第36卷第10期 东北大学学报(自然科学版) Vol. 36, No. 10
2015年10月 Journal of Northeastern University(Natural Science) Oct. 2015

doi: 10.3969/j.issn.1005-3026.2015.10.011

基于CP-ABE的云存储数据访问控制方案

高 健, 曾 康, 金恒展, 周福才
(东北大学 软件学院, 辽宁 沈阳 110819)

摘 要: 结合云存储的应用环境,构造了一种基于密文策略属性的加密(ciphertext policy attribute based encryption,CP-ABE)技术和收敛加密技术的混合加密数据访问控制方案. 该方案包括密钥发布中心、用户和云服务器三方实体,能高效、灵活、细粒度地进行数据的访问控制,可提高云存储服务器的空间利用率,并使用签名技术支持数据源认证和数据完整性认证. 理论分析与实验验证了该方案具有较高的实际应用价值.

关 键 词: 云存储;基于密文策略的属性加密;访问控制;收敛加密;数字签名
中图分类号: TP 309 **文献标志码**: A **文章编号**: 1005-3026(2015)10-1416-06

Data Access Control Scheme Based on CP-ABE in Cloud Storage

GAO Jian, ZENG Kang, JIN Heng-zhan, ZHOU Fu-cai
(School of Software, Northeastern University, Shenyang 110819, China. Corresponding author: ZHOU Fu-cai, E-mail: fczhou@mail.neu.edu.cn)

Abstract: A hybrid encryption and data access control scheme based on CP-ABE with convergent encryption technology was constructed by combining the application environment of cloud storage. Three entities are included in the proposed scheme which are key distribution center, customers and cloud server. It was indicted that the proposed scheme is efficient, flexible, fine-grained which can also improve the space utilization of the storage server. The data origin authentication and the data integrity certification were supported by using signature authentication technology. Theoretical analysis and experimental test showed that the proposed scheme has highly practical value.

Key words: cloud storage; ciphertext policy attribute based encryption (CP-ABE); access control; convergent encryption; digital signature

随着云计算的发展,云存储被广泛应用. 云存储将网络中大量不同类型的存储设备通过应用软件集合起来协同工作,对外提供数据存储和业务访问功能,以高可靠、低成本和免维护对存储服务实现了革命性变革. 安全性保障是云存储应用的一个核心问题,也是云计算用户一直担忧的主要问题. 文献[1]给出了云存储作为服务使用必须要解决的安全问题,其中最为突出的是访问控制问题.

对于安全性要求不高的数据,数据拥有者(data owner, DO)可以依赖于云服务提供商(cloud service provide,CSP)提供的基本访问控制功能,来实施对数据的保护. 但对于敏感性数据,DO必须通过加密数据并控制用户的解密能力以实现访问控制,这一方法称为密文访问控制. 在这种CSP不完全可信的情况下,引入密文机制的访问控制是必要的,是目前被广泛认可采用的云存储服务访问控制机制. 但是利用传统的对称密钥机制和非对称密钥机制,缺乏对细粒度访问控制和授权灵活性. 国内外学者针对云存储中访问控制存在的安全问题,进行了深入研究,并取得了一定的成果. Sahai等[2]提出了模糊的基于身份的加密方案,这是最早的基于属性加密(attribute based encryption,ABE)的雏形. Goyal等[3]提出了密钥

收稿日期: 2014-12-04
基金项目: 国家科技重大专项基金资助项目(2013ZX03002006); 辽宁省科技攻关项目(2013217004); 中央高校基本科研业务费专项资金资助项目(N130317002)
作者简介: 高 健(1977-),男,辽宁沈阳人,东北大学博士研究生; 周福才(1964-),男,辽宁沈阳人,东北大学教授,博士生导师.

策略属性的加密方案(key－policy attribute based encryption,KP－ABE). Bethencourt 等[4]提出了更接近于现实访问控制系统的密文策略属性的加密方案(ciphertext policy attribute based encryption,CP－ABE).

