# 云计算下的数据存储安全可证明性综述*

梁 彪[1,3†], 曹宇佶[2], 秦中元[2,3], 张群芳[4]

(1. 南京三宝科技股份有限公司, 南京 210049; 2. 东南大学 信息科学与工程学院, 南京 210096; 3. 信息网络安全公安部重点实验室, 上海 201204; 4. 南京炮兵学院 计算机教研室, 南京 211132)

**摘 要:** 云计算的数据服务外包可以减少数据所有者本地的存储和维护压力,然而用户会因此失去对数据可靠性和安全的物理控制。于是如何确保云中数据的安全就成为了非常有挑战性的任务和难题。在全面研究云计算数据存储安全现有成果的基础上,介绍了云计算数据存储的基本架构,并从可检索证明和可证明数据拥有两个角度分析了相关研究方案的发展,从公共认证、同态认证、数据动态化、隐私保护、批审计和多服务器环境得方面讨论了协议的功能设计,并且列表进行了功能和开销对比,在此基础上提出了一个比较完备的云计算环境下的协议框架。最后总结并阐述了后续工作。

**关键词:** 云计算; 数据存储安全; 可检索证明; 可证明数据拥有

**中图分类号:** TP309 **文献标志码:** A **文章编号:** 1001-3695(2012)07-2416-06

doi:10.3969/j.issn.1001-3695.2012.07.004

## Survey of proofs on data storage security in cloud computing

LIANG Biao[1,3†], CAO Yu-ji[2], QIN Zhong-yuan[2,3], ZHANG Qun-fang[4]

(1. *Nanjing Sample Technology Co.,LTD, Nanjing* 210049, *China*; 2. *School of Information Science & Engineering, Southeast University, Nanjing* 210096, *China*; 3. *Key Lab of Information Network Security, Ministry of Public Security, Shanghai* 201204, *China*; 4. *Computer Department , Nanjing Institute of Artillery Corps, Nanjing*, 211132, *China*)

**Abstract:** In cloud computing, data service outsourcing relieves the data owners of the burden of local data storage and maintenance. However, it also eliminates their physical control of the storage dependability and security, which makes the outsourced data assurances in cloud computing a very challenging and potentially formidable task. Based on the overall study of the recent researches on data storage security, it introduced the architecture of cloud data storage firstly. And studied the state-of-the-art schemes from proofs of retrievability and provable data possession. Then, this paper discussed the function design in protocol based on public verification, homomorphic authenticators, data dynamics, data privacy protection, batch auditing and multiple-server environment. Furthermore, made some comparisons of functions and the complexity in different protocols in tables. At last, summarized and discussed the further direction of research.

**Key words:** cloud computing; data storage security; proofs of retrievability(POR); provable data possession(PDP)

## 0 引言

自从云计算的概念提出以来,IT 界就对此产生了浓厚的兴趣。目前,Google、亚马逊、微软、IBM、雅虎、Oracle、Dell、Sun 等国际上知名的 IT 公司都投入了大量资金和人员,积极地研究和部署云计算,并已经开始提供云计算商业服务。比较成熟的云计算业务和应用包括 Google 的 AppEngine、Amazon 的弹性计算云 EC2 和简单存储服务 S3、微软的 Azure 云平台和 IBM 的蓝云等。

云计算的定义有很多种,得到业界最广泛认可的是 2011 年由美国国家标准和技术研究院(NIST)组织云计算产业界的主要厂商经过多次讨论研究后提出的[1]:云计算是一种通过网络以便捷、按需的形式从共享的可配置的计算资源池(这些资源包括网络、服务器、存储、应用和服务)中获取服务的业务模式。云计算业务资源应该支持通过简洁的管理或交互过程快速地部署和释放。

在云计算环境下,用户可以在云中远程存储自己的数据,从而享受到高质量的应用和服务,而且是按需提供的。以用户(包括个人和 IT 企业)的角度来看,以一种灵活的按需的方式把数据存储到云中可以带来如下好处[2]:a)解除了存储管理的负担;b)访问数据不受限于地理位置;c)避免了硬件、软件、人力维护等的大量投资。

云计算为人们的生活带来了很多好处,但是由于云计算的特殊性,仍然面临许多安全威胁。首先,云服务供应商(CSP)会面临内部和外部的各种安全威胁,许多著名的云计算服务商都出现过服务故障。例如,Amazon S3 的 downtime 事件[3],Gmail 的大量邮件被删除事件[4],Apple 的 MobileMe 的 post-launch downtime 事件[5]。其次,CSP 为了自身的利益可能会删

**收稿日期:** 2011-11-28; **修回日期:** 2012-01-06 **基金项目:** 国家科技支撑计划资助项目(2011BAF16B00);信息网络安全公安部重点实验室开放课题基金资助项目(C11605);江苏省网络与信息安全重点实验室开放课题基金资助项目(BM2003201)

**作者简介:** 梁彪(1969-),男(通信作者),江苏南京人,博士,主要研究方向为云计算安全;曹宇佶(1987-),男,江苏常州人,硕士研究生,主要研究方向为网络安全;秦中元(1974-),男,河南安阳人,副教授,博士,主要研究方向为云计算安全(zyqin@seu.edu.cn);张群芳(1981-),女,助教,硕士,主要研究方向为网络安全.

