DOI:10.3969/j.issn.1000-1565.2021.05.017

云存储环境下数据持有性证明研究综述

田俊峰,宋倩倩,王浩宁
(河北大学 网络空间安全与计算机学院,河北 保定 071002)

田俊峰 河北大学教授,博士生导师,现任网络空间安全与计算机学院院长,河北省高可信信息系统重点实验室主任.1986 年毕业于河北大学电子系,2004 年博士毕业于中国科技大学计算机科学与技术专业.河北省网络安全学会理事长,《通信学报》《网络与信息安全学报》《信息安全研究》编委,中国计算机学会分布式计算与系统委员会委员.先后被评为河北省具有突出贡献的中青年专家,河北省具有突出贡献的中青年教师,河北省"三三三人才工程"人选,2016 年度宝钢奖优秀教师.多年来一直从事分布计算、网络安全与可信计算等方向的教学和科研工作.主持国家自然科学基金、河北省杰出青年科学基金、河北省自然科学基金、军地委托开发项目等基金项目 30 余项.在国内外重要学术期刊和会议上发表学术论文 80 余篇,主编出版专著 2 部,课程教材 3 部.获河北省科技进步奖 4 项,军队科技进步奖 1 项,河北省优秀教学成果二等奖 1 项.

摘 要:随着网络存储技术的发展,越来越多的用户选择将数据存储到云端,从而用户失去对数据的直接控制.如何验证存储在云端的数据完整性便成为用户最关心的问题之一.学术界和工业界普遍认为数据持有性证明(provable data possession,PDP)机制是解决该问题的重要手段,本文对现有的部分经典数据持有性证明方案进行了梳理.给出了数据持有性证明系统模型和审计框架;分析了数据持有性证明系统的功能特性和安全需求;分别从实现原理、应用场景和不同实体 3 个不同的视角,对目前主要的数据持有性证明方案进行了总结归纳,并对未来的研究趋势进行了展望.

关键词:云存储;数据完整性;数据持有性证明
中图分类号:TP301 **文献标志码**:A **文章编号**:1000-1565(2021)05-0599-13

A review of proof of data possession in cloud storage environment

TIAN Junfeng, SONG Qianqian, WANG Haoning
(School of Cyberspace Security and Computer, Hebei University, Baoding 071002, China)

Abstract: With the development of network storage technology, more and more users choose to store data in the cloud, thus users lose direct control of the data. How to verify the integrity of the data stored in the cloud becomes one of the most concerned issue. Proof of Data Possession (PDP) mechanism is considered to be an important means to solve this problem in academia and industry. This paper discusses

收稿日期:2021-04-28
基金项目:河北省自然科学基金重点项目(F2016201244)
第一作者:田俊峰(1965—),男,河北蠡县人,河北大学教授,博士生导师,主要从事可信计算和信息安全方向研究.
E-mail:tjf@hbu.cn
通信作者:宋倩倩(1996—),女,河北邯郸人,河北大学在读硕士研究生,主要从事信息安全方向研究.
E-mail:songqianhbu@163.com

some of the existing major PDP schemes. Firstly, this paper describes the data possession attestation system model and audit framework, analyzes the functional characteristics and security requirements of the data possession attestation system. Secondly, from the three different perspectives of implementation principles, application scenarios and different entities, the current main data ownership certification schemes have been carried out. Finally, this paper introduces the future research trends in this field.

Key words: cloud storage; data integrity; proof of data possession

云存储是从云计算衍生而出的概念,是利用云计算技术和架构来为用户提供按需付费的存储服务,为用户减少了管理数据、硬件购置和维护的费用.随着网络的发展和人们生活方式的改变,现实生活的数据规模不断增大,越来越多的用户选择将自己的数据存储到云中.然而,云存储系统给人们带来便利的同时也伴随着安全的挑战.例如:2017年3月,京东内部人员与黑客长期勾结贩卖公民信息,泄露近50亿条公民信息[1];2017年9月,美国信用机构Equifax遭到黑客攻击,导致高达1.43亿客户的个人信息泄露[2];2018年7月,腾讯云由于物理硬盘出现故障,给创业公司"前沿数控"带来了毁灭性的打击等事件[3].这些安全问题影响了公众对云存储的信心,从而阻碍了其进一步的应用和发展.

当用户将数据存储到云服务器中,半可信的云服务提供商可能未经用户授权的情况下对存储在云中的数据进行修改和删除等操作.近来,威胁云数据安全的事件频繁发生,现将威胁云数据安全的原因总结如下:1)服务器发生故障,这种故障虽属于小概率事件,一旦发生,会给用户造成巨大的伤害;2)存储在云中的数据可能遭到黑客或者其他用户的窃取,导致用户信息泄露;3)云服务提供商为了经济利益,可能删除一些用户不常访问的数据.显然,云数据的安全问题已经成为亟待解决的问题.为此,数据持有性证明机制的研究越来越受到学术界和工业界的重视.

近些年来,为了验证用户存储在云中数据的完整性,越来越多的数据持有性证明方案被相继提出,进而满足不同用户的实际需求.本文针对目前的发展现状,探讨了数据持有性证明机制的系统模型、审计框架、功能特性和安全需求,并基于数据持有性证明的主要方案进行了归纳,分析了存在的问题并对其未来研究趋势进行了展望.

1 系统模型和审计框架

本节首先对云存储的系统模型和审计框架进行了介绍,随后对数据持有性证明机制的功能特性和安全需求进行了简单论述.

1.1 系统模型

云存储环境下数据持有性验证机制分为私有数据持有性验证机制和公开数据持有性验证机制,对应了不同的数据存储模型.

1) 私有的数据持有性验证机制包括2个实体:云服务提供商(cloud server provider,CSP)和用户(user).如图1所示.