CP－ABE 算法最突出的优点是适用于分布式环境下解密方不固定的情况,加密方加密信息时不需知道具体是谁解密,而解密方只需要符合相应条件便可解密. 由于 CP－ABE 算法的诸多优点,有很多学者对 CP－ABE 算法如何应用到密文访问控制中进行了深入的研究[5-6],其研究内容主要集中在属性撤销、属性授权管理、访问结构等方面,但没有涉及存储空间利用率低的问题. 本文结合云存储的应用环境,构造了一种基于 CP－ABE 加密和收敛加密技术的混合加密数据访问控制方案. 方案中所使用的收敛加密技术[7],即用数据的属性加密数据,相同数据的加密结果也是相同的,保证数据机密性,同时云存储服务器具有检测密文冗余功能,提高访问控制系统的存储空间利用率. 并使用数字签名技术以支持数据源认证和数据完整性认证,具有高效率、细粒度及高灵活性等性能,以及广阔的应用前景和发展空间.

1　基础知识

在 CP－ABE 方案中,访问控制策略由加密方制定,采用秘密共享[8]方式,密文与访问控制策略相关,解密密钥与属性集合相关. 在给出 CP－ABE 各算法之前,首先介绍算法中用到的访问树.

访问树 T 是一个表示访问结构的树. 树的每个叶子节点由属性描述,每个非叶子节点代表一个阈门,由它的子节点和阈值描述. num_x 是节点 x 的子节点数量,k_x 是其阈值,则有 $0<k_x\leqslant \mathrm{num}_x$. 当 $k_x=1$ 时,阈门是“或”门;当 $k_x=\mathrm{num}_x$ 时,阈门是“与”门,每个叶子节点 x 的阈值 $k_x=1$. 令 T 是一个根节点为 R 的访问树,T_x 表示树根为 x 的树 T 的子树,则 T 也可以看作 T_r. 如果属性集合 γ 满足访问树 T_x,则有 $T_x(\gamma)=1$. $T_x(\gamma)$ 可以递归计算得到:如果 x 是非叶子节点,计算 x 所有子节点 x'的 $T_{x'}(\gamma)$值,当且仅当至少 k_x 个子节点返回 1 时,$T_x(\gamma)$返回 1;如果 x 是叶子节点,则只有当 $\mathrm{att}(x)\in\gamma$($\mathrm{att}(x)$表示与叶子节点相关的属性)时,$T_x(\gamma)$返回 1.

CP－ABE 方案具体相关变量和定义如下:

G_0 是一个素数 p 阶生成元为 g 的双线性群;

e 表示双线性映射 $e:G_0\times G_0\rightarrow G_1$;

$\Delta_{i,s}$为拉格朗日系数;

H 为一个哈希函数,且有 $H:\{0,1\}^*\rightarrow G_0$;

$\mathrm{parent}(x)$表示 T 中节点 x 的父节点;

$\mathrm{att}(x)$表示与叶子节点相关的属性;

$\mathrm{index}(x)$表示节点 x 的子节点的索引.

CP－ABE 各个算法分别为初始建立算法 Setup,密钥生成算法 KeyGen,加密算法 Encrypt 和解密算法 Decrypt.

1.1　初始建立

算法 $\mathrm{Setup}(g,G_0,\alpha,\beta)\rightarrow(\mathrm{PK},\mathrm{MK})$,其中 $\alpha,\beta\in\mathbf{Z}_p$ 是两个随机选择的指数. 公钥 PK 由双线性群 G_0,生成元 g,构造系数 $h=g^\beta$ 和双线性配对 $e(g,g)^\alpha$ 组成,即 $\mathrm{PK}=(G_0,g,h,e(g,g)^\alpha)$.

主密钥 MK 由随机指数 β 和生成数据 g^α 组成,即 $\mathrm{MK}=(\beta,g^\alpha)$.

1.2　密钥生成

算法为 $\mathrm{KeyGen}(\mathrm{MK},S)\rightarrow\mathrm{SK}$. 该算法首先选择随机数 $r\in\mathbf{Z}_p$,为每个属性 $j\in S$ 选择随机数 $r_j\in\mathbf{Z}_p$ 后,计算 $D_y=g^r\cdot H(j)^{r_j}$和 $D_j'=g^{r_j}$. 密钥 SK 由 $D=g^{(\alpha+\beta)/\beta}$, $\forall j\in S$: $D_y=g^r\cdot H(j)^{r_j}$ 和 $D_j'=g^{r_j}$组成,即 $\mathrm{SK}=(D,D_y,D_j')$.

