j\in A_u} \tag{4}$$

最后产生的用户私钥为 $\mathrm{SK}_u=(D_0,D_j)$。同时可信授权机构 TA 为数据拥有者 DO 产生用于访问控制策略签名的公私钥对（spk，ssk），随机选择一个 $\alpha\in Z_p$，计算 v，如式(5)所示。

$$v=g^\alpha \tag{5}$$

3) 数据加密 encrypt(PK,AC,M)

数据拥有者输入明文 M、公钥 PK 和数据访问控制策略 AC，输出密文 C。数据拥有者首先随机选择 $s\in Z_p$，计算密文 C_1、C_2 和 C_3 如式(6)~式(8)所示。

$$C_1=MY^s=\mathrm{Me}(g,g)^{ys} \tag{6}$$

$$C_2=g^s \tag{7}$$

$$C_3=(g^{t_j})^{s_i}a_j\in\mathrm{AC} \tag{8}$$

最后得到的密文为 $C=(\mathrm{AC},C_1,C_2,C_3)$，然后数据拥有者用签名的私钥对访问控制策略进行签名，计算标签，如式(9)所示。

$$\sigma = (H(f_{\text{name}} \| C_3))^{a} \tag{9}$$

其中，f_{name} 是数据文件的唯一名字标识，最后上传 $\{f_{\text{name}}, C, \sigma\}$ 到云端。

4) 数据解密 encrypt(SK_u, C)

解密过程如下。

$$\frac{C_1 \prod_{a_j \in A_u} e(g^{t_j s_i}, g^{\frac{r}{t_j}})}{e(g^s, D_0)} = \frac{\text{Me}(g,g)^{ys} \prod_{a_j \in A_u} e(g,g)^{rs_i}}{e(g^s, g^{y+r})} =$$

$$\frac{\text{Me}(g,g)^{ys} e(g,g)^{\sum_{a_j \in A_u} rs_i}}{e(g,g)^{ys} e(g,g)^{rs}} = \frac{\text{Me}(g,g)^{ys} e(g,g)^{rs}}{e(g,g)^{ys} e(g,g)^{rs}} = M \tag{10}$$

其中，A_u 为用户的属性集合，对于每一个属性 $a_j \in A_u$，都随机选择一个随机数 $S_i \in Z_p$，且满足

$$\sum_{a_j \in A_u} S_i = S \tag{11}$$

其中，i 为访问控制属性 AC 中属性的序号。

5.2　数据确定性删除阶段

1) 删除信息生成 DelRequest(AC)

当数据拥有者想删除外包的数据时，首先生成数据的删除信息 $\text{DR} = (f_{\text{name}}, \text{AC})$，其中 f_{name} 是要删除数据的唯一名字标识。然后将 DR 分别发送给可信授权机构和云服务提供商。之后云服务提供商返回 $\{f_{\text{name}}, \sigma\}$ 给数据拥有者，数据拥有者再认证

$$e(\sigma, g) = e(H(f_{\text{name}} \| C_3), v) \tag{12}$$

如果成立，则证明 C_3 确实是要删除密文中的属性访问控制策略。

2) 删除密钥生成 ReKeyGen（DR，MSK）

可信授权机构收到数据拥有者发送的删除信息 DR，根据主密钥 MSK，随机选择 $t'_j \in Z_p$，计算 ck

$$\text{ck} = \frac{t'_j}{t_j} \tag{13}$$

然后将 $\text{newk} = (f_{\text{name}}, \text{AC}, \text{ck})$ 返回给数据拥有者，数据拥有者收到 newk 后，立即将 newk 信息发送给云服务提供商。

3) 访问控制策略重加密 ReEncrypt（newk,C）

云服务提供商接收到 newk 信息后，选择密文 C，然后计算

$$C'_3 = C_3^{\text{ck}} \tag{14}$$

然后替换原来密文的 C_3 部分，组成新的密文 $\text{NC} = (C_1, C_2, C'_3, \text{AC}')$。

4) 覆写验证 verfiy(NC, $H(R)$)