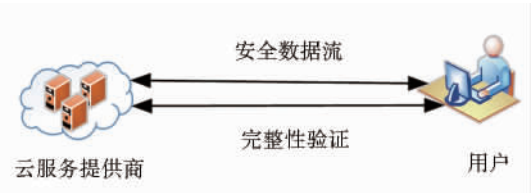


图1 私有数据完整性验证机制

Fig.1 Private data integrity verification mechanism

云服务提供商:提供弹性伸缩服务的实体,为用户提供按需付费的服务,拥有海量的存储空间和较强的计算能力.在图1中,云服务提供商根据用户的需求提供相关资源,为其提供存储、管理和共享等服务.

用户:数据持有者,可以是个人或者公司机构.由于存储空间有限,他们需将其大量数据外包给云服务提供商.在图 1 中,用户需要将其数据外包给云服务提供商,并需要对存储在云端的数据进行完整性验证.

在图 1 中,用户和云服务提供商建立通信产生数据流,并将其数据存储到云端.为了验证数据是否完整地存储在云端,用户执行完整性验证算法.然而,用户可能不会投入大量的精力去随时随地充当验证者,进而引入了第三方审计者(third party auditor),为此有了公开数据持有性证明机制.

2) 公开的数据持有性验证机制包括 3 个实体:用户、云服务提供商和第三方审计者.如图 2 所示.

第三方审计者:第三方认证机构,通常由政府和可信机构担当,具备专业的验证知识和丰富的经验,为用户和云服务提供商提供令人信服的结果.

在图 2 中,用户将数据上传至云端,同时委托第三方审计者充当验证者向云服务提供商发起数据完整性挑战,云服务提供商将证据发送给第三方审计者,第三方审计者执行数据完整性验证,并将审计结果告知用户.

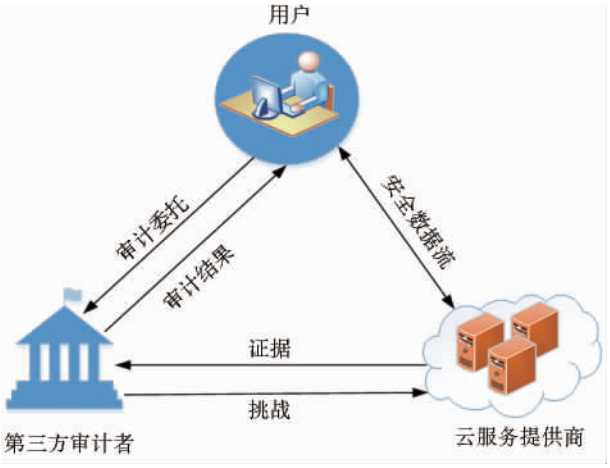


图 2　公开数据完整性验证机制

Fig.2　Public data integrity verification mechanism

1.2　审计框架

数据持有性审计框架实质是基于“挑战-响应”协议的数据完整性验证,通过随机取样的方法来验证存储在云中数据的完整性,根据部分数据的完整性概率来推测整个数据的完整性.详细审计步骤由 *Setup* 和 *Verify* 2 个阶段组成.*Setup* 阶段由密钥生成算法和数据块标签生成算法组成,用于完成系统设置的准备工作.*Verify* 阶段由挑战生成算法、证据生成算法和证据验证算法组成,用于完成数据完整性的验证.方案具体执行过程如图 3 所示.

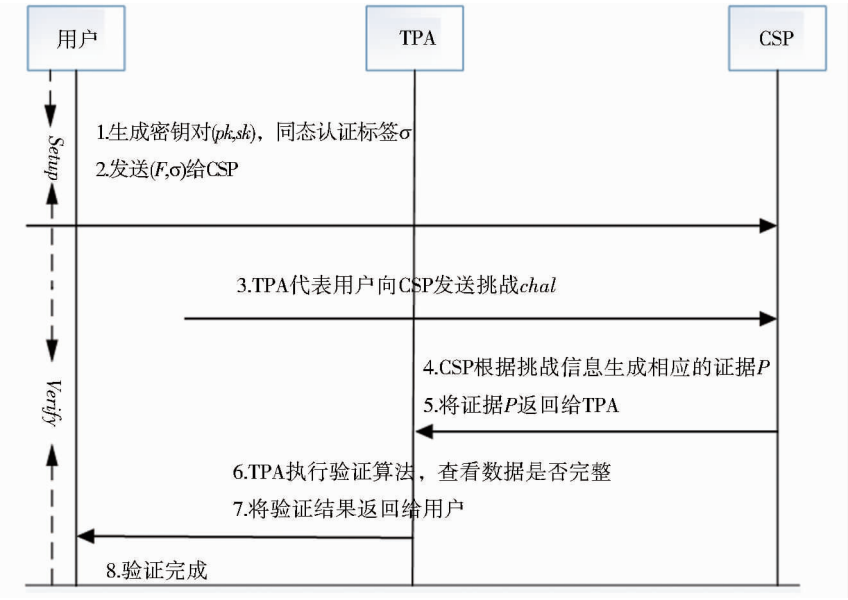


图 3　数据持有性证明机制示意

Fig.3　Schematic diagram of data possession proof mechanism

1) *Setup* 阶段

密钥生成算法:$KeyGen(1^k)\rightarrow(pk,sk)$.该算法由用户在本地执行,输入系统安全参数 k,输出密钥对(pk,sk).

标签生成算法:$TagGen(sk,F)\rightarrow\delta$.该算法由用户执行,输入私钥 sk 和数据文件 F,为每个数据文件 F 生成同态认证标签集合 δ,即输出认证元数据集 δ.

2)*Verify* 阶段

挑战生成算法:$ChalGen(c)\rightarrow chal$.该算法由用户或者第三方审计者执行,输入参数 c,生成相关挑战信息 $chal$.