除用户不访问或很少访问的数据,或者由于数据的备份、转移等原因造成的数据丢失,而等到用户发现已经太晚了。

虽然以上情况发生的概率很小,因为云服务器的防护级别显然要高于个人计算机的级别。但是云服务器一旦发生问题,会损失成千上万的用户数据,从而严重影响用户把数据转移到云服务器上的信心。因此,如何确保用户数据完整地存储在云服务提供商上就成为一个研究的热点。

要实现这个目的,最简单的方法是把用户数据下载到本地并进行比对,但是这种方案有两个问题:a)用户的数据量很大,全部下载到本地需要消耗大量的网络带宽和本地存储空间;b)用户很可能在本地并没有备份。为了解决这个问题,最近国际上从可证明性角度提出了很多不同的方案。本文在全面研究云存储安全可证明性研究成果的基础上,对现有技术进行了分析和对比,并提出了进一步的研究方向。

## 1 云数据存储模型

云数据存储服务的架构如图 1 所示[2]。

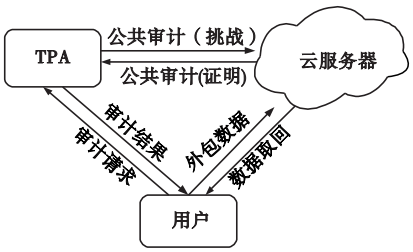


**图1　云数据存储服务架构**

在这个架构中一共有三个角色:用户、云服务器(cloud server)和第三方审计者(third party auditor,TPA)。其中,TPA 的作用是代表数据所有者完成数据的完整性认证和审计任务等,这样用户就不需要亲自去做这些事,这对云计算的经济规模化是有价值的。用户(可以是个人或企业)就是希望利用云服务器来存储自己的大量数据,从而节省了建立本地存储基础设施的费用。云服务器除了提供数据存储的服务,还可以提供可用性服务和分享服务。

本文关心的重点是如何保障云服务器中数据的完整性。由于数据所有者失去了对自己数据的直接物理控制,如何进行数据完整性的认证是十分关键的。考虑到自身的资源限制和云计算的经济规模化,数据所有者会把认证这项工作交给 TPA 来做,但同时又希望不会把数据隐私泄露给 TPA;同时,人们希望 TPA 做的认证工作是有效率的,能够尽量地缩减计算开销和存储开销,尽量减少数据所有者的在线负担,比如密钥或 MAC 的更新。

在本文中考虑的对手(adversary)包括外部入侵者和半信任的云服务器。外部入侵者有能力攻击云服务器,并且损坏其中的数据却不被发现;云服务器在多数情况下是不会破坏数据的,但是为了自身的利益,也可能删除服务器中长时间不用的数据,以此减轻负担和开支,也有可能发现数据被外部入侵者损坏,却对数据所有者隐瞒实情,以此来维护自己的名誉。

## 2　云存储安全可证明性的研究发展

目前,对云计算数据存储安全的研究主要提出了两种基本的技术来解决数据的可用性和完整性。一种是可检索证明(proofs of retrievability,POR);另一种是可证明数据拥有(provable data possession,PDP)。可证明的数据拥有性表示服务器可以给出拥有用户数据的证明,这个证明对数据完整性的保障有确定性和概率性两类,这里主要介绍概率性的一类;而可检索证明表示服务器可以给出数据可检索的证明,数据可检索的意思是:数据可以有一定程度的损坏,但是利用编码技术仍然可以恢复出原文件。在这两种方法的基础上,经过不断地完善和改进,人们又提出了一些基于云计算环境的方案。

图 2 展示了 POR 和 PDP 的发展历程。因为很多论文提出的算法并没有单独的名字,所以对于这些论文笔者直接以文献编号表示。

首先 Juels 等人[6]于 2007 年 10 月最先提出了 POR 系统的概念。简单地说,POR 方案就是档案或备份服务(prover)能生成一个简洁的证明,来证明用户(verifier)可以检索目标文件 F,并且不需要用户下载文件来完成验证。同年 10 月,Ateniese 等人[7]提出了 PDP 的概念,可以说与 POR 很类似,不过 PDP 只能检测出文件是否损坏,并不能保证文件是可检索的。至此,两种技术就各自走上了不同的发展道路。

2008 年 6 月,Curtmola 等人[8]把 PDP 系统拓展到了多服务器环境,通过做副本的方式提高了文件的可用性和完整性。同年 9 月,Ateniese 等人[9]改进了自己之前的方案,使用对称密钥并加入了文件更新的功能,为之后云计算环境下的数据动态化奠定了基础;12 月,Shacham 等人[10]改进了原始的 POR 模型,使用了同态认证(homomorphic authenticators)缩减了通信开销,并且询问次数是无限的。