数据拥有者首先构造基于脏数据覆写的可搜索路径散列二叉树，根据要删除数据块的多少生成最小二叉树，从数字 1 开始，层次遍历二叉树给节点赋值。然后准备一个和外包数据一样大小的二进制随机脏数据块，从二叉树根节点到每个叶子节点都有一条最短路径，将路径经过的节点序号记录下来再转换为二进制，和脏数据文件数据逐位进行异或运算，得到的新的数据就是此叶子节点对应的要删除的数据块需要覆写的数据，然后叶子节点存储这个脏数据块的散列值，作为验证的根据。按照上述操作遍历完所有的叶子节点。按照新生成的数据对云端存储的数据进行数据覆写，写操作完成后让云服务提供商返回覆写完数据的散列值，和本地存储的进行验证，如果一致，说明覆写删除步骤完成。

数据覆写算法如下。

步骤 1　输入 $\text{Deldata}_{\text{num}}$DirtyData

步骤 2　DSMHT<—GetTree($\text{Deldata}_{\text{num}}$)

步骤 3　Levelsearch(DSMHT)

步骤 4　for i to n

步骤 5　　Road<—search(i)

步骤 6　　Broad<—Binary(Road)

步骤 7　　for j to m:

步骤 8　　DC<—DirtyData^Broad

步骤 9 $H(R)$<—getHash($H(R)_{\text{L}} \| H(R)_{\text{r}}$)

步骤 10　overwrite(DC)

步骤 11　end

下面以一个删除 8 个数据块的例子详细说明数据覆写的过程，生成的二叉树如图 4 所示。

要删除的数据 a1，它到根节点的最短路径如图中的曲线所示，其中经过的节点序号为 8421，将 8421 转换为二进制得到 10000011100101，然后与准备的脏数据进行异或运算得到新的数据文件，即是 a1 数据文件需要覆写的脏数据。然后依照此步骤完成其余 7 个数据文件的覆写操作，最后通过递归算法得出 DSMHT 根节点的散列值。等到云端要删除数据覆写结束后，让云端返回覆写完数据的散列值，再与本地的散列值进行判断，以此判定覆写删除过程是否正确完成。

