第43卷 第6A期
2016年6月

计 算 机 科 学
Computer Science

Vol. 43 No. 6A
June 2016

基于 Paillier 加密的数据多副本持有性验证方案

王惠清[1] 周 雷[2]

(四川医科大学现代教育技术中心 泸州 646000)[1] (中南大学信息科学与工程学院 长沙 410083)[2]

摘 要 云存储服务中,用户将数据存储在不可信的云储存服务器上,为检查云存储中服务提供商(CSP)是否按协议完整地存储了用户的所有数据副本,提出一种支持对数据副本进行动态操作的基于 Paillier 加密的数据多副本持有性验证方案,即 DMR-PDP 方案。该方案为实现多副本检查,将文件块以文件副本形式存储在云服务器上,将各副本编号与文件连接后利用 Paillier 密码系统生成副本文件以防止 CSP 各服务器的合谋攻击。利用 BLS 签名实现对所有副本的批量验证。将文件标志和块位置信息添加到数据块标签中,以保证本方案的安全性,支持对文件的动态更新操作。安全性分析和仿真实验结果表明,该方案在安全性、通信和计算开销方面的性能优于其他文献提出的方案,极大地提高了文件存储和验证的效率,减少了计算开销。

关键词 云计算,云存储服务提供商,多副本,动态多副本持有性验证,Paillier 密码系统

中图法分类号 TP309.2 文献标识码 A

Multiple-replica Provable Data Possession Based on Paillier Encryption

WANG Hui-qing[1] ZHOU Lei[2]

(Modern Education Technology Center, Sichuan Medical University, Luzhou 646000, China)[1]

(College of Information Science and Engineering, Central South University, Changsha 410083, China)[2]

Abstract In cloud storage service, the user data are stored in untrusted cloud storage server and faced with security threat. In order to check whether all the file replicas are stored by the CSP intactly, a multiple-replica provable data possession scheme based on Paillier encryption and supporting the dynamic operation of data replica was proposed, namely the DMR-PDP scheme. To realize multiple-replica check, the file blocks are stored in the cloud server in the form of copies, and differentiable replicas are generated by using Paillier encryption system to encrypt the concatenation of the serial numbers of replicas and the file. The verifying tags are generated by BLS signature, which can batch checking of all replicas. The information of file identification and block position are added into the block tags to prevent both of the replacing and replay attacks from the CSP. The security analysis and simulation results show that the scheme is better than other literature methods in terms of security, communications and computational overhead, greatly improves the efficiency of file storage and validation, and reduces the computational overhead.

Keywords Cloud computing, Cloud server provider(CSP), Multiple-replica, Dynamic multiple-replica provable data possession(DMR-PDP), Paillier encryption system

云存储服务中,CSP 是不可信的,为了谋取更大的利益,CSP 可能将存储在云服务器上的数据进行篡改和删除。为了保护数据的安全和完整性,研究人员提出数据持有性证明(Provable Data Possession, PDP)来检测云存储中数据的正确性和完整性。数据完整性验证研究主要集中在可恢复证明(Proof of Retrievability, POR)和数据持有性证明,其主要的区别是 PDP 支持检测数据完整性,但无法保证数据可恢复性;POR 可以确保存储数据的可恢复性[1]。

在文献[2]中,Ateniese 等人正式定义了 PDP 方案,并使用基于 RSA 的模指运算构造同态标签来实现持有性证明,但并未考虑数据的动态存储以及多副本数据系统。Ateniese 等人[3]考虑到文献[2]中的静态数据更新问题,提出对基本数据的动态更新,但是并不支持对块数据的插入操作。文献[4]中,Erway 等人扩展了 PDP 模型以支持对存储的数据进行动态更新,但是该方案提出的方法比较模糊,且仅仅支持对单一数据备份进行完整性验证。文献[5]中,Wang 等人运用梅克尔散列树(MHT)进行数据的完整性验证,并支持数据的动态操作,然而,该方案对数据并没有加密,且仅对单一副本数据的动态操作有效。文献[6]中,Hao 等人提出一种,既可以支持数据的动态更新也能够进行公开验证的方案,允许任何人通过执行挑战_反馈协议来验证存储在云服务器上的数据的完整性,但是该方案也没有考虑到对数据进行加密,以及对多副本数据存储的应用。文献[7]中,Curtmola 等人提出一个多副本持有性数据验证(Multiple-Replica Provable Data Pos-

本文受国家自然科学基金青年科学基金项目(51308465),四川医科大学校级课题(JG2015086)资助。

王惠清(1983—),男,硕士,助教,主要研究方向为计算机网络、云计算、分布式系统;周 雷(1982—),男,博士,讲师,主要研究方向为网格计算、分布式存储、系统体系结构。

session)方案，在该方案中数据拥有者能够验证一个文件的几个副本，不同的副本由最初加密的数据创建，然后，由伪随机函数(PRF)生成的随机数将其屏蔽，这些随机化的数据就存储在多服务器上。该方案将RSA签名用于标签的创建，但是该方案并没有提出数据授权用户如何访问云服务器上的文件副本，而且不支持数据的动态更新。