2009 年 2 月,Dodis 等人[11]从理论上建立了一个 POR 模型,主要利用信息论的技术保障了服务器正确回复率低时的提取可能性。同年 11 月,Bowers 等人[12]从理论和实现两方面考虑,综合了的 POR 模型,提出了更有现实意义的 POR 模型;Erway 等人[13]在 ACM 上发表的文章提出了动态 PDP,进一步完善了 PDP 的数据动态化功能。同年 12 月,Bowers 等人[14]提出了一个 HAIL 模型,把 POR 系统拓展到了多服务器环境,获得了更好的效率和安全性,可以对抗 mobile adversary。图 3 展示了云数据存储的相关研究发展历程。

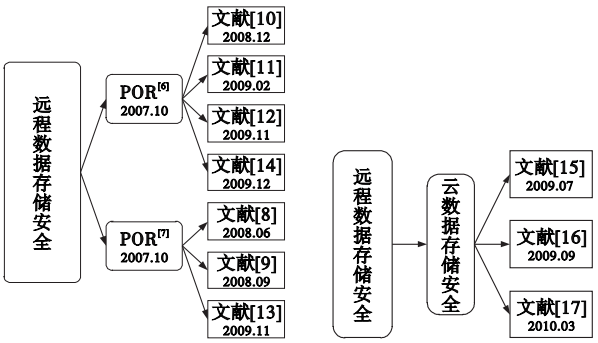


**图2　POR与PDP的发展历程　图3　云数据存储安全的发展历程**

2009 年 7 月,Wang 等人[15]发表了对云计算环境下的数据存储安全的研究成果,他们利用 homomorphic token 和擦除码数据认证这样一种有效灵活的分布式方案来确保云中用户的数据正确性。此方案成功整合了存储正确性的保障和数据的错误定位这两项功能。同年 9 月,Wang 等人[16]提出了包含公共审计的方案,在云计算环境下成功结合了公共认证和数据动态化这两个功能,其中利用梅克尔散列树(MHT)解决了云计

算环境下的数据动态化问题。2010年3月,Wang等人[17]又从用户隐私保护方面完善了之前的方案。

2010年9月,Hao Zhuo等人[18]在MR-PDP的基础上加入了公共审计的功能;Kumar等人[19]用sobol序列计算token的方法,提出了功能与本文相同的方案。

## 3 云数据存储安全协议研究综述

根据国内外的研究现状和发展趋势,可以从中总结出云数据存储安全协议所要满足的各种功能需求,这对提出更加完善的云数据存储安全协议是非常有意义的。下面就从公共认证、同态认证、数据隐私保护、数据动态化、批审计和服务器间冗余这六个方面来阐述。

其中前五项功能来源于单服务器协议,都是相对独立的功能,不过完全可以整合到云服务器的环境中来,也就是说在单服务器和云服务器环境下都适用。而最后一项功能不能在单服务器协议中使用,只适用于云服务器环境,是一种多服务器环境下的协议优化功能。

### 3.1 公共认证

公共认证指的概念是:一个系统中任何实体(不可信)都可以执行认证审计的任务,而不仅仅是数据所有者本身可以做这件事。要满足公共认证的要求,简单说就是要应用公钥密码学。而应用对称密码学的协议由于对数据加了密,这样公共机构就无法对数据进行审计了。

支持公共认证的研究工作主要有:Ateniese等人[7]提出的PDP模型使用的是RSA签名,Shacham等人[10]提出的Compact POR模型使用的是BLS签名;Erway等人[13]提出的DPDP方案使用的是基于rank的认证跳转列表;Wang等人[16]提出的公共认证方案使用了RSA和BLS两种签名。这些文献都是基于公钥密码系统的,因此是支持公共认证的。Bowers等人[14]提出的HAIL模型虽然是基于POR的,也对文件加了密,但是HAIL的完整性认证只与服务器间数据块的一致性有关,并不需要解密文件,所以HAIL也是支持公共认证的。

### 3.2 同态认证技术

在提出同态认证技术之前,一种简单的方法是使用MAC签名技术。在数据外包前,数据所有者先随机生成一组MAC密钥,用来对计算文件的MAC值,也就是文件的数字签名;然后把MAC密钥和签名发给TPA。TPA执行认证任务时,先发给云服务器一个密钥,让它计算一个文件的签名发给TPA进行比较。这样做的缺陷就是密钥数量是有限的,一旦用光,数据所有者要重新选择一组密钥,并且计算MAC值,然后再发给TPA[2]。这样数据所有者的在线负荷就增加了。