6　安全性分析

针对 4.3 节定义的安全模型，本节模拟游戏来

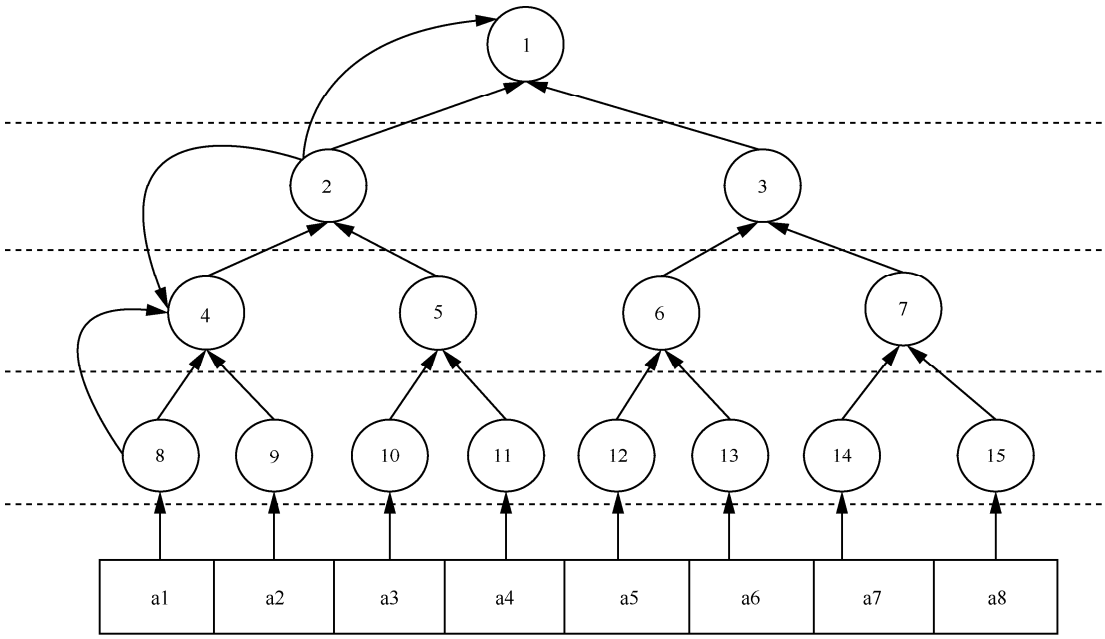


图 4 二叉树

证明本文提出的解决方案是安全的。

判定的双线性 Diffie-Hellman 问题（简称 DBDH）：假设输入（$P, aP, bP, cP, abcP$）和（$P, aP, bP, cP, \mu P$），其中，a、b、c、μP 均为随机的，如果（$P, aP, bP, cP, abcP$）和 aP、bP、cP、μP 在多项式时间内可区分出来，则输出 true。

定理 1 如果敌手在安全模型下可以攻破本方案，则至少存在一个概率多项式时间算法内敌手以不可忽略的优势解决 DBDH（Diffe-Hellmen）问题。

证明假设存在一个敌手 A 以优势 ε 在多项式时间内攻破本方案，下面证明如下的 DBDH 问题游戏可以以优势 $\frac{\varepsilon}{2}$ 完成。

假设 $e: G_0 \times G_0 \to G_V$ 是一个双线性映射，首先 DBDH 问题挑战者设以下情况。

$$\begin{cases} (g, A, B, C, Z) = (g, g^a, g^b, g^c, e(g,g)^{abc}), \phi=0 \\ (g, A, B, C, Z) = (g, g^a, g^b, g^c, e(g,g)^z), \phi=1 \end{cases}$$

其中，随机选取 $a, b, c, z \in Z_p$。挑战者给出模拟器

$$S :< g, A, B, C, Z >=< g, g^a, g^b, g^c, Z >$$

敌手模拟游戏的具体过程如下。

1) 敌手 A 尝试创建访问控制策略 AC。

2) 模拟器 S 初始化公共参数 $Y = e(A,B) = \mathrm{e}(g,g)^{ab}$，发送给敌手 A。

3) 敌手 A 为了满足用户属性集合 A_u，向模拟器 S 请求多个私钥。模拟器 S 接收到信息后，对于每一个属性 $a_j \in A_u$，随机选择 $r_j \in Z_p$ 计算用户私钥 $D_j = \{g^{\frac{r_j}{t_j}}\} a_j \in A_u$，然后返回给敌手 A。

4) 敌手为了能猜测出加密使用的密钥，提交 2 个长度一样的内容不同的明文 M_0、M_1 给模拟器 S，模拟器 S 随机选取 $r = \{0,1\}$，然后用访问控制策略 AC 加密明文 Mr，返回密文 C 给敌手 A。$C = \{AC, C_1, C_2, C_3\}$，其中密文 C 包括 $C_1 = MY^S\phi = \mathrm{Me}(g,g)^{ys}$，$C_2 = g^S Z$，$C_3 = (g^{t_j})^{s_i} a_j \in \mathrm{AC}$。当 $\phi = 0$ 时，由假设 $Z = e(g,g)^{abc}$，其中 α_l ($l \in \{1,2,\cdots,N\}$)可以使 $ab = \sum \alpha_l, c = s$，这样就可以得到 $Z = e(g,g)^{abc}(e(g,g)^{ab})^c = e(g,g)^{\sum \alpha_l s} = Y^s$，可知道密文 C 是关于 Mr 的一个有效密文。当 $\phi = 1$ 时，$r' \neq r$，$C_2 = g^s Z = g^s e(g,g)^z$，因为 z 为随机数，所以密文 C 不包括明文 Mr 的任何有用信息。