综合考虑以上方案，本文提出一种基于Paillier的数据多副本持有型验证方案(DMR-PDP方案)，并且支持对数据副本的动态操作。本方案运用Paillier加密副本，运用BLS签名创建副本的标签，同时支持对存储在云服务端的副本进行动态更新操作。安全性分析和实验结果显示，与文献[8]相比，本方案是正确和安全的。

1 DMR-PDP方案结构

本方案的云计算数据存储模型由3部分构成，包括数据拥有者(Data Owner)：可以是个人或是组织，拥有存储在云服务器上的原始数据；云服务提供商(CSP)：管理云服务器以及提供为用户提供付费存储空间进行数据存储；授权用户(Authorized User)：一系列具有访问远程数据权限的用户，且拥有数据拥有者分发的密钥。具体方案结构如图1所示。

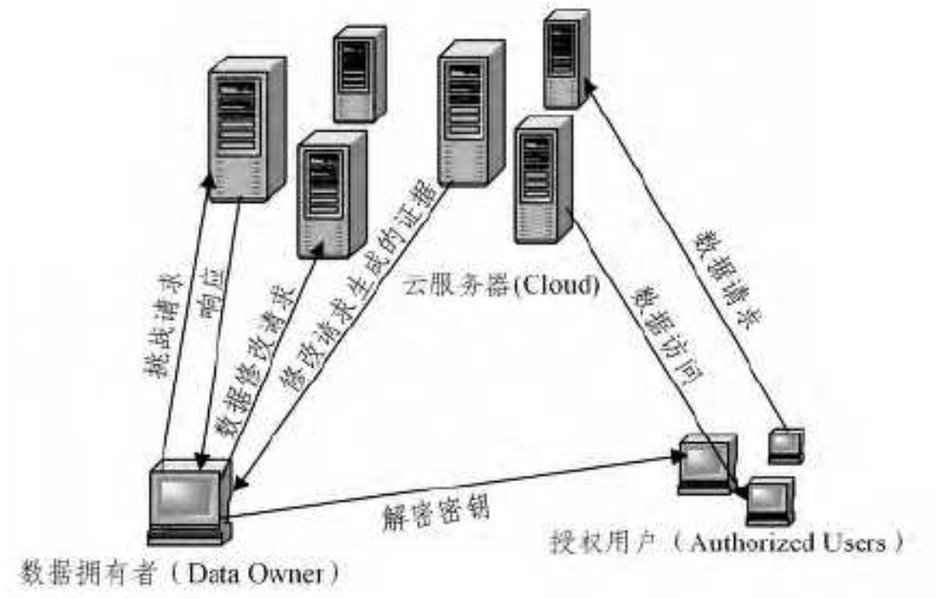


图1 方案结构

2 DMR-PDP持有性验证方案算法

2.1 相关定义

本算法中用到双线性映射和Paillier加密方案。下面给出具体介绍：

1)假设外包的数据文件为F，由m个序列的文件块组成，如$F=\{b_1,b_2,\cdots,b_m\}$，$b_i\in Z_N$，这里Z_N是一个余数的集合，N在Paillier加密方案中是一个公钥。

2)设F_i是第i个文件副本，因此$F_i=\{b_{i1},b_{i2},\cdots,b_{im}\}$，其中$b_{ij}$表示第$i$个文件副本的文件块$b_j$。

3)BLS签名：BLS签名是较短的同态签名，运用某些椭圆曲线的双线性对性质。这些签名允许并发数据验证，许多数据块能够同时进行验证[9,10]。

4)双线性对：设G_1，G_2和G_T是素数阶a的循环群，u，v分别是由G_1，G_2生成。双线性映射$e:G_1\times G_2\to G_T$具有如下性质[11]：

·双线性：

$$e(u_1u_2,v_1)=e(u_1,v_1)\cdot e(u_2,v_1)$$

$$e(u_1,v_1v_2)=e(u_1,v_1)\cdot e(u_1,v_2)$$

其中，$\forall u_1,u_2\in G_1;v_1,v_2\in G_2$。

·非退化性：

$e(u,v)\neq 1$

5)$H(\cdot)$是一个哈希函数[12]：$\{0,1\}^*\to G_1$。

6)Paillier加密：Paillier密码系统是同态概率加密方案[13]。其步骤如下：

·计算$N=p*q$，$\lambda=LCM(p-1,q-1)$，p，q是素数。

·选择一个是N的倍数的随机数g，$g\in Z_{N^2}^*$。

·公钥为(N,g)，私钥为λ，$N=p*q$。

·信息m的密文被计算作为$c=g^m r^N \bmod N^2$，r是随机数，$r\in Z_N^*$，$c\in Z_{N^2}^*$，$m\in Z_N$。