证据生成算法:$GenProof(pk,F,\delta,chal)\rightarrow P(D,T)$.该算法由云服务提供商运行,输入参数包括公钥 pk、数据文件 F、元数据集 δ 和挑战请求 $chal$,输出本次挑战请求的完整性证据 $P(D,T)$.

证据验证算法:$VerifyProof(pk,chal,P)\rightarrow\{\text{“}true\text{”},\text{“}false\text{”}\}$.该算法由用户或者第三方审计者运行,输入参数公钥 pk、挑战请求 $chal$ 和证据 P,对云服务提供商返回的证据进行验证,输出验证成功或失败.

支持动态的 PDP 方案还需要以下 2 个算法支持.

更新执行算法:$ExecUpdate(F,\delta,Update)\rightarrow\{F',\delta',V_{update}\}$.该算法由云服务提供商运行,输入文件 F,相应标签集合 δ 和更新数据操作 $Update$,输出更新文件 F'和更新标签集合 δ'以及相对应的更新证据 V_{update}.

更新验证算法:$VerifyUpdate(pk,Update,P_{update})\rightarrow\{\text{“}true\text{”},\text{“}false\text{”}\}$.该算法由用户或第三方审计者执行,返回更新操作成功或失败.

1.3 功能特性

本节对数据持有性证明方案的基本功能特性进行了介绍.

1) 支持无块验证:用户或第三方审计者无需从云端下载原数据,验证数据的完整性.

2) 支持动态操作:用户可以随时对存储在云中的数据进行删除、修改和插入操作,数据可以保持持续更新状态.

3) 支持隐私保护:第三方审计人员无法从云服务提供商端获取用户的任何相关信息.

4) 支持轻量高效性:在数据持有性证明方案中要尽可能地减少用户生成文件标签和验证过程中产生的计算和通信开销.

5) 支持批量审计:第三方审计者可以同时审计多个用户的多个文件.

1.4 安全需求

为了保证验证方案是安全的,数据持有性证明方案必须充分考虑其安全需求.安全需求主要包括:伪造攻击、替换攻击和重放攻击.

伪造攻击:云服务提供商为了通过验证者的验证,通常伪造用户数据块的标签生成证据.

替换攻击:云服务提供商为了通过验证者的验证,使用其他可用且未损坏的数据块和对应的标签来代替被挑战的数据块和标签.

重放攻击:云服务提供商为了通过验证者的验证,使用之前验证通过的证据返回给用户.

数据持有性证明方案必须抵御上述攻击才可保证该验证方案的安全性.

2 数据持有性证明方案

数据持有性证明机制主要是用于验证云中数据的完整性.本节对主要的数据持有性证明方案进行了归纳和分析.

2.1 基于实现原理的 PDP 验证机制

基于实现原理的 PDP 验证机制分为基于 HMAC 函数的验证机制、基于 RSA 签名的验证机制、基于 BLS 签名的验证机制、基于身份的验证机制和基于代数签名的验证机制.如图 4 所示.

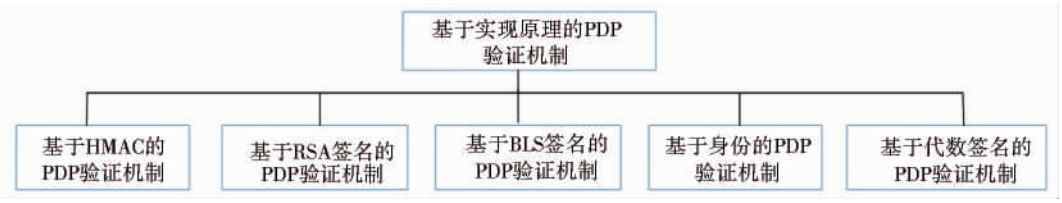


图 4 基于实现原理的 PDP 验证机制

Fig.4 PDP verification mechanism based on realization principle

最初始的数据持有性证明机制是 Deswarte 等[4]首次利用 HMAC 函数构造消息认证码,用以验证数据的完整性.用户将数据上传到云服务器前,预先计算数据的 MAC 值,并保存在本地.验证时,用户需要从云服务提供商中下载原数据,并计算其 MAC 值,与保存在本地的 MAC 值对比验证,判断存储在云端的数据是否是完整的.由于验证过程中需要下载整个原数据且验证次数有限,给用户带来了巨大的计算和通信开销.

为此,Deswarte 等[4]利用 RSA 签名机制的同态特性来验证数据的完整性,该机制可以进行无限次验证,但针对较大的数据,计算开销还是很大.数据持有性证明这个概念是由 Ateniese 等[5]提出的,2007 年,Ateniese 等[2]最先对数据持有性证明方案进行了形式化建模,提出了对数据文件进行分块的思想,降低了标签生成的代价,并利用同态认证标签将多个数据的标签聚合成一个值,有效地减少了计算和通信开销.验证时,采取随机抽样的方法对云中数据进行完整性验证,通过对部分数据块的检测来推测整个数据的完整性.Ateniese 等[5]提出的基于 RSA 的 PDP 验证机制示意如图 5 所示.