解决这个问题的方法就是使用同态认证技术。首先介绍一下同态认证标签(homomorphic verifiable tags,HVTs)的主要性质[7]:通过$T_{m_i}$和$T_{m_j}$可以计算出数据块$m_i + m_j$的标签$T_{m_i+m_j}$,其中$T_{m_i}$为数据块$m_i$的标签。

可以看到,多个数据块的同态认证标签是可以聚集成一个标签值的。

下面对同态认证技术的具体应用进行简单阐述:先把文件分成$n$块,分别对每个数据块$m_i$计算同态认证标签$\sigma_i$作为文件的元数据(而不是之前的MAC值),以此来验证数据的完整性。当进行认证任务的时候,数据所有者或者是TPA发给云服务器一对随机挑战值$\{(i,v_i)\}$,用来随机采样文件的数据块。其中,$i$是一个随机值,$v_i$是$i$对应的随机值,$I$是集合[1,$n$]的子集。这就是所谓的blockless认证,不需要访问所有的数据块就能进行认证。云服务器利用同态认证的良好性质,可以把被采样的数据块的元数据聚集为一个线性组合:

$$\mu = \sum_i v_i \times m_i \quad \sigma = \prod_i \sigma_i^{v_i} \tag{1}$$

然后把这两个值发给TPA进行认证(这是一个概率认证,不是绝对认证[7])。可以看到$\mu$和$\sigma$的大小与整个文件的大小是独立的,意味着通信开销是一个常量,而且挑战值是可以不断再生的,不会出现用完的情况,因此不会对数据所有者产生在线负荷。

### 3.3 数据隐私保护

使用同态认证技术的确可以带来很多好处:a)大幅削减了通信开销;b)不会产生在线负荷。但是也会有问题:考虑到服务器最后计算出来的$\mu$是一个线性组合,如果TPA收集了足够多的线性组合,就很有可能破解出里面的数据(相当于解一个线性方程组),造成数据所有者的数据泄密。在数据外包前加密是一种方法[6],但是不能完全解决这个问题,而是把问题转移到了密钥管理上面。

比较好的解决方法是Wang等人[17]提出的在同态认证的基础上应用随机掩饰(random masking),其核心内容也就是在线性组合后面加上一个随机掩饰值:

$$\mu' = \mu + r \times h(R) = \sum_i v_i \times m_i + r \times h(R) \tag{2}$$

这样,无论收集多少线性组合,因为有随机值的存在,是无法解线性方程组的。这个方法就相当于用随机掩饰保护了同态认证。可以看到,这个改变并没有以换取某些性能作为代价,通信开销仍然是一个常量,也不会产生在线负荷。

### 3.4 数据动态化

云服务器中的数据不一定都是静态数据,数据所有者也会有更新文件的需求。因此数据动态化是设计云数据存储安全协议所要考虑的一个非常重要的功能。完全的数据动态化包括数据块的修改、删除、增添和插入。

数据动态化主要面临的一个难题是数据块下标的变化,下标一旦变化,数据块的同态认证标签也将随之发生变化,必须得重新计算。这点在数据块插入上体现得尤为明显,比如说在数据块$m_i$后面插入一个数据块,那么这个数据块之后的所有标签都要重新计算,这会带来非常大的计算负担。解决这个问题的关键是同态认证标签的计算不会跟随下标的变化而变化。

一种比较好的解决方法是Wang等人[16]提出的方法。该方法使用由Merkle于1980年提出梅克尔散列树(Merkle hash tree,MHT)。可以把数据块$m_i$看做散列树的叶子,利用对应的附加认证信息$\Omega_i$(auxiliary authentication information,AAI)以及文件的根值$\mathcal{R}$就可以完成认证任务。直接使用MHT会带来通信开销过大和数据隐私的问题,所以要把MHT与同态认证相结合。可以预见的是,随机掩饰一样可以加入进来,从而解决数据隐私问题。

### 3.5 批审计

当有多个数据所有者需要TPA执行审计任务的时候,批审计的设想是很自然的。因为批审计的效率要大于依次执行

单个审计任务的效率。执行批审计任务的关键就是要有能力把不同文件的签名聚集成一个值,而双线性聚集签名是可以满足这个要求的,文献[10,16]使用的正是这种签名技术,因此可以直接应用于批审计。文献[17]详细地研究了批审计的数学细节。为了防止出现单个审计任务认证失败而导致整个批审计的的失败,一种分割制服递归法(recursive divide-and-conquer)可以把服务器的回复分成两部分,清理出失效的回复,然后用对半递归的方式完成审计。实验证明,即使有 20% 的回复失效,批审计的速度仍然比单个审计的速度快[17]。