·明文通过$m=L(c^\lambda \bmod N^2)*(L(g^\lambda \bmod N^2))^{-1} \bmod N$计算得到。

7)公钥g在Paillier方案中性质：

·$g\in Z_{N^2}^*$

·如果$g=(1+N) \bmod N^2$，则有如下性质：

①$(1+N)$的阶是N。

②$(1+N)^m\equiv(1+mN) \ mod \ N^2$

$(1+mN)$代替$(1+N)^m$，是为了避免数据加密过程中的指数计算开销。

2.2 DMR-PDP算法

本文中，数据拥有者创建多加密副本，上传到云服务器。CSP将这些副本数据存储到不同地理位置的多个服务器上，以提高容灾抗毁和数据可恢复的能力。数据拥有者与授权用户共享解密密钥。授权用户访问远程数据时，需要发送一个数据请求到CSP，然后接收一个加密的数据副本，通过共享的解密密钥对加密的数据副本进行解密。该方案算法共7个步骤：KeyGen、ReplicaGen、TagGen、Prove、Verify、PrepareUpdate、ExecUpdate。其具体算法如下。

(1)$(pk,sk)\leftarrow KeyGen()$。数据拥有者运行该算法，生成公钥$pk$和私钥$sk$；同时还生成5个密钥集。a)生成数据标签使用的密钥：该密钥用于为数据生成数据标签。数据拥有者选择一个双线性映射e和一个私钥$l\in Z_a$，l是私钥，公钥为$y=v^l\in G_2$，v由循环群G_2生成。b)数据加密的密钥：该密钥用于加密数据和创建多副本数据。数据拥有者选择$Paillier$公钥(N,g)，其中$g=(1+N)\bmod N^2$，以及私钥λ。c)用于验证的PRF密钥：数据拥有者生成一个PRF密钥Key_{PRF}，以及由此密钥生成s个随机数，每个随机数都被用于创建一个数据副本。设$\{k_1,k_2,\cdots,k_n\}\in Z_N^*$是由PRF密钥$key_{PRF}$生成的随机数序列。数据拥有者保存密钥$Key_{PRF}$，这样云存储服务端并不知道这些用于创建多副本的随机数。d)用$Paillier$加密的PRF密钥：数据拥有者生成一个PRF密钥Key_{rand}，用于在$Paillier$加密中生成随机数。e)用于标签生成的PRF密钥：数据拥有者生成一个PRF密钥key_{tag}，用于标签的生成。

(2)$\{F_i\}_{1\leqslant i\leqslant s}\leftarrow ReplicaGen(s,F)$。该算法由数据拥有者运行，用于生成数据副本。输入副本数量s和文件F，生成s个独特且可区分的文件副本$\{F_i\}_{1\leqslant i\leqslant s}$。运用Paillier加密算法得到文件$F$的$s$个加密副本。对于一个文件$F=\{b_1,b_2,\cdots,b_m\}$，运用Paillier加密算法生成多数据副本$F_i$，即

$$F_i=\{(1+N)^{b1}(k_ir_{i1})^N,(1+N)^{b2}(k_ir_{i2})^N,\cdots,(1+N)^{bm}(k_ir_{im})^N\}_{1\leqslant i\leqslant m}$$

其中，i表示文件副本数，k_i表示由PRF密钥key_{PRF}生成的随机数，r_{ij}表示由PRF密钥Key_{mnd}生成的随机数。在文件块中k_i标识属于哪个文件块副本。

(3)$\phi\leftarrow TagGen(sk,F)$。该算法由数据拥有者执行。私钥$sk$和文件$F$作为输入，输出标签$\phi$。本方案使用BLS签名

创建数据标签，BLS 签名的特征是签名短、同态性和数据的并发验证，因此多副本数据块可以同时被验证。本方案中，为每个文件块 b_i 生成的标签为 $\phi_i=(H(F)\cdot u^{b_iN+a_i})^l\in G_1$，$u\in G_1$，$H(\cdot)\in G_1$ 表示文件 F 的哈希值，$\{a_i\}_{1\leqslant i\leqslant m}$ 是由 PRF 密钥 key_{tag} 生成的随机数，随机化的目的是避免对类似的数据块产生相同的标签。然后数据拥有者发送副本 $\{F_i\}_{1\leqslant i\leqslant s}$ 的标签集 $\phi=\{\phi_i\}_{1\leqslant i\leqslant m}$ 到云服务端。