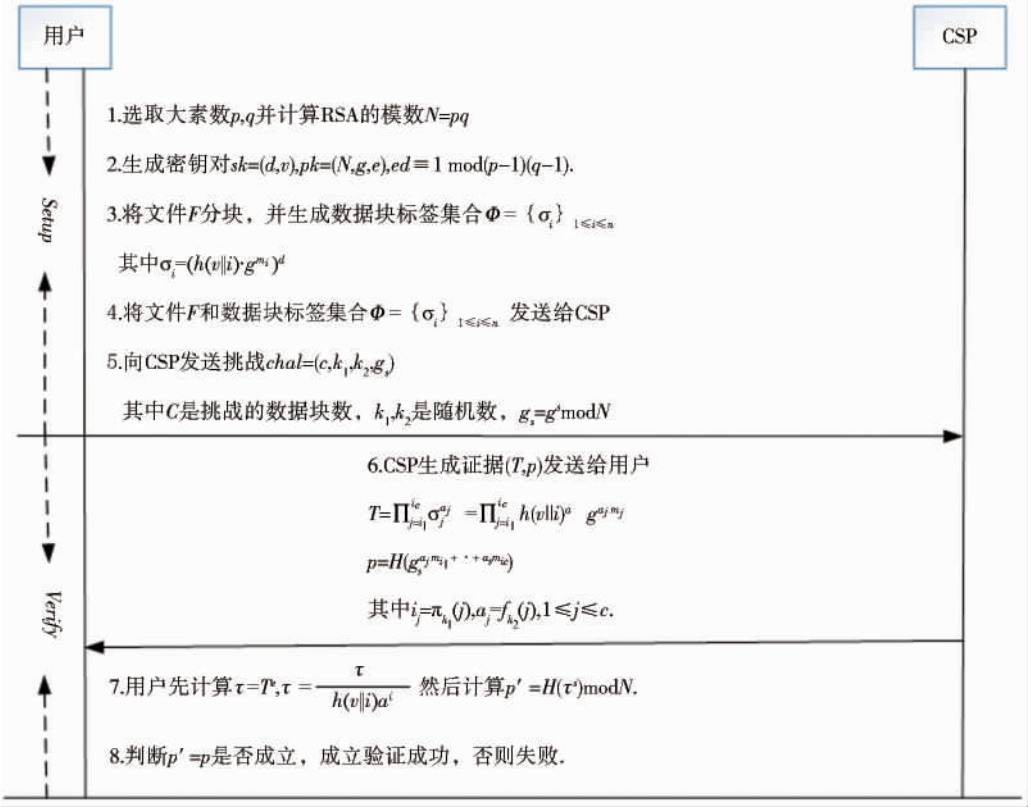


图 5　基于 RSA 的 PDP 验证机制

Fig.5　PDP authentication mechanism based on RSA

Boneh 等[6]提出的 BLS 签名机制是一种具有同态特性的短消息签名机制,在同等安全条件下(模数的位数为 1 024 bits),RSA 的签名位数是 1 024 bits,而 BLS 的签名位数大约为 160 bits,因此,BLS 签名是比 RSA 签名更短的签名机制.此外,BLS 签名机制具有同态特性,可以将多个数据块的值聚合成一个值,因此基于 BLS 的 PDP 验证机制有效地降低了存储和通信开销,且基于 BLS 的 PDP 验证机制支持公开验证,用户将审计任务委托给第三方审计,由第三方审计者代替用户完成审计工作,进而减轻了审计负担.基于 BLS 的数据持有性证明机制示意如图 6 所示.

上述的 PDP 验证机制是基于公钥基础设施(public key infrastructure,PKI)的验证机制,需要耗费资源来管理和维护证书.为了减少繁杂的证据管理工作,2006 年,GENTRY 等[7]提出了基于身份的聚合签名,使得验证的总消息最短. Zhao 等[8]基于身份的集合签名提出了第一个基于身份的公共验证方案,该方案只有私钥生成器(private key generator,PKG)拥有传统的公钥,用户只是保留其身份而不与证书绑定,简化了密

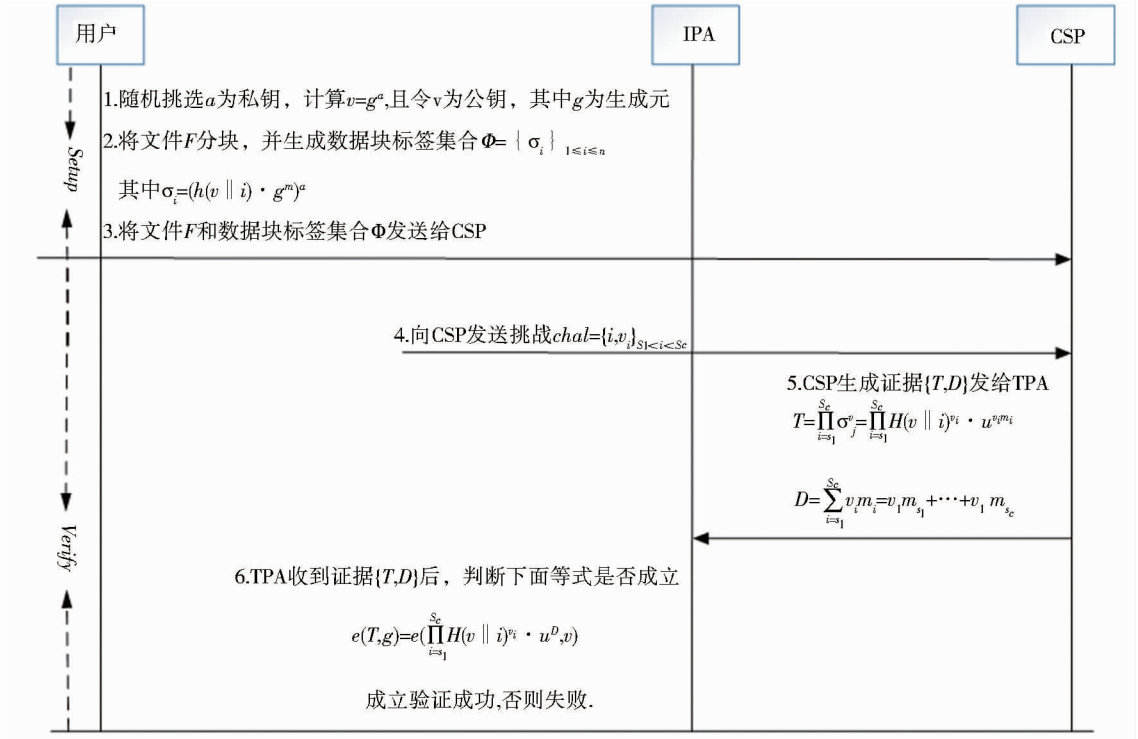


图 6 基于 BLS 的 PDP 验证机制

Fig.6 PDP verification mechanism based on BLS

钥管理,减少了通信和计算开销.该方案在计算 Diffie-Hellman 假设的严格性条件下,在随机预言模型中可证明是安全的.Li 等[9]引入基于模糊身份的验证机制解决了云数据完整性中复杂密钥管理问题,通过利用生物识别技术作为模糊身份,提出了一种基于模糊身份的审计结构,且该协议有一定的容错性,但该方案所需的计算和通信开销较大.