### 3.6　服务器间冗余

POR 与 PDP 的研究最初都是建立在单个服务器基础上的,而云计算环境是海量的服务器。虽然 POR 和 PDP 都可以应用于云服务器环境,但是在云服务器环境下是否会发生一些变化,能否利用多服务器带来的优势是必须考虑的问题。笔者发现在多服务器环境下,通过服务器之间的文件冗余可以提高效率和文件的完整性保障,而单服务器在这方面是有缺陷的。Wang 等人[15~17]提出的三种方案虽然是在云计算环境下,但是并没有利用多服务器的优势,基本上还是应用的单服务器环境下的协议。文献[14]提出的 HAIL 系统便是将 POR 模型比较完整地拓展到了多服务器环境,应用了多个创新的编码方法。作者认为 POR 和 PDP 是设计过度的(overengineered),因为都要服务器存储检测值。而像 HAIL,检测值是从服务器组中得到的。另一方面,传统的分布式协议是设计不足的(underengineered),缺乏强健的检测和再分配。

HAIL 中传播码(dispersal code)的应用可以把服务器的存储开销从 $n|F|$ 减少到 $(n/l)|F|$,其中 $n$ 为服务器数量,$l$ 为文件段的数量(也是主服务器的数量),$(n,l)$ 是传播码参数。

首先阐述传播码的原型。该原型叫做完整性保护纠错码(integrity-protected error-correcting code,IP-ECC)。这个 IP-ECC 结构将伪随机函数、纠错码和通用散列函数结合到一起,成为一个密码原型。这个原型就是一个纠错码,同时也是一个抗损坏的消息 MAC。

概括地说,IP-ECC 的性质是基于 UHF 族 $h$ 的三个性质:a)$h$ 是线性的,如 $h_\kappa(m)+h_\kappa(m')=h_\kappa(m+m')$,其中 $m$ 和 $m'$ 是消息,$\kappa$ 是密钥;b)对于伪随机函数 $g$,函数 $h_\kappa(m)+g_{\kappa'}(m)$ 是消息的 MAC 值;c)$h_\kappa(m)$ 可以被看做对消息进行纠错码纠错后产生的冗余块。

除了传播码,服务器码也是不可或缺的。服务器码是对每个服务器中的文件块进行编码的纠错码。服务器码可以抵御文件块的小规模损坏,这种损坏多数发生在认证失败的时候。

具体做法如下:

a)把文件分成 $l$ 段,然后把文件段 $F(j)$ 发给主服务器 $j$(primary servers),其中 $j=[1,l]$。得到一个矩阵:

$$\{F_{ij}\}_{i=[1,m_F],j=[1,l]} \tag{3}$$

其中,文件段数量 $m_F=|F|/l$。

b)用服务器码对服务器 $j$ 上的文件段 $F(j)$ 编码,得到长度为 $m$ 的文件段(其中 $m_F+1,\cdots,m$ 为 parity blocks(奇偶块))。

c)用传播码 $\mathrm{ECC}_d$ 对 a)得到的矩阵的行进行编码,得到了 $F(l+1),\cdots,F(n)$。

最后,把编码完的整个矩阵定义为

$$F_d=\{F_{ij}^l\}_{i=[1,m],j=[1,n]} \tag{4}$$

其中,$\{F_{ij}^l\}_{i=[1,m_F],j=[1,l]}$ 等于原文件 $\{F_{ij}\}$。$\{F_{ij}^l\}_{i=[m_F+1,m],j=[1,l]}$ 是用服务器码编码产生的冗余块。剩余的冗余块生成方法如下:

$$F_{ij}^l=\text{RS-UHF}_{\kappa_j}(F_{i1}\cdots F_{il})+g_{\kappa_j'}(\tau_{ij}) \tag{5}$$

其中:$i=[1,m]$;$j=[l+1,n]$;$\kappa_j$、$\kappa'_j$ 分别是 UHF 和 PRF 的密钥;$\tau_{ij}$ 是文件句柄的位置标记。函数 RS-UHF 的具体定义如文献[14]。

一个完整编码过程的例子如图 4 所示。

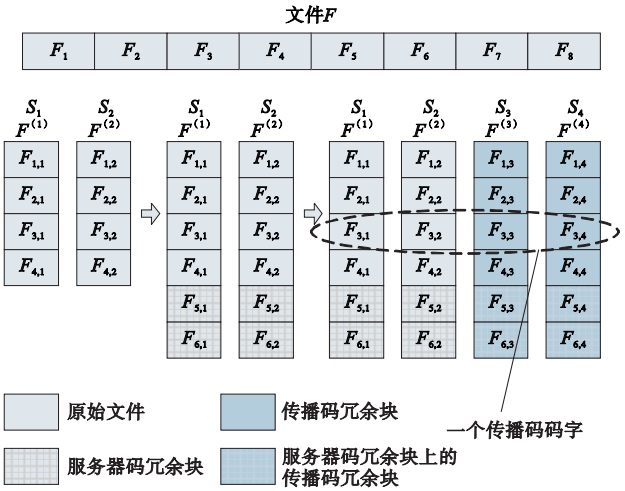


**图4　文件分段**

一般来说,在多个服务器上制作多副本可以提高数据的完整性保障,但是单纯地复制文件会增加服务器的存储开销。而应用传播码编码技术可以显著地缩减开销,同时这些跨服务器冗余让受损文件的恢复成为可能。