(4) $P\leftarrow Prove(F,\phi,challenge)$。证据生成算法。该算法由 CSP 执行。由数据拥有者发送到云端文件的副本、标签和挑战块向量作为输入，返回云服务端生成的证据 P。该证据 P 保证 CSP 的确存储了文件的所有副本，且数据副本都是完整的。算法由以下两个过程组成：a) 挑战：数据拥有者发送挑战到云服务端来验证外包数据的完整性。有两种类型的验证方案：确定性（所有副本的所有文件块都用于数据验证）和概率性（仅副本中的一部分数据块被用于验证），每次挑战，数据拥有者选择相应的验证类型。b) 响应：这个过程由 CSP 在接收挑战以后执行。本方案选择概率性验证方案展示证据。CSP 接收两个来自数据拥有者的 PRF 密钥，即 key_1 和 key_2。运用 key_1 密钥，CSP 生成 c 个随机数的文件块索引集 $\{C\}_{1\leqslant c\leqslant m}$，运用 key_1 密钥，生成 s 个随机数的标签集 $T=\{t_1,t_2,\cdots,t_s\}$。云端执行两种操作，一种在标签上，另一种在文件块上。a) 计算标签 σ，即

$$\sigma=\prod_{j\in C}(H(F)\cdot u^{b_jN+a_j})^l=\prod_{j\in C}(H(F))^l\cdot\prod_{j\in C}u^{(b_jN+a_j)l}$$
$$=H(F)^d\cdot u^{(N\sum_{j\in C}b_j+\sum_{j\in C}a_j)l} \tag{1}$$

b) 计算文件块 μ，即

$$\mu=(1+N\sum_{i=1}^{s}(t_i)\sum_{j\in C}(b_j))(\prod_{i=1}^{s}(k_i)^{Nct_i})(\prod_{i=1}^{s}(\prod_{j\in C}(r_{ij})^{t_{iN}})) \tag{2}$$

然后，CSP 发送 σ 和 $\mu \bmod N^2$ 到 owner。

(5) $\{1,0\}\leftarrow Verify(pk,P)$。证据验证算法。该算法由数据拥有者运行。输入公钥 pk 和 CSP 返回的证据 P，如果待验证的所有数据副本的完整性验证都是正确的，返回 1；否则返回 0。接收到 σ 和 μ 的值后，owner 首先计算

$$v=(\prod_{i=1}^{s}(k_i)^{t_{ikN}}) \tag{3}$$

$$d=Decrypt(\mu)*Inverse(\sum_{i=1}^{s}t_i) \tag{4}$$

其次，验证 $\mu*Inverse(\prod_{i=1}^{s}(r_i)^{t_{ikN}})\bmod v\equiv 0$ 是否成立。若成立，则确保云服务端在响应阶段使用了所有文件副本；验证 $(H(F)^c u^{dN+\sum_{j\in C}a_j})^l=\sigma$ 是否成立，若成立，则确保云服务端在响应阶段使用了所有文件块。如果上述等式都成立，则说明 owner 确定存云服务器上的数据是完整的，其 CSP 存储了数据的所有副本。

(6) $Update\leftarrow PrepareUpdate()$。该算法由 owner 对存储在 CSP 端的外包的文件副本执行更新操作，输出更新请求。owner 以 $\langle Id_F,BlockOp,j,b_i',\phi'\rangle$ 的形式发送更新操作请求到云存储服务端，这里 Id_F 是文件标识器，$BlockOp$ 对应具体的更新操作类型，j 表示文件块的索引，b_i' 表示更新的文件块，ϕ' 是更新的标签，$BlockOp$ 的更新类型有数据修改、插入、删除操作。

(7) $(F',\phi')\leftarrow ExecUpdate(F,\phi,Update)$，执行更新算法。该算法由 CSP 运行，输入文件副本 F、标签 ϕ 以及更新请求，输出随有更新签名 ϕ' 的文件副本 F' 的更新版本。对一些文件块操作以后，数据拥有者运行挑战协议来确保云存储服务端正确地执行了数据更新操作。更新操作可以修改一个文件块、插入一个文件块和删除一个文件块。

a) Modification 操作：数据修改是最经常使用的动态操作之一。本文提到的 DMR-PDP 方案中的数据修改操作如图 2 所示。

owner　　CSP
1. 加密新文件块 s 次。
2. 为新文件块创建标签 ϕ。
3. 生成 PRF 密钥 key_1,key_2 来验证插入操作的正确性。

$\langle Id_F,insert,j,s,\phi\rangle,key_1,key_2$ →

4. 插入新文件块在第 j 个位置。
5. 存储新的标签 ϕ。
6. 计算响应 μ,σ。

μ,σ ←

7. 计算 v,d。
8. 验证 $\mu*Inverse(\prod_{i=1}^{s}(r_i)^{t_{icN}})\bmod\equiv 0$ 和 $H(F)^d\cdot u^{(Nd+\sum_{j\in C}a_j)l}=\sigma$ 是否相等。