基于代数签名的 PDP 验证机制与其他的验证机制相比,只需低网络带宽且有较低的计算开销和较高的效率.代数签名是指具有某些代数性质的哈希函数将较大的数据文件压缩成很小的比特串参与运算和通信.Wang 等[10]提出了一种基于代数签名的远程数据审查方案,以验证存储在云中数据的完整性.该方案中的代数签名,运算速度可能达到数十至数百兆字节,在挑战阶段和响应阶段数据块的大小仅为 200 B 和 8 kB,进一步减少了带宽开销.当云服务器中的一部分数据块损坏或删除时,第三方审计者只能验证少量的数据块来检查数据的完整性,但为了数据的绝对安全需要验证完整的数据,因此需要较大的计算开销去保证数据的安全性.

2.2 基于应用场景的 PDP 验证机制

为了满足不同用户的需求,不同应用场景的 PDP 验证机制被相继提出.

2.2.1 支持动态操作的 PDP 验证机制

考虑到用户会随时更新存储在云中的数据,Ateniese 等[11]提出了支持部分动态的数据持有性证明方案,但该方案无法执行插入操作.为了解决该问题,Erway 等[12]基于跳表结构提出了支持全动态操作的 PDP 机制,但每次在认证过程中需要大量的辅助认证消息,且认证路径过长,使其计算和通信开销较大.Wang 等[13]提出基于 Merkle-Tree 的 PDP 验证机制,该动态结构相比跳表更为简单,且可确保数据节点在位置上的完整性.

2013 年,Zhu 等[14]引入了一个简单的数据结构(如表 1),称为索引哈希表(index hash table,IHT),用于记录每个数据块的变化.IHT 的结构就像一个一维数组,其中包含索引号、块号、版本号和随机值.基于 IHT 的 PDP 验证机制减少了存储成本和通信开销.但在进行插入和删除操作上效率不高,因为它们会导致平均 $N/2$ 个元素的调整,其中 N 是所有数据块的总数.大多数动态 PDP 方案将数据块的索引运用到其标签的计算中,来认证数据块在云服务器中的位置的正确性,例如 $h(i \| v)$、$h(name \| i)$等.但是,如果插入或删除一个块,所有后续块的索引都会改变,那么这些块的验证标签必须重新计算,这需要用户耗费巨大的计算资源.

表 1　索引哈希表

Tab.1　Index hash table

	No.	B_i	V_i	R_i
表的头部	0	0	0	0
添加	1	1	2	r_1^*
	2	2	1	r_2
删除	3	4	1	r_3
	4	5	1	r_5
插入	5	5	2	r_5^*
	…	…	…	…
	n	n	1	r_n
添加	$n+1$	$n+1$	1	r_{n+1}

为此,Tian 等[15]基于新的动态结构(如图 7)-动态哈希表(dynamic hash table,DHT)提出了新的标签构建方式,该表由 2 种元素构成:文件元素和块元素.文件元素由序列号和 ID 号标识,块元素由时间戳唯一标识,文件元素和块元素之间由指针进行连接,进行插入删除操作只需更改指针即可,有效地减少了通信成本且提高了效率.为了支持动态操作,大部分数据持有性证明方案都引入了动态结构,但动态结构需要一定的存储空间,为此 Jin 等[16]提出基于索引切换器的支持动态和仲裁的公共审计方案,通过索引切换器将数据块的序列号和索引号进行切换,在没有引入动态结构的前提下,有效地处理了数据动态的问题. 但 Jin 等[16]的方案没有考虑到数据块的序列号和索引号之间进行切换所带来的隐私泄露问题.

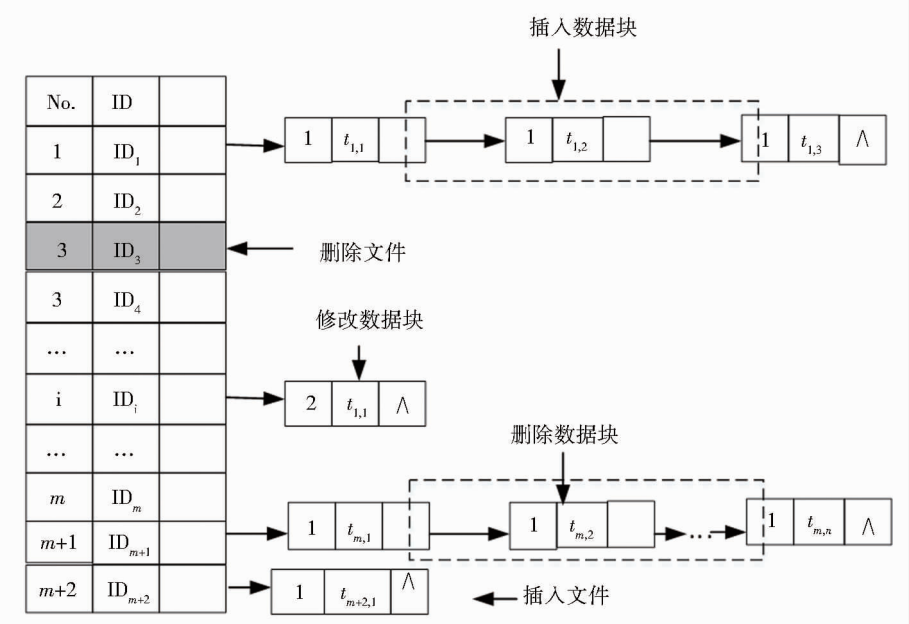


图 7　动态哈希表

Fig.7　Dynamic hash table

为了提高动态操作效率,跳表、Merkle-Tree、哈希表和双链表等多种动态结构被相继提出,支持动态操作已经成为目前多数 PDP 验证方案重要的功能,在不泄露用户隐私且轻量级的情况下,需要设计出一套完整的动态 PDP 验证机制,该机制支持将动态操作应用到各种场景,以满足更多的应用.