## 4　各种方案的功能与开销比较

表 1 从公共认证、无在线负荷、数据隐私保护、数据动态化、批审计和服务器间冗余六个方面对九篇文献进行了比较。

表 1　各种方案的功能比较

| 比较项 | 公共认证 | 无在线负荷 | 隐私保护 |
|---|---|---|---|
| POR | × | × | √@ |
| PDP | √ | √1 | × |
| Scalable PDP | × | × | √ |
| Compact POR | √ | √2 | × |
| DPDP | † | √3 | × |
| HAIL | √ | √ | × |
| 文献[15] | × | × | × |
| 文献[16] | √ | √1,2 | × |
| 文献[17] | √ | √2 | √ |

| 比较项 | 数据动态化 | 批审计 | 服务器间冗余 |
|---|---|---|---|
| POR | × | × | × |
| PDP | × | × | × |
| Scalable PDP | √#% | × | × |
| Compact POR | × | √ | × |
| DPDP | √ | × | × |
| HAIL | × | × | √ |
| 文献[15] | √# | × | √& |
| 文献[16] | √ | √ | √& |
| 文献[17] | √ | √ | √& |

表 1 中,1 为基于 RSA 结构的同态认证;2 为基于 BLS 签名的同态认证;3 为基于 rank 的认证跳转列表;@ 为此方案的数据是加密的;#为不包含 insert 操作;% 为操作次数是有限的(预先给定的);& 为没利用多服务器的优势;† 为文中没有明确实现。

在比较算法孰优孰劣时,算法的效率是至关重要的,而一个算法的效率主要从时间复杂度和空间复杂度来考虑,计算开销相当于时间复杂度,而通信开销和存储开销均相当于空间复杂度。表 2 是几种比较典型的方案的开销比较。

表 2 各种方案的开销比较

| 比较项 | 服务器计算开销 | 用户端计算开销 | 通信开销 | 用户端存储开销 |
|---|---|---|---|---|
| PDP | $O(1)$ | $O(1)$ | $O(1)$ | $O(1)$ |
| Scalable PDP | $O(1)$ | $O(1)$ | $O(1)$ | $O(1)$ |
| Compact POR | $O(1)$ | $O(1)$ | $O(1)$ | $O(1)$ |
| DPDP | $O(\log n)$ | $O(\log n)$ | $O(\log n)$ | $O(1)$ |
| 文献[16] | $O(\log n)$ | $O(\log n)$ | $O(\log n)$ | $O(1)$ |

由表 2 可以看到,随着功能的不断升级,还是要付出开销增大的代价的。不过 $O(\log n)$ 的开销还是可以接受的。如果到了 $O(n)$ 的话,开销就比较大了。

## 5 一个比较完备的协议框架

在满足公共认证、同态认证、数据动态化、隐私保护、批审计和服务器间冗余等功能的基础上,本文把 POR 系统中的文件提取环节整合进来,以进一步完善云数据存储安全协议。其中文件提取的框架可分为两步:

a)用户检测服务器的文件数据损坏率是否小于 $\xi$(这个服务器就称为 $\xi$-adversary)。文件 $F$ 首先通过服务器码编码,并且拆分给主服务器,生成 $F_{serv}$;然后再用传播码编码,生成 $F_{out}$,这样服务器间的冗余就完成了。之后用户发送一组挑战,服务器作出回复。用户通过回复的正确率来判断服务器的文件数据损坏率是否小于 $\xi$。如果小于 $\xi$,用户接下来就可以提取数据了。

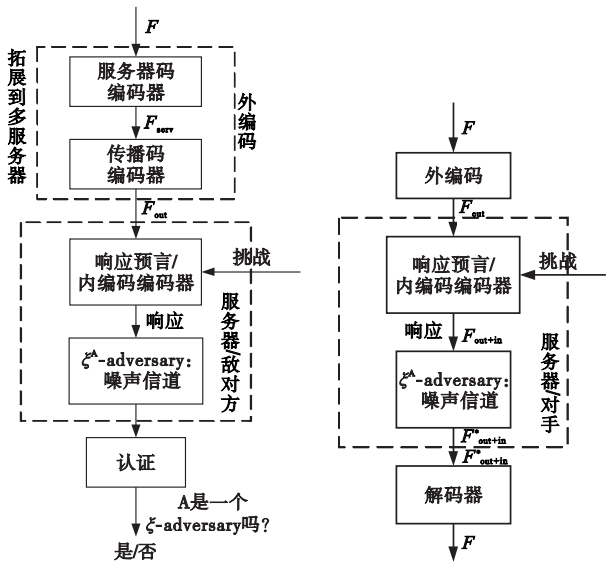


图5 一个比较完备的协议框架

b)用户从损坏率小于 $\xi$ 的服务器中提取数据。用户发送挑战后,服务器使用内编码对被采样的数据块编码,生成 $F_{out+in}$;然后通过噪声通道,相当于受到一定程度的损坏,生成 $F^*_{out+in}$;最后用户对受损文件解码恢复出原文件 $F$。