图 2　数据块修改操作

b) Insertion 操作：文件块执行插入操作时，owner 在文件的第 j 个位置插入一个新的文件块，如果文件 F 有 m 数据块，执行插入操作后，文件 F 将有 $m+1$ 数据块。文件块插入操作如图 3 所示。

owner　　CSP
1. 计算 $\Delta b_j=b_j'-b_j$。
2. 使用双线性加密方法加密 Δb_j，$E(\Delta b_j)=(1+\Delta b_jN)r^N$，r 是随机数。
3. 计算新的文件标签 b_j'，$\phi'=(H(F)u^{b_j'N})^l$。
4. 生成 PRF 密钥 key_1,key_2 来验证修改操作的正确性。

$\langle Id_F,modify,j,E(\Delta b_j),\phi'\rangle,key_1,key_2$ →

5. 执行同态加法操作对所有的文件副本。
6. 删除旧的标签，用新标签替换。
7. 计算响应 μ,σ。

μ,σ ←

8. 计算 v,d。
9. 验证 $\mu*Inverse(\prod_{i=1}^{s}(r_i)^{t_icN})\bmod\equiv 0$ 和 $H(F)^d\cdot u^{(Nd+\sum_{j\in G}a_j)l}=\sigma$ 是否相等。

图 3　数据块插入操作

c) Deletion 操作：删除操作与插入操作正好相反，删除一个文件块时，后面的所有文件块都向前移动一个位置。为了从所有副本中删除第 j 个位置的指定块，owner 发送删除请求 $\langle Id_F,delete,j,\text{null},\text{null}\rangle$ 到云存储服务端，CSP 收到请求以后，删除所有文件副本的标签和索引位置为 j 的文件块。

3　安全性分析

这部分主要对所提出的 DMR-PDP 方案的安全性进行分析。数据拥有者将加密的文件存储在不可信的云端，将云端服务作为本方案的主要敌手，只要云端不是通过删除文件及数据拥有者仍然通过验证来欺骗用户，本方案就是安全的。

1) 针对敌手伪造响应的安全分析：本方案在挑战阶段，owner 发送两个 PRF 密钥，一个参数‘c’，用来指示需要验证文件块的数量，DMR-PDP 方案对于 owner 发送验证挑战提供了灵活性，在每次向 CSP 发送挑战的阶段，owner 可以发送不同的‘c’和 PRF 密钥，这就确保了每次由 owner 发送的挑

战及 CSP 产生的响应都是不同的，从而消除了 CSP 没有经过精确计算就伪造响应的可能。

2)针对相同索引值的文件块删除的安全分析：文件分成数据块时，一些数据块可能会有相同的索引值，数据标签的值也可能相同，但是文件块的密文的值不同。通过文件块加密，加密的文件块有不同的值，云端就能够通过相同值的标签识别出相同值的文件块。为了避免 CSP 欺骗用户，DMR-PDP 方案在创建标签之前，随机化数据文件。然后在标签数据中添加由密钥 key_{tag} 生成的随机数，这样即使数据标签值相同，文件块值也不相同。对于文件块 $b_i=b_j$：

$$Tag(b_i)=(H(F)\cdot u^{b_i N+a_i})^l$$

$$Tag(b_j)=(H(F)\cdot u^{b_j N+a_j})^l a_i, a_j$$

是由密钥 key_{tag} 生成的随机数，尽管数据块是一样的，但是生成的标签值是不同的。

3)Paillier 密码系统的安全分析：DMR-PDP 方案运用 Paillier 密码系统，因此，DMR-PDP 方案的安全性依赖于 Paillier 密码系统的安全。依据文献[8]提出的方案，公钥 $g=N+1$ 满足语义安全和单向性属性。如果公钥 g 是由公有信息 N 生成，那么 Paillier 密码系统与分解模数 N 一样困难。文献[8,10]都将 $g=N+1$ 运用在改进后的 Paillier 加密系统中。因此，DMR-PDP 方案也是安全的。

DMR-PDP 方案的安全性来自以上安全方案的结合，Paillier 加密提出数据安全问题，PDP 方案提出数据可用性问题。数据安全由 Paillier 加密方案给予保障，对于 PDP 的保证，其算法是 Paillier 加密数据生成 μ，运用 BLS 签名的标签生成 σ。这些算法操作的安全性在文献[8]和文献[5]中都给予说明，PDP 方案的安全性在文献[1]中给予分析。DMR-PDP 方案的安全性通过结合上述这些方案得到保证。