1.3　加密算法

算法为 $\mathrm{Encrypt}(\mathrm{PK},M,T)\rightarrow\mathrm{CT}$. 该算法首先为树 T 中的节点 x 选择一个多项式 q_x. 对于每个节点 x,设多项式 q_x 的次数 d_x 比节点的阈值 k_x 少 1,即 $d_x=k_x-1$. 从根节点 R 开始,选择随机数 $s\in\mathbf{Z}_p$ 并设 $q_R(0)=s$. 然后,随机选择多项式 q_R 的其他点 d_R,将其定义完整. 对于其他节点 x,设 $q_x(0)=q_{\mathrm{parent}(x)}(\mathrm{index}(x))$并随机选择其他点 d_x 来完整定义 q_x.

密文 CT 通过给定树形访问结构 T 进行构建. 令 Y 是树 T 中的叶子节点集合,对于每个叶子节点 $y\in Y$, 计算 $C_y=g^{q_y(0)}$ 和 $C_y'=H(\mathrm{att}(y))^{q_y(0)}$. 密文由 $T,C'=\mathrm{Me}(g,g)^{\alpha s},C=h^s$, $\forall y\in Y$: $C_y=g^{q_y(0)}$ 和 $C_y'=H(\mathrm{att}(y))^{q_y(0)}$ 组成,即$\mathrm{CT}=(T,C',C,C_y,C_y')$.

1.4　解密算法

算法为 $\mathrm{Decrypt}(\mathrm{CT},\mathrm{SK})\rightarrow M$. 指定解密过程是一个递归算法,且定义递归算法 $\mathrm{DecryptNode}(\mathrm{CT},\mathrm{SK},x)$,执行过程如下:

1)如果节点 x 是叶子节点,设 $i=\mathrm{att}(x)$,若 $i\in S$,则定义

$$\mathrm{DecryptNode}(\mathrm{CT},\mathrm{SK},x)=\frac{e(D_i,C_x)}{e(D_i',C_x')}=$$

$e(g,g)^{rq_x(0)}$,

否则定义 DecryptNode(CT,SK,x) = ⊥.

2) 如果 x 是非叶子节点,调用 DecryptNode(CT, SK, z)并存储输出为 F_z. 设 S_x 是任意 k_x 大小的子节点 z 的集合,子节点 z 满足 $F_z \neq \perp$. 如果没有这样的集合存在,函数返回 ⊥. 否则,计算

$$F_x = \prod_{z \in S_x} F_z^{\Delta_{i,S'_x}(0)} = e(g,g)^{r \times q_x(0)},$$

其中 $i = \mathrm{index}(z)$.

由递归算法 DecryptNode,可以定义解密算法. 算法从树 T 根节点 R 处简单调用函数 DecryptNode 开始,如果属性集合 S 满足访问树 T,则设

$A = \mathrm{DecryptNode}(\mathrm{CT},\mathrm{SK},r) = e(g,g)^{rs}$,

由此,通过下列方法解密得到明文消息 M:

$C'/(e(C,D)/A) = C'/(e(h^s, g^{(\alpha+r)/\beta})/e(g,g)^{rs}) = M$.

2 云环境中 CP-ABE 访问控制方案

为设计一个基于 CP-ABE 算法的密文访问控制方案,能够灵活、高效、安全地进行访问控制,本方案支持数据认证(包括完整性认证和数据源认证),数据认证采用基于身份的签名技术,减少了 PKI 的公钥证书的验证过程,提高了效率. 同时支持密文数据冗余检测,提高了存储空间利用率.

为了实现对文件数据的认证,包括完整性认证和数据源认证,需要用私钥对文件的摘要进行签名,这样既保证了数据的完整性又可以验证数据源,通过验证摘要值验证完整性,通过验证签名验证数据源的真伪.

为了实现密文数据冗余检测,这里采用收敛加密技术. 收敛加密技术即数据加密的密钥为数据明文属性派生来的,相同明文经过加密后,生成的密文也相同. 收敛加密的优势有两点:一是减少存储空间,因为密钥既可以解密,又可以验证数据完整性;二是提高存储空间利用率,可以检测加密冗余数据,因为相同明文加密的结果也相同,所以可以进行检测. 这里用了一个数据自加密的方式,如图 1 所示. 自加密的思想主要参考收敛加密,文件块的摘要和偏移量连接作为密钥,对文件块本身进行加密.