5) 敌手 A 重复上述攻击。

6) 敌手 A 根据收到的信息猜测 r' 的值。如果 $r' \neq r$，则模拟器 S 输出 $\phi' = 1$，敌手无法获取任何关于 r 的信息，则有 $\frac{1}{2}\Pr[\phi' = \phi \mid \phi = 0] + \frac{1}{2}\Pr[\phi' = \phi \mid \phi = 1] - \frac{1}{2} = \frac{1}{2} \times \left(\frac{1}{2} + \varepsilon\right) + \frac{1}{2} \times \frac{1}{2} - \frac{1}{2} = \frac{\varepsilon}{2}$，当然也有

$\Pr[\phi'=\phi\,|\,\phi=1]=\frac{1}{2}$。如果$r'=r$，则模拟器 S 输出$\phi'=0$，敌手获取 Mr 的密文，之前定义过敌手的优势为$\varepsilon$，则有$\Pr[r'=r\,|\,\phi=0]=\frac{1}{2}+\varepsilon$，得$\Pr[\phi'=\phi\,|\,\phi=0]=\frac{1}{2}+\varepsilon$。最后得到的整体优势为$\frac{1}{2}\mathrm{P_r}[\phi'=\phi\,|\,\phi=0]+\frac{1}{2}\Pr[\phi'=\phi\,|\,\phi=1]-\frac{1}{2}=\frac{1}{2}\times\left(\frac{1}{2}+\varepsilon\right)+\frac{1}{2}\times\frac{1}{2}-\frac{1}{2}=\frac{\varepsilon}{2}$。

综上可以知道敌手在多项式时间内解决 DBDH 问题的优势是$\frac{\varepsilon}{2}$，根据 DBDH 假设可知，敌手的优势是可以忽略的，所以可以证明本方案是安全的。

为进一步说明本文所提出的 WV-CP-ABE 方案的安全可靠性。本文从加密方式、删除机制、细粒度安全访问和删除验证 4 个方面将 WV-CP-ABE 方案与现有的云数据确定性删除方案 Vanish[16]、ISS[17]、ESITE[18]和 SelfDOC[19]进行对比分析，详细结果如表 2 所示。结果表明本方案一方面在云数据删除时采用重加密访问控制策略和数据覆写双重保证，另一方面在云数据删除后增加验证过程，防止不可信云服务提供商伪造删除信息，具有较高的安全性。

7　实验仿真

本文采用腾讯云服务器和本地电脑搭建实验所需的环境。腾讯云服务器为专业型服务器，CPU 为四核、内存为 8 GB，充当方案中的云服务提供商。本地 3 台电脑件配置为戴尔 OptiPlex 3020 Mini Tower 台式机，处理器为 Inter Core(TM) i5-4590@3.30 GHz 四核，内存为 8 GB，主硬盘为影驰 CX0128ML106-P(128 GB 固态硬盘)，分别充当方案中的数据拥有者、授权机构和用户。部署的 Linux 系统为 Centos6.7，Hadoop 版本为 hadoop-2.6.0，采用 C 语言并基于 PBC（pairing- based cryptography library）函数库进行编程开发。

实验主要是测试所提 WV-CP-ABE 方案在文件加解密、云端数据重加、二叉树生成以及数据覆写验证等过程的时间消耗情况。

图 5 测试不同文件大小方案加密时间的消耗。首先，在访问控制策略固定为 15 个属性的情况下，为更好构建现实的云存储环境，选用的文件大小分别为 1 MB、2 MB、4 MB、8 MB、16 MB、32 MB、64 MB、128 MB 和 256 MB 测试数据文件的加密时间。图 6 是在相同条件下测试用户解密数据的时间消耗。从图 5 和图 6 中可以发现与文献[9]、文献[21]相比，加解密消耗时间随着数据文件的增多逐渐增多，但是数据文件增大到 256 MB 时本方案的加解密时间明显少于对比方案。主要原因是本方案和文献[21]采用 CA-ABE 加密，密文仅和一个访问控制策略有关，而文献[9]采用 KP-ABE 加密，密文和属性相关，随着文件的增多，将属性关联到文件中时间消耗增大，导致对应的加解密时间增多。