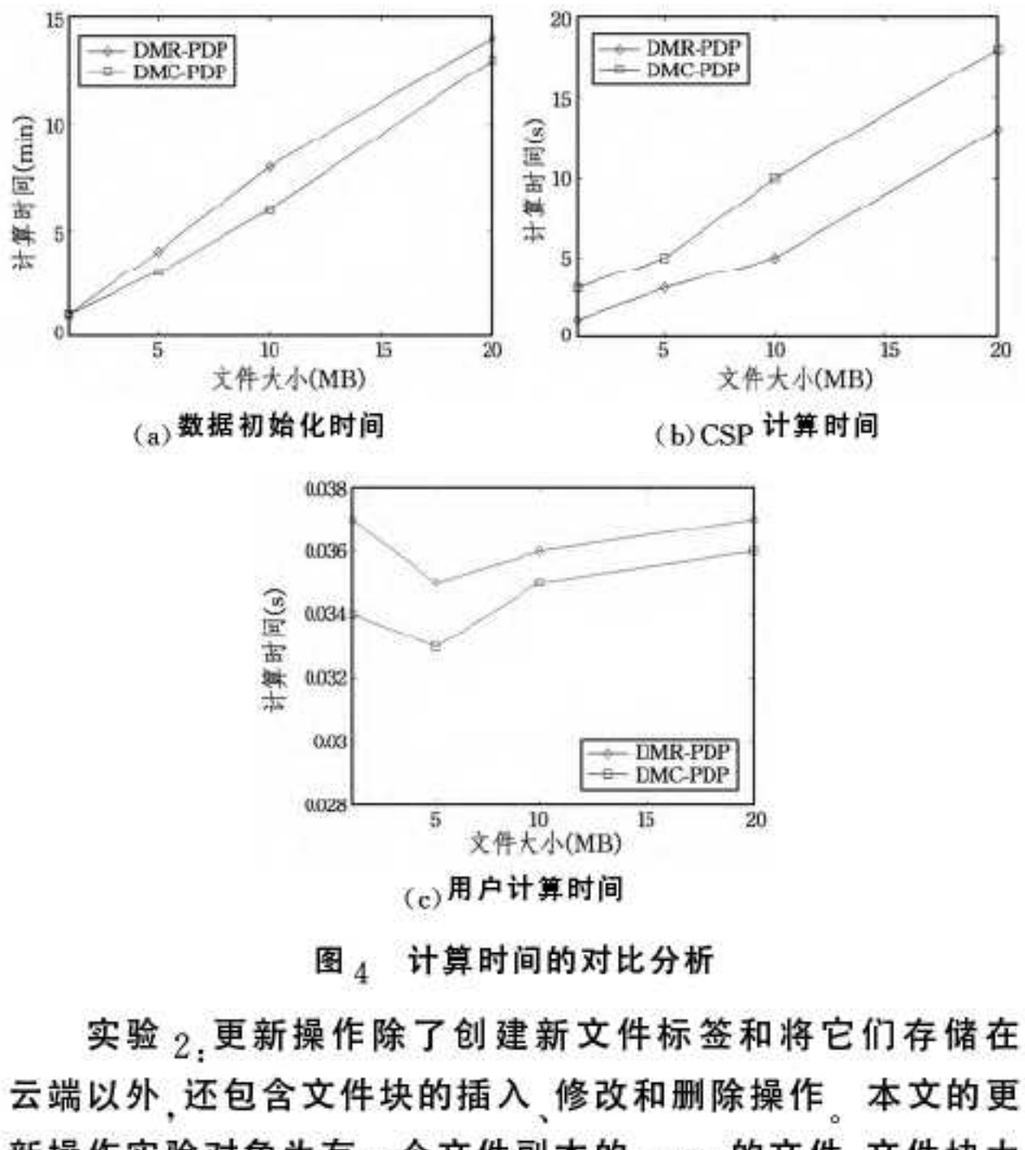
4 实验结果和分析

本方案的算法实现过程使用 C 语言编写，实验数据是通过装有 CentOS 系统的本地服务器和配有不同参数的 EC2 云平台得到的。在该云平台环境下测量用户和 CSP 执行各种操作的计算开销，此外，也对通信开销的延迟进行测量。实验运行在 Intel(R)Xeon(R)2.67GHz 处理器、11GB RAM、CentOS6.3 操作系统上，PBC 类库的版本是 0.5.11。在本方案中，每个阶段的通信开销如表 1 所列。

表 1 DMR-PDP 通信对比(bit)

阶段	开销	从	到
Challenge	256	owner	Cloud
Verification	2048+160	Cloud	owner
Update	2048+160	owner	Cloud