因此本方案对称密钥加密采用收敛加密,用文件的 Hash 值加密文件,相同文件加密的结构相同,从而云端可以对密文冗余数据进行检测.

整个方案实现了在不完全可信环境(云服务提供商)下的文件安全共享系统. 本方案有一个可信第三方和两个角色. 密钥发布中心 PKG 作为可信第三方,负责整个系统的初始化,管理用户属性和系统全体属性,根据用户提供的属性集发放对应私钥. 角色一是用户,包括了加密上传用户和解密下载用户. 角色二是 Hadoop 云端数据中心,作为云存储服务器提供商,云端并不完全可信.

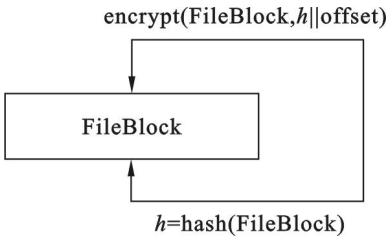

图 1 数据自加密

Fig. 1 Data self-encryption

为了保障云存储中的数据安全,本方案对数据的访问控制权限进行设定,采用基于访问树设置访问控制结构. 总体方案分为四个模块:系统初始化、私钥申请、文件上传及文件下载. 系统总体架构如图 2 所示.

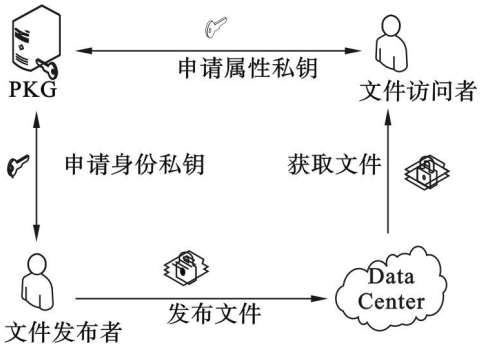

图 2 系统总体架构

Fig. 2 Architecture of system

为了便于描述实体间的交互过程,本方案将采用表 1 的标识进行表述.

2.1 系统初始化

系统初始化主要包括三部分内容:PKG 生成属性公钥 PK_A 和属性主密钥 MK_A、身份公钥 PK_I 和身份主密钥 MK_I;构建属性集合整体;PKG 为每个用户产生属性证书.

1) 生成公共参数 $PK_c = \{p, G_0, G_T, e, g, H\}$,其中 G_0, G_T 为素数 p 阶的循环群,e 为双线性映射 $e: G_1 \times G_1 \rightarrow G_T$,$g$ 为群 G_0 的生成元,H 为哈希函数 $H: \{0,1\}^* \rightarrow G_0$. 然后随机选择三个伪随机

数 $\alpha,\beta,s\in\mathbf{Z}_p$，获得属性公钥 $PK_A=(PK_c,e(g,g)^{\alpha},g^{\beta})$，属性私钥 $MK_A=(\beta,g^{\alpha})$，身份公钥 $PK_I=(PK_c,g^{s})$，身份私钥 $MK_I=(s)$.

表 1　标识及描述
Table 1　Identifier and description

标识	含义
PKG	密钥生成中心
DO	数据拥有者
DU	数据使用者
HDPS	Hadoop 云端服务器
S	用户的属性集合
PK_A	属性公钥
MK_A	属性主密钥
PK_I	身份公钥
MK_I	身份主密钥
$Enc_k(M)$	用密钥 K 加密明文 M
$Dec_k(C)$	用密钥 K 解密密文 C
USB－KEY	用户独有的物理存储介质
key	对称密钥
Hash(F)	计算 F 的 Hash 值
Hash(F)/sign	对 F 的 Hash 值的签名
Verify(file)	判定文件 file 是否被篡改

2）PKG 负责为申请私钥的用户生成与其属性相关的私钥，因此 PKG 需要具备一个包含所有用户属性的全体属性集合. 系统初始化时，PKG 创建属性列表，向其添加不同的属性.

3）在系统初始化阶段，PKG 需要为每个用户生成一个属性证书，将其身份与具有的属性进行绑定，防止用户申请自己不具有的属性所对应的私钥，产生安全隐患.

2.2　私钥申请

当数据使用者从云分享系统中下载完文件后，由于该文件是以密文形式存在的，为了得到最终的文件，需要向密钥发布中心申请其私钥. 当密钥发布中心接收到用户的申请时，会生成对应的私钥，返回给用户. 该过程包括两部分：属性私钥的申请及身份私钥的申请.