在这个数据提取框架的基础上,本文在认证阶段使用的是公共认证、同态认证和随机掩饰这三种技术,同时再把数据动态化和批审计这两项功能加入到这个框架[16,17]。

根据文件是静态数据还是动态数据来决定是否使用数据动态化功能。对于静态数据来说,一般文件都是非常巨大的,用户不会随时修改文件内容,所以数据动态化是没有必要应用的。对于动态数据来说,文件一般比较小,用户会频繁地对文件进行修改,所以数据动态化是必要的。而对于编码后的文件是无法实现数据动态化的,因为一旦原文件数据块发生变化,那么编码之后的所有冗余块都会发生变化,都必须重新计算,这样会产生大量的计算开销。所以对于动态数据来说,没必要应用服务器间的编码保障,这样可以高效地实现数据动态化。

如果遇到多个用户同时对 TPA 发出审计请求,批审计功能的应用可以提高审计效率。

## 6 结束语

作为把计算变为公共设施的梦想,人们已经认为云计算是 IT 企业的新一代结构。在云计算中,用户可以在云中远程存储自己的数据,从而享受到高质量的应用和服务,而且是按需提供的。通过数据外包,用户可以减少本地的存储和维护压力。但是这样用户就失去了对数据的直接物理控制,传统的密码原型就不能直接应用。于是,云计算中的数据完整性保护就成为了非常具有挑战性的任务,尤其是当数据文件需要不断更新的时候。

本文分析了国外对于这项任务的一些科研成果,从功能和需求上对云数据存储安全协议进行了讨论,并且对多篇文献的方案进行了功能和开销比较。可以看到在云计算环境下,公共认证或者说公共审计是协议设计中必须考虑的一点。TPA 的存在可以对大量用户的审计需求进行管理和批处理。数据的隐私保护是十分重要的,因为 TPA 也不是完全可信的。当用户的文件需要不断更新,协议的数据动态化也是必须满足的一项功能。数据隐私也是不可忽视的。

云计算环境下面临的挑战还很多,以上这些工作都是基于服务器不可信而用户可信的情况,当用户本身不可信的时候又应该如何做呢。在多服务器环境下如何更好地拓展单服务器的各种方案也是值得研究的。还有性能的改进也是有前景的,面对海量用户和服务器,开销尽可能地缩减是很有价值的。

**参考文献:**

[1] MELL P, GRANCE T. NIST SD 800-145, The NIST difinition of cloud computing[S]. Gaithersburg, MD: NIST Special Publication, 2011.

[2] WANG Cong, REN Kui, LOU Wen-jing, *et al*. Toward publicly auditable secure cloud data storage services[J]. IEEE Network, 2010, 24(4): 19-24.

[3] Amazon. com. Amazon S3 availability event[EB/OL]. [2008-07-20]. http://status. aws. amazon. com/s3-20080720. html.

[4] ARRINGTON M. Gmail disaster: reports of mass email deletions [EB/OL]. [2006-12-28]. http://www. techcrunch. com/2006/

12/28/gmail-disaster-reports-of-mass-email-deletions/.

[5] KRIGSMAN M. Apple's mobileme experiences post-launch pain [EB/OL]. [2008-07-11]. http://blogs.zdnet.com/projectfailures/?p=908.

[6] JUELS A, Jr KALISKI B S. PORs: proofs of retrievability for large files[C]//Proc of the 14th ACM Conference on Computer and Communications Security. New York: ACM Press, 2007: 584-597.

[7] ATENIESE G, BURNS R, CURTMOLA R, *et al*. Provable data possession at untrusted stores[C]//Proc of the 14th ACM Conference on Computer and Communications Security. New York: ACM Press, 2007: 598-609.

[8] CURTMOLA R, KHAN O, BURNS R, *et al*. MR-PDP: Multiple-replica provable data possession [C]//Proc of the 28th International Conference on Distributed Computing Systems. Washington DC: IEEE Computer Society, 2008: 411-420.

[9] ATENIESE G, PIETRO D R, MANCINI L V, *et al*. Scalable and efficient provable data possession [C]//Proc of the 4th International Conference on Security and Privacy in Communication Networks. New Yofk: ACM Press, 2008: 9.

[10] SHACHAM H. WATERS B. Compact proofs of retrievability[C]// Lecture Notes in Computer Science, vol 5350. Berlin: Springer, 2008: 90-107.

[11] DODIS Y, VADHAN S, WICHS D. Proofs of retrievability via hardness amplification[C]//Proc of the 6th theory of Cryptography Conference on Theory of Cryptography. Berlin: Springer-Verlag, 2009: 109-127.