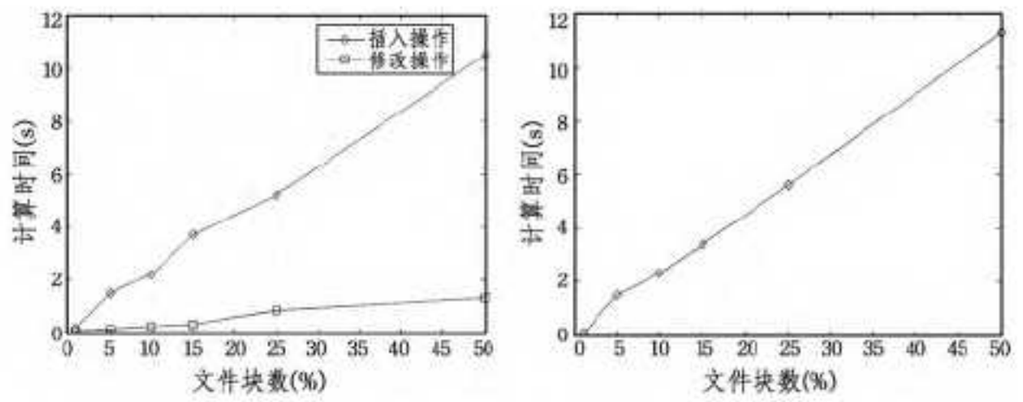
实验 1：对比分析本方案中的 DMR-PDP 算法和文献[8]中提到的 DMC-PDP 算法的性能。图 4 显示随有 3 个文件副本(文件大小为 1、5、10 和 20MB)的数据初始化，CSP 和用户的计算开销。数据初始化在数据拥有者端执行一次，图 4(a)显示 DMR-PDP 和 DMC-PDP 的数据初始化开销。图 4(b)显示 CSP 计算开销的性能对比。DMR-PDP 和 DMC-PDP 方案中的 CSP 计算开包含两种操作：一种是对文件标签的操作，另一种是对文件块的操作。在 DMR-PDP 方案中，数据拥有者对所有文件副本仅创建一个标签集；而 DMC-PDP 方案中，数据拥有者为每个文件副本创建文件标签。因此，与 DMC-PDP 方案相比，本文中的 DMR-PDP 方案在文件标签操作上，CSP 的计算开销较小；对于文件块的操作，两种方案的计算开销基本相同。所以就 CSP 计算时间开销来说，文中的 DMR-PDP 方案比文献[8]中的 DMC-PDP 方案性能更好。图 4(c)显示了用户计算开销的对比，DMR-PDP 方案中的用户计算时间花费在图 2 中的步骤 8、9。在 DMR-PDP 方案中，用户计算开销比 DMC-PDP 方案多执行一个额外的加密功能，其他的开销上两种方案都一样。因此，DMC-PDP 方案在用户计算开销方面性能较好，但是两种方案的用户计算开销的差别很小。



(a)数据初始化时间　(b)CSP 计算时间　(c)用户计算时间

图 4 计算时间的对比分析

实验 2：更新操作除了创建新文件标签和将它们存储在云端以外，还包含文件块的插入、修改和删除操作。本文的更新操作实验对象为有 3 个文件副本的 1MB 的文件，文件块大小为 128 字节。图 5(a)显示了用户和 CSP 共同对文件块进行单独的插入和修改操作所花费的计算开销。图 5(b)显示数据拥有者在文件块上既运行插入操作也运行修改操作的计算开销。可以看出，运行修改操作比插入操作所花费的计算开销要小得多，修改操作的计算开销取决于两个 256 字节加密文件块的 Paillier 加密的计算时间，而插入操作的计算开销取决于写入 256 字节加密文件块到硬件设备的时间。然而，这里并未计算删除操作的时间开销，是因为删除操作并不涉及用户和 CSP 的计算开销。



(a)插入、修改操作的计算时间　(b)插入+修改操作的计算时间

图 5 文件块的插入、修改计算对比分析

结束语　本文提出了一种在云计算环境下验证副本数据完整性的方案，即 DMR-PDP 方案。该 DMR-PDP 方案会定期地对存储在云端的多副本数据进行正确性和完整性验证。本方案在验证过程中引入动态的数据更新操作，使用 Paillier 加密算法对数据副本进行加密实验结果显示本方案在动态数

(下转第 409 页)

结束语　目前绝大多数影响力问题的研究都是基于静态网络的，我们针对网络演化给出了一种基于事件的影响力分析方法，首先描绘出网络演化中的影响力扩散模型，总结了网络演化中扩散的特点，即不断出现的社区划分和节点的加入离开，接下来对网络演化中的个体事件和社区事件进行了定义，实验中对10年的计算机学科研究热点进行分析，基于个体事件给出了两个指标即社交指数和影响力指数来衡量学科的影响力，并找出影响力传播过程中的关键节点，最后对两个指标的性能进行了对比分析，结果表明用社交指数和影响力指数挖掘出的节点分别在扩散初期和扩散的瓶颈期更有利于信息的扩散。

参 考 文 献

[1] Goldenberg J, Libai B, Muller E. Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic cellular automata[J]. Academy of Marketing Science Review, 2001, 9(3): 1-18

[2] Goldenberg J, Libai B, Muller E. Talk of the network: A complex systems look at the underlying process of word-of-mouth [J]. Marketing Letters, 2001, 12(3): 211-223

[3] Ma H, Yang H, Lyu M R, et al. Mining social networks using heat diffusion processes for marketing candidates selection[C]// Proceedings of the 17th ACM Conference on Information and Knowledge Management. ACM, 2008: 233-242