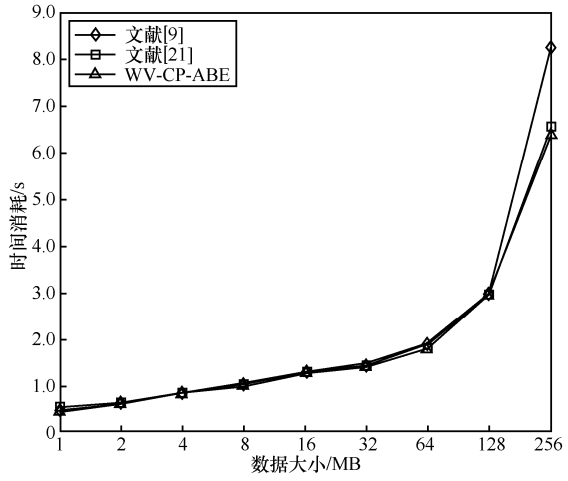


图 5　不同文件大小加密时间消耗

表 2　　**不同方案对比**

方案	加密方式	删除机制	细粒度安全访问	删除验证
Vanish[16]	对称密钥	删除密钥	无	无
ISS[17]	IBE	删除密钥	身份控制访问粒度	无
ESITE[18]	IB-TRE	删除密钥	身份+时间控制访问粒度	无
SelfDOC[19]	ABE	删除密钥+抽样密文	多安全等级+访问控制策略	无
WV-CP-ABE	CP-ABE	重加密访问策略+数据覆写	基于属性访问控制策略	有

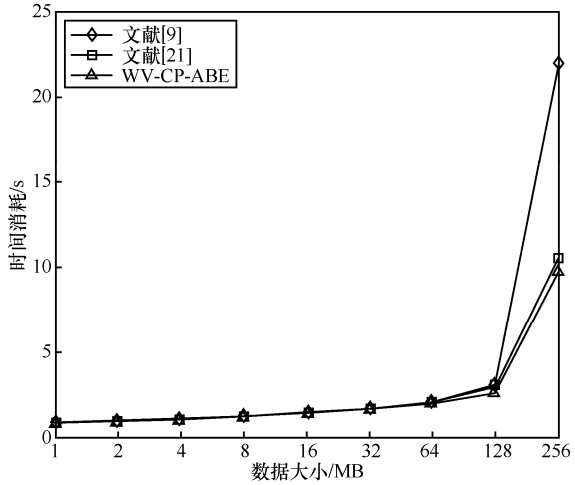


图 6　不同文件大小解密时间消耗

图 7 测试数据大小固定，随着访问控制策略中属性个数变化，数据加解密时间消耗。根据云存储中个人数据使用的调查报告[20]发现，云数据中文档类型占比最大，其次为照片类型。基于此情况本实验选用 1 MB 大小的数据作为测试数据，分析已存在的加密方案同时结合本文的设计目标，属性个数大多数在 5~15 个之间变化，而当属性个数为 15 个时已能满足方案安全要求。为此本实验数据大小固定为 1 MB 情况下，访问控制策略 AC 中属性个数从 5 个增加到 15 个。从图中可以看出随着访问控制策略里属性增多，数据加密和解密的时间大致呈现线性上升关系，而且相同属性个数情况下，数据加密消耗时间小于数据解密时间消耗。