2.2.2 支持隐私保护的PDP验证机制

为了防止第三方审计者泄露用户的隐私,Wang等[17-18]采用随机掩码技术解决了该问题.该方案的实现原理是在基于BLS签名的PDP验证机制,为了防止在证据计算过程中 $\mu=\sum_{i=s_1}^{s_c} v_i m_i$ 导致数据隐私泄露,为此,引入2个参数 r、γ 来隐藏 μ 值.Wang等[17-18]保护数据隐私的PDP验证机制能防止第三方审计者泄露用户的隐私,为用户的隐私提供了安全保障.支持隐私保护的PDP验证机制的具体实现过程中的 *Setup* 阶段与图6中 *Setup* 阶段一致,验证阶段如图8所示.

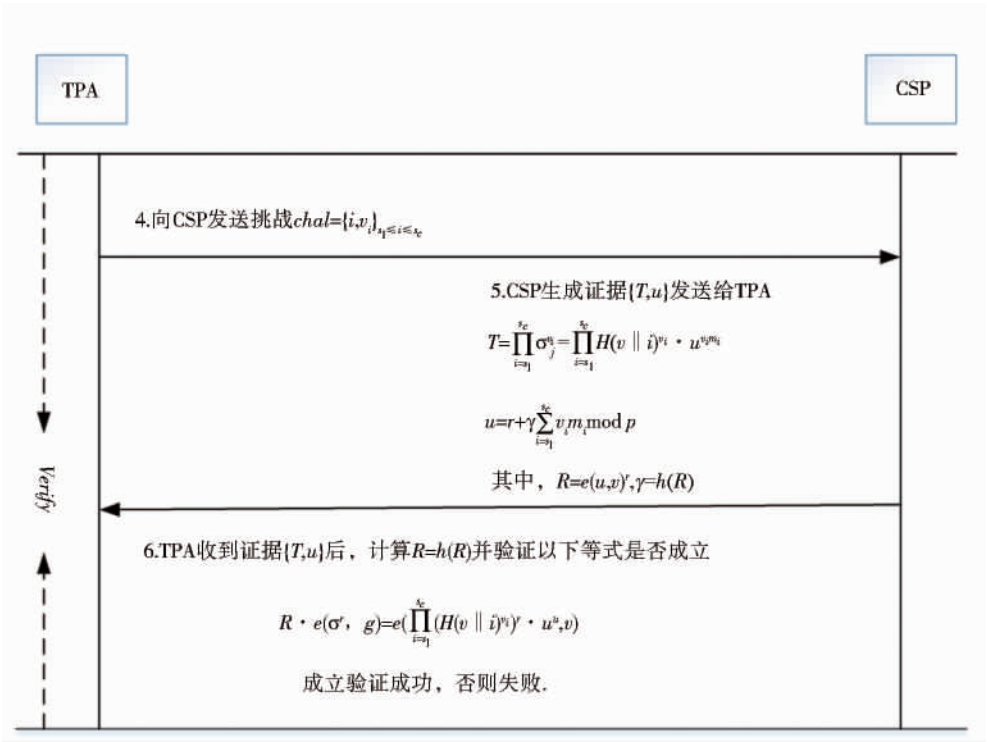


图8 支持隐私保护的PDP验证机制

Fig.8 PDP authentication mechanism supporting privacy protection

Wang等[19]利用环签名概念构造同态身份验证者并命名为"Oruta",使得第三方审计者和云服务提供商无法知晓数据,但不支持数据动态操作.Patil等[20]提出了利用Merkle哈希树(MHT)对编码数据进行索引,有效地保护了用户数据隐私且实现了动态的公共审计.它支持数据进行插入、修改和删除操作,且所提出的审计方案使得通信和计算成本最小化.Kumar等[21]提出了一种增强保护数据隐私的系统,利用RSA算法和AES算法对数据加密,在数据存储到云端之前,这2种算法的混合为其提供了更好的机密性.