1）属性私钥的申请. 方案中的属性私钥申请/颁发是整个系统的关键部分，是 DU 与 PKG 交互. 其中属性私钥分发过程有两方面的难点：一是 PKG 生成的属性私钥如何安全发送到申请者；二是用户如何向 PKG 证明其申请属性集合的合法性. 对应问题一，方案采用 DH 密钥交换解决. 对于问题二，方案中引入属性证书，属性证书包含用户属性集合，属性证书由 PKG 授权机构进行颁发，PKG 验证通过才为相应的属性集合颁发属性私钥. 具体交互过程如图 3 所示.

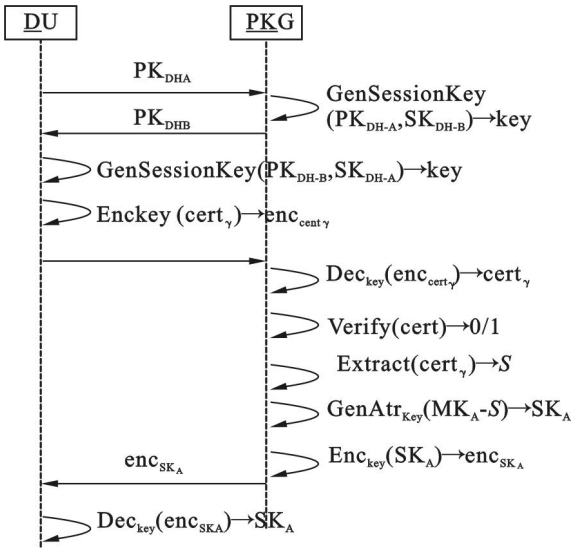

图 3　DU 与 PKG 的交互过程
Fig. 3　Interactive process of DU and PKG

具体流程如下：

① 用户将 DH 公钥 PK_{DH-A} 发送给 PKG，PKG 用自己的 DH 私钥 SK_{DH-B} 生成会话密钥 key.

② PKG 将 DH 公钥 PK_{DH-B} 发送给用户，用户用自己的 DH 私钥 SK_{DH-A} 生成会话密钥 key.

③ 用户用会话密钥 key 加密属性证书 $cert_\gamma$ 生成 enc_{cert_γ}，发送给 PKG.

④ PKG 用会话密钥 key 解密 enc_{cert_γ}，获取属性证书 enc_{cert_γ}.

⑤ PKG 验证属性证书的有效性，如果有效继续，否则结束.

⑥ PKG 提取属性证书中的属性集合 S.

⑦ PKG 用属性集合 S 和主密钥 MK_A 生成属性私钥 SK_A.

⑧ PKG 用会话密钥 key 加密 SK_A 生成 enc_{SK_A}.

⑨ 用户用密钥 key 解密 enc_{SK_A} 获属性私钥 SK_A.

2）身份私钥的申请. 身份私钥的申请过程同属性私钥的申请，身份的验证是通过用户持有的验证数据进行验证，例如用户身份标识为邮件，用户持有的验证数据即为邮件的口令，PKG 将身份私钥发送到对应的邮箱中，只有有邮件口令的用户才可以获取身份私钥.

2.3　文件上传

由数据拥有者执行文件上传. 数据拥有者通

过制定文件的访问控制规则,将文件用访问控制规则加密上传到云端,并对文件摘要进行签名,来验证文件的完整性和数据源真实性.

本方案采用 128 位 ECB 模式的 AES 算法加密欲上传文件. DO 与 HDPS 的交互过程如图 4 所示.

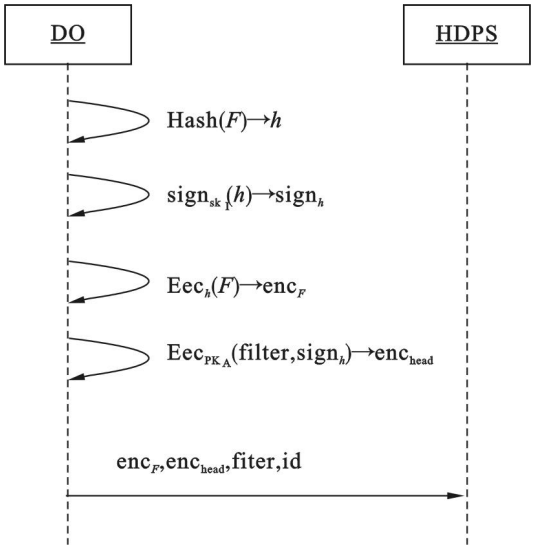

图 4 文件上传时序图

Fig. 4 Sequence diagram of file uploading

具体流程如下:

1) 发布者计算文件 F 摘要为 h.