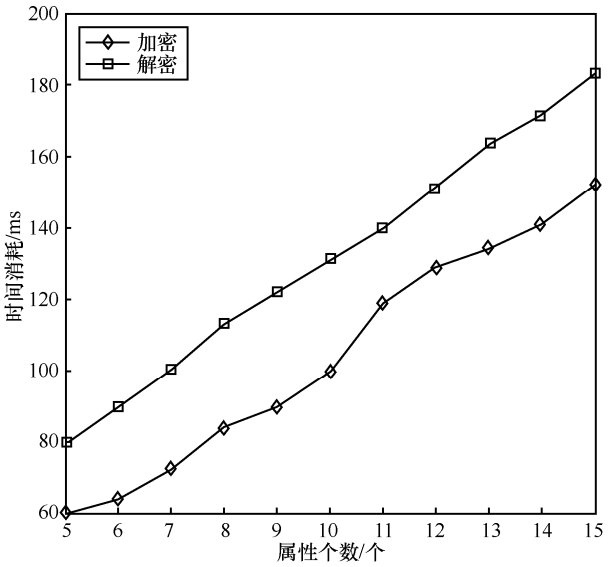


图 7　不同属性个数加解密时间消耗

图 8 测试的是不同属性个数情况下云端对密文中访问属性控制结构重加密的时间消耗。数据删除阶段的重加密密钥在可信授权结构生成时，TA 只需要在 Z_p 寻找一个随机数，所以计算时间消耗很小，而主要的时间消耗在云端访问控制策略重加密。当固定文件大小为 1 MB，访问控制策略中属性个数从 5 个增加到 15 个，从图中可以看出来，随着属性个数增多，时间消耗基本维持在 95 ms 左右。

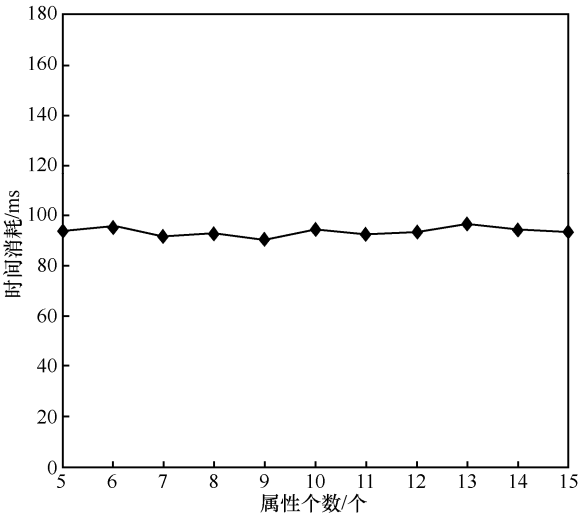


图 8　不同属性个数情况下重加密时间消耗

图 9 测试的是生成不同高度 DSMHT 树的时间消耗。由于数据分块的大小不同，导致数据块个数不同，致使 DSMHT 的高度也不同，为不失一般性，每个准备的脏数据块大小为 4 KB，生成树的高度从 14 增加到 21，当树高度为 21 时，基本可以满足数据覆写验证要求。从图中可以看出随着树高度的增加，时间消耗不再呈现线性关系。当高度为 21 时，时间消耗大约为 5.30 s，在可接受的范围。

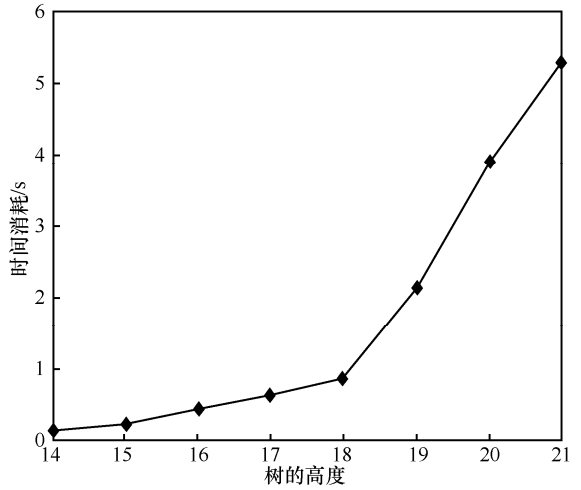


图 9　不同高度 DSMHT 生成时间消耗

图 10 测试的是文件大小从 1 MB 到 64 MB，采用全零覆写、随机覆写和本方案覆写时的时间消耗，从图中可以看出随着文件的增大，覆写时间消耗大致呈比例增大，且本方案的时间消耗虽然比全零覆写方式多，但是却和随机覆写方式的时间消耗基本一致。分析其原因，全零覆写模式直接对文件数据进行覆写，所以相同文件大小，时间消耗最少；随机覆写模式需要产生随机数，本方案覆写模式需要读取生成好的脏数据，因此比全零覆写模式消耗时间多，而这 2 种模式的时间消耗大致相同。