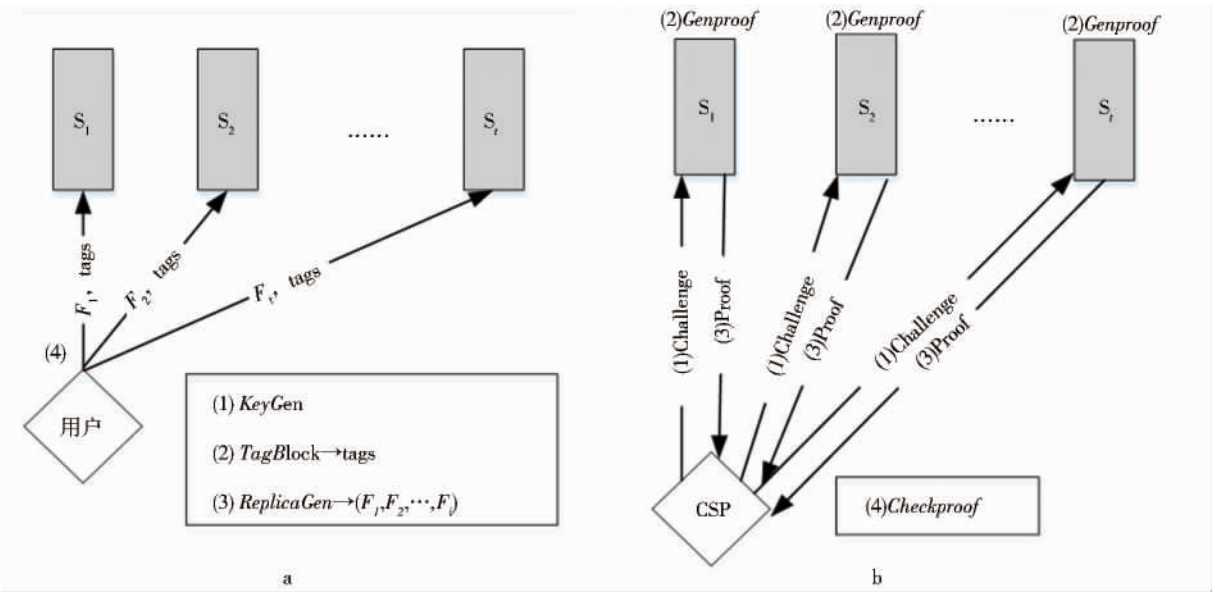
技术是保护隐私最直接的手段,基于隐私的PDP验证机制多数采用随机掩码技术,对于多样性和复杂性的大数据时代,有待进一步开发更多的技术来保护用户的隐私.

2.2.3 支持多副本的PDP验证机制

为了防止云服务器被攻击或发生故障导致数据永久丢失,Curtmola等[22]提出了第一个多副本PDP验证机制,该机制使用一组标签来验证任意一个副本,减少了验证标签所带来的计算负担. 基于Ateniese等[5]设计RSA签名的PDP机制,Curtmola等[22]对其进行了扩展,该验证机制的详细步骤如图9所示.

Merkle[23]提出了将每个副本文件组成一个Merkle哈希树,便于支持更新操作.Liu等[24]提出多副本动态公共审计方案,将所有副本所包含文件块的哈希值都组成一棵Merkle哈希树,与文献[23]方案相比,减少了更新开销.然而,Merkle哈希树的存储空间随着副本的数量增加而增大,给CSP带来了负担.Guo等[25]提出的动态多副本数据持有性证明(DPDPR)方案,设计了一个新的标签,只需在云服务器端存储一棵基于

原文件数据块哈希值的隐式索引 Merkle 哈希树,减少了云服务提供商的存储负担.但在数据上传到云服务器时,云服务提供商需验证每个数据块的哈希值,为其带来巨大的计算负担.在云存储环境中,数据的完整性和可靠性是至关重要的安全问题,而动态更新也是必不可少的部分.现有的大部分方案都使用 Merkle 哈希树来支持动态操作,但是在验证过程中其结构非常庞大.Zhao 等[26]引入多分支树来实现动态更新,进而简化了验证的结构.然而,在这些个方案中,数据副本和标签的生成都由用户来完成,增加了用户的计算压力.



a.初始化过程;b.验证过程.

图 9　支持多副本的 PDP 验证机制

Fig.9　PDP verification mechanism supporting multiple copies

由于云服务提供商给用户提供低价且海量的存储空间,越来越多的用户为了提高数据的可用性选择采用多副本方式存储数据,如何确保云服务提供商按照用户的要求进行存储,这是一个值得深入研究的问题.此外,在确保数据完整性的同时并支持对多个副本进行动态操作有待探索和进一步研究.

2.2.4　支持批量的 PDP 验证机制

近些年来,在云存储环境中,人们为了优化审计性能和提高更新效率,批量审计和批量更新也成为数据持有性证明机制不可缺少的功能.Qi 等[27]针对基于秩的 Merkle 哈希平衡树,设计了一种基于更新锁定的批处理更新方案,该方案消除了更新和重新平衡树的相邻分支之间的冲突.Guo 等[28]基于 Merkle 哈希树提出了一种批量更新算法,该算法可以 1 次执行和验证多个更新操作,避免了重复的计算和传输.如图 10 所示,挑战 w_3 和 w_7 2 个叶节点的所需路径分别为 $\Omega_3=\{w_{26},w_{22},w_{15},w_4,w_3\}$ 和 $\Omega_7=\{w_{26},w_{21},w_{17},w_8,w_7\}$,相同的节点 w_{26} 被检索了 2 次,节点 w_{25} 和根节点的哈希值被计算了 2 次,这仅仅是挑战 2 个节点,随着挑战节点数量的增多,那重复节点的检索和计算必会浪费巨大的计算和通信资源.为此,Guo 等[28]的方案提出的方法使得被挑战的节点同时进行,必要的节点只检索 1 次,必要节点的哈希值只计算 1 次,节省了大量的计算和传输成本,如图 11 所示.

为了便于动态操作,基于批量的 PDP 验证机制的多数方案采用的动态结构为 Merkle 哈希树,但该动态结构所占存储空间较大,如何利用更简单的动态结构确保云存储环境下数据的完整性且高效实现批处理审计和更新需要进行更深一步的研究.