2) 发布者用私钥对摘要进行签名生成 sign_h.

3) 发布者用 h 加密文件 F 生成 enc_F.

4) 发布者制定访问控制规则 filter,用 filter 加密文件摘要 sign_h 和文件的摘要值 h 生成加密头部 enc_{head}.

5) 将加密后的文件 enc_F,封装的头部 enc_{head},加密规则 filter,用户身份 id 发送到云端.

2.4 文件下载

由数据使用者执行. 当文件的访问者属性私钥对应的属性集合满足文件的访问控制规则,文件访问者可以对文件进行解密,同时还可以进行文件完整性和数据源正确性的判断,防止文件篡改.

具体的系统用户与 Hadoop 云端服务器的交互过程如图 5 所示.

具体流程如下:

1) 访问者向云端请求文件.

2) 云端将加密后的文件 enc_F,封装的头部 enc_{head},加密规则 filter,用户身份 id 返回给访问者.

3) 如果访问者的属性私钥对应的属性集合不满足加密规则 filter,结束,否则解密封装的头部 enc_{head}获取文件摘要 sign_h 和文件的摘要值 h'.

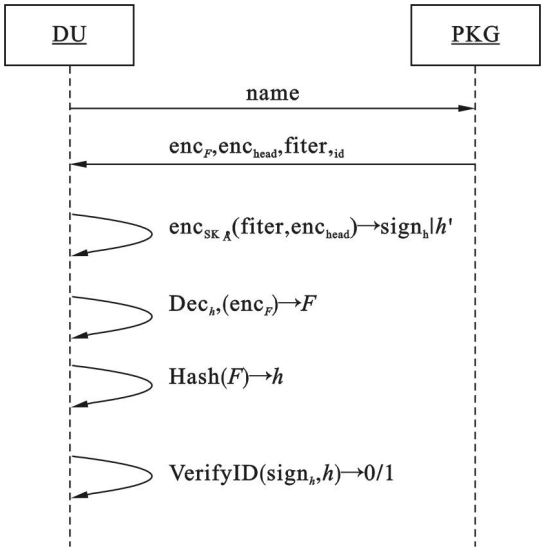

图 5 文件下载时序图

Fig. 5 Sequence diagram of file downloading

4) 访问者用对称密钥 h' 解密加密文件 enc_F 获取文件 F.

5) 计算文件 F 的摘要值 h',判断 h' 与解密获取的摘要值 h 是否相等,不相等表示完整性被破坏,结束,否则继续.

6) 验证摘要的签名是否有效,如果有效表示数据源真实,否则为伪造.

3 方案分析

3.1 数据访问控制机制

数据的访问控制机制主要基于 CP – ABE 方案,在文献[4]中基于判定性 BDHE 假设对 CP – ABE 方案的安全性进行了证明,充分证明的CP – ABE 方案在应对密文破解和共谋攻击上是非常有效的;CP – ABE 方案中加密数据与访问控制规则绑定,解密秘钥与用户属性绑定,当用户属性满足访问控制规则的解密秘钥才可以进行解密,其中的访问控制规则支持属性的“与”、“或”、“门限”操作,满足多数访问控制规则的设定并且支持较好的细粒度. 综上,本方案能够安全、灵活、细粒度地进行访问控制.

3.2 数据的认证性

访问控制的数据支持认证,认证包括完整性认证和数据源认证. 方案利用 Hash 对数据作摘要,数据发布者用私钥对摘要进行签名,从而保证云端数据的篡改和伪造都能被识别出来,保证数据的完整性和数据源的真实性.

3.3　数据存储空间利用率

本方案利用收敛加密技术对数据进行加密后存储在云端，收敛加密技术使相同数据对应的密文也相同，云端即使接收到的是密文也能够识别出冗余数据，从而提高了数据存储空间的利用率.