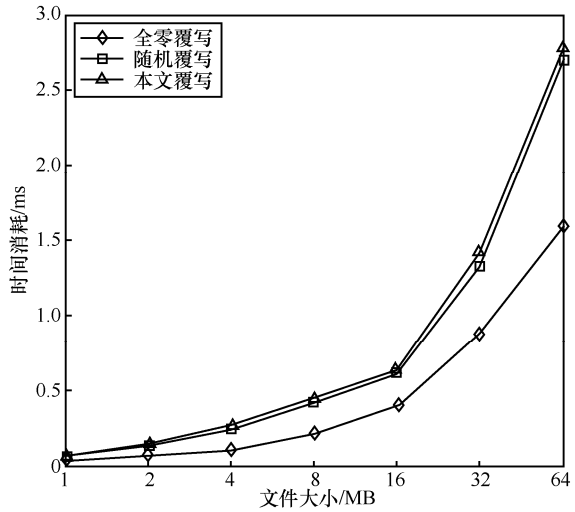


图 10　不同覆写方法时间消耗

表 3 测试的是对文件大小从 1 MB 到 64 MB 进行覆写验证的时间消耗。从表中可以看出随着文件大小的增加，时间消耗逐渐增多。64 MB 的文件覆写的时间大约为 2.70 s，覆写验证时间为 9.45 s，均在允许接受的范围。

表 3　不同大小数据覆写及覆写验证时间消耗

文件大小/MB	覆写时间/s	覆写验证时间/s
1	0.060	0.175
2	0.144	0.42
4	0.260	0.9
8	0.450	1.61
16	0.630	3.34
32	1.421	5.14
64	2.786	9.45

8　结束语

采用删除验证思想，提出一种基于密文重加密与覆写验证技术结合的数据确定性删除方案。当数据删除时，首先采用重加密云端密文的访问控制策略，使数据文件不能解密；其次构建基于脏数据块覆写的可搜索路径散列二叉树对云端密文进行覆写验证处理，保证删除过程准确地完成；最终通过敌手模拟游戏证明方案满足细粒度操作和确定性删除目标。

下一步研究目标是对上传到云端的数据进行安全等级分类，对不同安全等级的数据文件采用不同的数据确定性删除方法，以便达到资源合理的动态分配。

参考文献：

[1] 王国峰,刘川意,潘鹤中,方滨兴.云计算模式内部威胁综述[J].计算机学报,2017,40(02):296-316.
WANG G F, LIU C Y,PAN H Z, FANG B X. Survey on Insider to Cloud Computing[J]. Chinese Journal of Computers,2017, 40(02): 296-316.

[2] KAUSPADIENE L, RAMANAUSKAITE S, CENYS A, et al. Modeling of enterprise management structure for data Leakage evaluation[J]. Information Security Journal: A Global Perspective, 2018, 27(1): 1-13.

[3] RAMOKAPANE K M, RASHID A, SUCH J M. Assured deletion in the cloud: requirements, challenges and future directions[C]//ACM On Cloud Computing Security Workshop. 201: 97-108.

[4] 熊金波,李凤华,王彦超,马建峰,姚志强.基于密码学的云数据确定性删除研究进展[J].通信学报,2016,37(08):167-184.
XIONG J B, LI F H, WANG Y C. Research progress on cloud data assured deletion based on cryptography[J]. Journal on Communication, 2016, 37(8): 168-184.

[5] GEAMBASU R,KOHNO T,LEVY A, et al. Vanish: increasing data privacy with self-destructing data [C]//ACM Conference on USENIX Security Symposium. 2009: 299-316.