[4] Richardson M, Domingos P. Mining knowledge-sharing sites for viral marketing[C]// Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2002: 61-70

[5] Kempe D, Kleinberg J, Tardos é. Maximizing the spread of influence through a social network[C]// Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2003: 137-146

[6] Galstyan A, Musoyan V, Cohen P. Maximizing influence propagation in networks with community structure[J]. Physical Review E, 2009, 79(5): 056102

[7] Cao T, Wu X, Wang S, et al. OASNET: an optimal allocation approach to influence maximization in modular social networks[C]// Proceedings of the 2010 ACM Symposium on Applied Computing. ACM, 2010: 1088-1094

[8] Asur S, Parthasarathy S, Ucar D. An event-based framework for characterizing the evolutionary behavior of interaction graphs [J]. ACM Transactions on Knowledge Discovery from Data (TKDD), 2009, 3(4): 913-921

[9] Berger-Wolf T Y, Saia J. A framework for analysis of dynamic social networks[C]// Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2006: 523-528

[10] Habiba, Yu Y, Berger-Wolf T Y, et al. Finding spread blockers in dynamic networks[M]// Advances in Social Network Mining and Analysis. Springer Berlin Heidelberg, 2010: 55-76

[11] Zhuang H, Sun Y, Tang J, et al. Influence maximization in dynamic social networks[C]// 2013 IEEE 13th International Conference on Data Mining (ICDM). IEEE, 2013: 1313-1318

[12] Wu Bin, Wang Bai, Yang Sheng-qi. the evolution of the social network analytical framework based on events [J]. Journal of Software, 2011(7): 1488-1502

[13] Ilhan N, Oguducu I G. Community Event Prediction in Dynamic Social Networks[C]// 2013 12th International Conference on Machine Learning and Applications (ICMLA). IEEE, 2013: 191-196

(上接第373页)

据操作方面的性能优于之前所提方案的，且计算开销较小。云存储环境下数据完整性验证问题的研究是一个非常活跃的方向，关于云存储数据完整性验证方面的研究还不成熟，无论是技术方面还是理论方面都有很大的不足，希望科技工作者们能够继续深入研究，争取有更大的研究成果。

参 考 文 献

[1] 陈兰香. 一种基于同态Hash的数据持有性证明方法. 电子与信息学[J]. 2011, 33(9): 2200-2204

[2] Ateniese G, Berns R, Cutmola R, et al. Provable Data Possession at Untrusted Stores[C]// Proc of the 14th ACM Conference on Computer and Communications Security. New York: ACM, 2007: 598-609

[3] Ateniese G, Pietro, R D, Mancini L V, et al. Scalable and Efficient Provable Data Possessin[C]// Proc of the 4th International Conference on Security and Privacy in Communication Netowrks Istanbul. Turkey: ACM, 2008: 1-10

[4] Erway C, Kupcu A, Papamanthou C, et al. Dynamic Provable Data Possession[C]// Proc of the16th ACM Conferenceon Computer andCommunications Security. Chicago, Illinois, USA: ACM, 2009: 213-222

[5] Wang Q, Wang C, Li J, et al. Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing[C]// Proc of the14th European Conferenceon on Research in Computer Security. Heidelberg, Berlin, 2009: 355-370

[6] Hao Z, Zhong S, Yu N. A Privacy-preserving Remote Data Integrity Checking Protocol with Data Dynamics and Public Verifiability[J]. IEEE Transactions on Knowledge and Data Engineering, 2011, 23(9): 1432

[7] Curtmola R, Khan O, Burns R, et al. Multiple-Replica Provable Data Possession [C]// 28th IEEE ICDCS. 2008: 411-420

[8] Barsoum A F, Hasan M A. On Verifying Dynamic Multiple Data Copies over Cloud Servers [R]. Cryptology ePrint Archive, 2011: 447

[9] Damgard I, Ren K, Lou W, et al. Toward Publicly Auditable Secure Cloud Data Storage Services[J]. IEEE Network, 2011, 24: 19-24

[10] Wang C, Jurki M. A Simplification and some Application of Paillier's Probabilistic Public Key System [C]// 4th Internation Workshop on Practice and Theory in Public Key Cryptosystems. 2001: 13-15

[11] 于洋洋, 虞慧群, 范贵生. 一种云存储数据完整性验证方法[J]. 华东理工大学学报(自然科学版), 2013, 39(2): 211-216

[12] 胡德敏, 余星. 一种基于同态标签的动态云存储数据完整性验证方法[J]. 计算机应用研究, 2014, 31(5): 1362-1365

[13] Chen L, Guo G. An Efficient Remote Data Possession Checking in Cloud Storage[J]. International Journal of Digital Content Technology and its Applications, 2011, 5(4): 43-50