整个方案在 Hadoop 集群一台 Name 节点（即 Master），两台 Node 节点（即 Slaves），Linux 平台操作系统为 Ubuntu，函数库 OpenSSL，TinyXML，HDFS 环境下，进行了仿真实验，将生成的私钥存放于合法用户的安全物理存储介质 USB－KEY 里，验证了整个方案的有效性.

4　结　　语

本方案设计了密钥发布中心、用户和云服务器三方实体，并使其能够通过 Socket 正常通信. 密钥发布中心负责为用户产生上传和下载文件所需的公私密钥，用户用从密钥发布中心申请的公私密钥向云服务器上传或下载文件. 设计的访问控制方案能高效、灵活、细粒度地进行数据的访问控制，同时方案支持数据认证，包括完整性认证和数据源认证. 方案结合收敛加密技术对文件进行加密，从而使云存储端支持密文冗余检查，增加了存储空间的利用率. 方案不仅能够提供海量的存储空间和计算资源，使互联网服务更加便捷，还能够防止非法用户进入窃取机密数据，保证机密性，保障了云存储环境中数据的安全.

参考文献：

[1] Cachin C, Keidar I, Shraer A. Trusting the cloud[J]. *ACM SIGACT News*, 2009, 40(2): 81－86.

[2] Sahai A, Waters B. Fuzzy identity-based encryption[M]. Heidelberg: Springer, 2005: 457－473.

[3] Goyal V, Pandey O, Sahai A, et al. Attribute-based encryption for fine-grained access control of encrypted data[C]// The 13th ACM Conference on Computer and Communications Security. New York: ACM, 2006: 89－98.

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

[5] 王鹏翩，冯登国，张立武. 一种支持完全细粒度属性撤销的 CP-ABE 方案[J]. 软件学报，2012，23(10)：2805－2816. (Wang Peng-pian, Feng Deng-guo, Zhang Li-wu. CP-ABE scheme supporting fully fine-grained attribute revocation[J]. *Journal of Software*, 2012, 23(10): 2805－2816.)

[6] Hohenberger S, Waters B. Attribute-based encryption with fast decryption[M]. Berlin: Springer, 2013: 162－179.

[7] Douceur J R, Adya A, Bolosky W J, et al. Reclaiming space from duplicate files in a serverless distributed file system[C]// The 22nd International Conference on Distributed Computing Systems. Piscataway: IEEE, 2002: 617－624.

[8] Shamir A. How to share a secret[J]. *Communications of the ACM*, 1979, 24(11): 612－613.

(上接第 1415 页)

语义，索引策略及两种查询算法，并设计了基于分值剪枝及空间剪枝策略的优化机制，最后通过基于多个真实数据集的实验验证了所提算法的有效性.

参考文献：

[1] Yan X, Chen R, Cheng C, et al. Spatial query processing engine in spatially enabled database[C]//Geoinformatics. Beijing: IEEE, 2010: 1－6.

[2] Xia T, Zhang E, Kanoulas E, et al. On computing top-*t* most influential spatial sites[C]//Very Large Data Bases. Trondheim: ACM, 2005: 946－957.

[3] Du Y, Zhang D, Xia T. The optimal-location query[C]// Symposium on Spatial and Temporal Databases. Angra Dos Reis: Springer, 2005: 163－180.

[4] Yiu M, Dai X, Mamoulis N, et al. Top-*k* spatial preference queries[C]//International Conference on Data Engineering. Istanbul: IEEE, 2007: 1076－1085.

[5] Yiu M, Lu H, Mamoulis N, et al. Ranking spatial data by quality preferences[J]. *Transactions on Knowledge and Data Engineering*, 2011, 23(3): 433－446.

[6] Guttman A. R-Trees: a dynamic index structure for spatial searching[C]//International Conference on Management of Data. Boston: ACM, 1984: 47－57.

[7] Joao B, Vlachou A, Doulkeridis C, et al. Efficient processing of top-*k* spatial preference queries[C]//Very Large Data Bases. Seattle: ACM, 2011: 93－104.

[8] Papadias D, Shen Q, Tao Y, et al. Group nearest neighbor queries[C]//International Conference on Data Engineering. Boston: IEEE, 2004: 301－312.

[9] Li H, Lu H, Huang B, et al. Two ellipse-based pruning methods for group nearest neighbor queries[C]// Advances in Geographic Information Systems. Bremen: ACM, 2005: 192－199.