[6] ZENG L, SHI Z, XU S, et al. SafeVanish: an improved data self-destruction for protecting data privacy [C]//IEEE International Conference on Cloud Computing Technology and Science. 2010: 521-528.

[7] LI C L, CHEN Y, ZHOU Y D. A data assued deletion scheme in cloud storage[J].China Communication,2014,11(04):98-110.

[8] YAO W B, CHEN Y J, WANG D B. Cloud multimedia files assured deletion based on bit stream transformation with chaos sequence[J]. Algorithms and Architectures for Parallel Processing. ICA3PP 2017: 441-451.

[9] XUE L, YU Y, LI Y, et al. Efficient attribute-based encryption with attribute revocation for assured data deletion [J]. Information Sciences, 2018: 1-11.

[10] 张坤, 杨超, 马建峰, 等. 基于密文采样分片的云端数据确定性删除方法[J].通信学报,2015,36(11):108-117.
ZHANG K,YANG C, MA J F, et al. Novel cloud data assured deletion approach based on ciphertext sample slice［J］. Journal on Commu-

nications,2015,36(11):108－117.

[11] MO Z, XIAO Q J, ZHOU Y, On deletion of outsourced data in cloud computing[C]//International Conference on Cloud Computing, IEEE, 2014: 344-351.

[12] XIONG J, YAO Z, MA J, et al. A secure document self-destruction scheme with identity based encryption[C]//The International Conference on Intelligent Networking and Collaborative Systems. 2013: 239-243.

[13] MO Z, QIAO Y, CHEN S. Two-party fine-grained assured deletion of outsourced data in cloud systems[C]//International Conference on Computing System.. 2014: 308-317.

[14] YU Y, XUE L, LI Y, et al. Assured data deletion with fine-grained access control for fog-based industrial applications[J]. IEEE Transactions on Industrial Informatics, 2018, PP(99):1-1.

[15] 曹来成,刘宇飞,董晓晔,郭显.基于属性加密的用户隐私保护云存储方案[J].清华大学学报(自然科学版),2018,58(02):150-156.
CAO L C,LIU Y F,DONG X Y,GUO X. User privacy-preserving cloud storage scheme on CP-ABE[J]. Journal of Tsinghua University (Science and Technology),2018,58(02):150-156.

[16] GEAMBASU R, KOHNO T, LEVY A, et al. Vanish: increasing data privacy with self-destructing data[C]//The USENIX Security Symposium. 2009: 299-315.

[17] XIONG J B, YAO Z Q, MA J F, et al. A secure self-destruction scheme with IBE for the internet content privacy [J]. Chinese Journal of Computers, 2014, 37(1): 139-150.

[18] YAO Z Q, XIONG J B, MA J F, et al. A secure electronic document self-destructing scheme in cloud computing [J]. Journal of Computer Research and Development, 2014, 51(7):1417-1423.

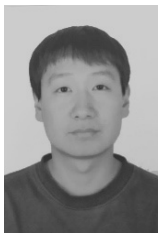
[19] XIONG J B, YAO Z Q, MA J F, et al. A secure self-destruction scheme for composite documents with attribute based encryption [J]. ACTA Electronica Sinica, 2013, 42(2): 366-376.

[20] ISOIT I,DAVID S,GUSTSVO A. Active pages 20 years Later: active storage for the cloud[J]. IEEE Internet Computing, 2018, 22(4): 6-14.

[21] JUNG T, LI X Y, WAN Z, et al. Privacy preserving cloud data access with multi-authorities[C]// INFOCOM. IEEE, 2013:2625-2633.

[作者简介]

杜瑞忠（1975− ），男，河北献县人，博士，河北大学教授，主要研究方向为可信计算与信息安全等。



石朋亮（1992− ），男，河北唐县人，河北大学硕士生，主要研究方向为可信计算与信息安全等。

何欣枫（1976− ），男，天津人，河北大学副教授，主要研究方向为云计算安全与可信计算等。