[12] BOWERS K D, JUELS A, OPREA A. Proofs of retrievability: theory and implementation[C]//Proc of ACM Workshop on Cloud Computing Security. New York: ACM Press, 2009: 43-53.

[13] ERWAY C, KÜPCÜ, PAPAMANTHOU C, *et al*. Dynamic provable data possession[C]//Proc of the 16th ACM Conference on Computer and Communications Security. New York: ACM Press, 2009: 213-222.

[14] BOWERS K D, JUELS A, OPREA A. HAIL: a high-availability and integrity layer for cloud storage[C]//Proc of the 16th ACM Conference on Computer and Communications Security. New York: ACM Press, 2009: 187-198.

[15] WANG Cong, WANG Qian, REN Kui, *et al*. Ensuring data storage security in cloud computing [C]//Proc of the 17th International Workshop on Quality of Service. 2009: 1-9.

[16] WANG Qian, WANG C, LI Jin, *et al*. Enabling public verifiability and data dynamics for storage security in cloud computing[C]//Lecture Notes in Computer Science, vol 5789. Berlin: Springer, 2009: 355-370.

[17] WANG Cong, WANG Qian, REN Kui, *et al*. Privacy-preserving public auditing for data storage security in cloud computing[C]// Proc of IEEE INFOCOM. 2010: 1-9.

[18] HAO Zhuo, YU Neng-hai. A multiple-replica remote data possession checking protocol with public verifiability[C]//Proc of the 2nd International Symposium on Data, Privacy and E-Commerce. Washington DC: IEEE Computer Society, 2010: 84-89.

[19] KUMAR P S, SUBRAMANIAN R, SELVAM D T. Ensuring data storage security in cloud computing using Sobol sequence[C]//Proc of the 1st International Conference on Parallel Distributed and Grid Computing. Washington DC: IEEE Computer Society, 2010: 217-222.

(上接第 2415 页)

[19] NGUYEN H V, TRAN F D, MENAUD J M. Performance and power management for cloud infrastructures[C]//Proc of the 3rd IEEE International Conference on Cloud Computing. [S. l.]: IEEE Press, 2010: 329-336.

[20] HYSER C, MCKEE B, GARDNER R, *et al*. HP Labs technical report, Autonomic virtual machine placement in the data center[EB/OL]. (2007). http://www.hpl.hp.com/techreports/2007/HPL-2007-189.html.

[21] BOBROFF N, KOCHUT A, BEATY K. Dynamic placement of virtual machines for managing SLA violations[C]//Proc of the 10th IEEE Symposium on Integrated Management. Wahington DC: IEEE Computer Society, 2007: 119-128.

[22] MACHIDA F, KIM D S, PARK J S, *et al*. Toward optimal virtual machine placement and rejuvenation scheduling in a virtualized data center[C]//Proc of IEEE International Conference on Software Reliability Engineering Workshops. 2008: 1-3.

[23] MACHIDA F, KAWATO M, MAENO Y. Redundant virtual machine placement for fault-tolerant consolidated server clusters[C]//Proc of Network Operations and Management Symposium. 2010: 32-39.

[24] 黄昌勤,李翠菊,宋广华,等. 计算网格中的任务管理研究及示范应用[M]. 北京:科学出版社,2009.

[25] 刘宴兵,尚明生,肖云鹏. 网格高性能调度及资源管理技术[M]. 北京:科学出版社,2010.

[26] BUYYA R, MURSHED M. GridSim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing[J]. Concurrency and Computation: Practice and Experience, 2002, 14(13-15): 1175-1220.

[27] BUYYA R, MURSHED M, ABRAMSON D. A deadline and budget constrained cost-time optimization algorithm for scheduling task farming applications on global grids[C]//Proc of International Conference on Parallel and Distributed Processing Techniques and Applications. 2002: 540-552.

[28] 刘晓茜. 云计算数据中心结构及其调度机制研究[D]. 合肥:中国科学技术大学,2011.

[29] WOOD T, SHENOY P, VENKATARAMANI A, *et al*. Black-box and gray-box strategies for virtual machine migration[C]//Proc of the 4th USENIX Symposium on Networked Systems Design & Implementation. 2007: 229-242.

[30] ZHOU Wen-yu, YANG Shou-bao, FANG Jun, *et al*. VMCTune: a load balancing scheme for virtual machine cluster based on dynamic resource allocation[C]//Proc of the 9th International Conference on Grid and Cloud Computing. 2010: 81-86.

[31] 刘鹏程,陈榕. 面向云计算的虚拟机动态迁移框架[J]. 计算机工程,2010, 36(5): 37-39.

[32] 张彬彬,罗英伟,汪小林,等. 虚拟机金系统在线迁移[J]. 电子学报,2009, 37(4): 894-899.

[33] 周文煜,陈华平,杨寿保,等. 基于虚拟机迁移的虚拟机集群资源调度[J]. 华中科技大学学报, 2011, 39(sup1): 130-133.