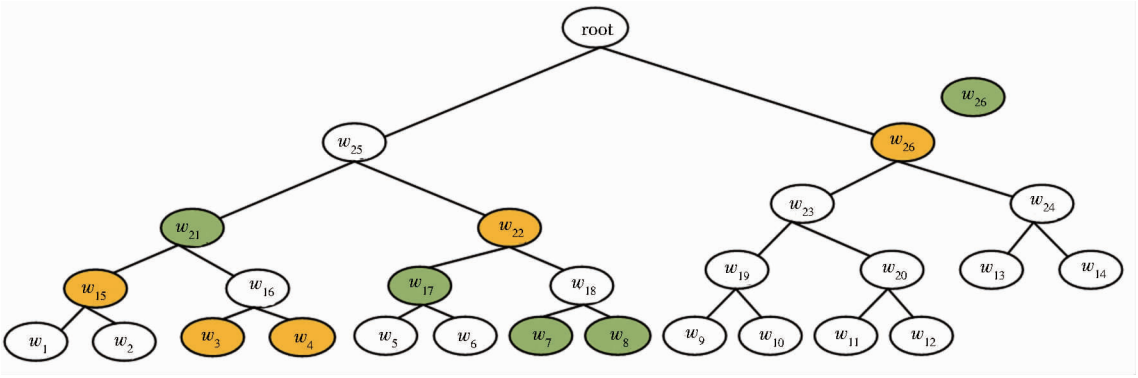


图 10 依次挑战节点 w_3 和 w_7

Fig.10 Challenge w_3 and w_7 nodes in turn

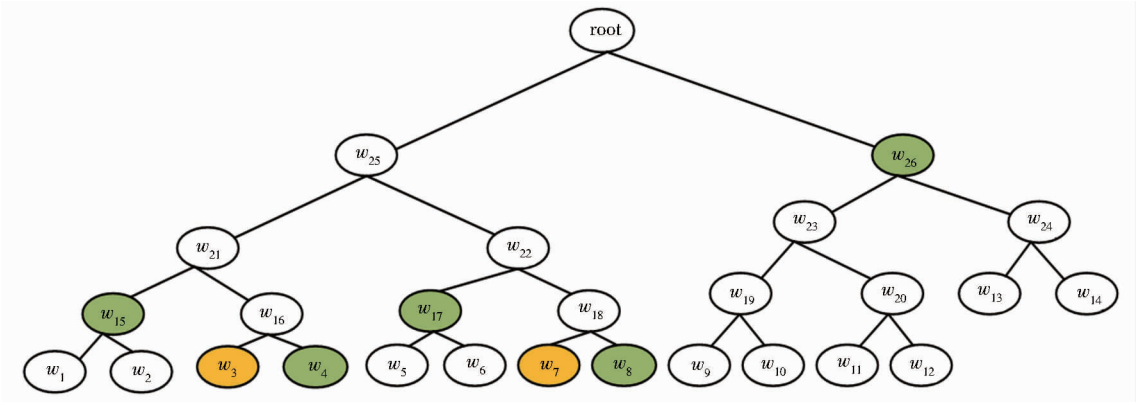


图 11 批量挑战节点 w_3 和 w_7

Fig.11 Challenge w_3 and w_7 nodes in batches

2.2.5 支持共享的 PDP 验证机制

为了满足多个用户在云端共享数据文件，Wang 等[29]通过使用代理重新签名的思想，在多个用户共享数据的同时能安全的实行用户撤销，是一种新颖的数据共享审核机制.该方案通过利用云当作代理签名者，减轻了用户的负担，而且第三方审计也能验证共享数据的完整性，但该方案的通信开销耗费较大.Trueman 等[30]使用数据签名算法生成配对密钥，并用同态可认证环签名方案为每个数据块生成有效的签名，只有组内用户才能验证签名，为其提供了有效的审核机制，且该方案使用覆盖树算法确保了数据的最新版本，但该方案的计算开销较大.Yang 等[31]提出了一个有效的支持共享数据完整性验证方案，不仅保护了用户的身份隐私，还实现了身份可追溯性.但该验证方案还不支持动态操作功能.

随着越来越多共享资源的出现，支持共享的 PDP 验证机制越来越受到学术界和工业界的重视.在实现群组成员共享资源的同时能安全的实行用户撤销成为支持共享的 PDP 验证机制的基本功能.一个支持共享数据完整性验证方案，如何实现用户高效且安全的撤销并实现轻量级的验证是该机制的目标，此外，在功能上结合动态操作的共享 PDP 验证机制有待于探索和研究.

2.3 基于不同实体的 PDP 验证机制

针对单点故障，Raziqa 等[32]基于分布式哈希表提出了一个分布式公共设计方案，在这项工作中有多个第三方审计者，基于点对点协议将审计者组成一个分布式哈希表，第三方审计者统一使用一致的哈希函数，通过其物理标识(例如 IP 地址、端口号或某个唯一数字)生成唯一的 m 位的关键值(A_1，A_3 和 A_6)，该值可以帮助第三方审计者组织成如图 12 所示的适当结构.图 12 显示了分布式哈希表环是以模为 2^m 形成$\{0-(2m-1)\}$个地址空间，且每个外包的数据或文件都需经过哈希处理形成一个 m 位的文件密钥，该密钥映射

到$\{0-(2m-1)\}$个地址空间的公共审计模型,该模型审核每个审计者提供的文件密钥及其唯一标识.例如,生成的文件密钥 f_3、f_4 和 f_5 可以通过 A_3 进行审核.该方案中使用关键值可以将外包的大文件进行分组审核,该方案被称为一种经济有效的解决方案且没有单点依赖性.

针对多用户的公共审计问题,Yuan 等[33]提出支持多用户修改和高效撤销某用户的数据持有性证明方案.此外,为了提高数据的可用性和可靠性,用户可能选择多个云平台来存储自己的数据,如何确保跨云存储数据的完整性,这也是需要深入研究的问题.Zhu 等[34-36]对跨平台云存储的数据完整性证明机制过程模型化,但对下一步的实现过程没有详细说明,在实际应用中需要设计该证明机制,进而去满足更多的应用.

近些年来随着区块链技术迅速崛起,它被应用到数据持有性证明中,Xu 等[37]基于区块链技术提出了分布式可仲裁数据审计方案,利用区块链网络作为自记录通道实现了不可抵赖性验证.现如今已步入 5G 时代,区块链和神经网络技术等各种应用给人们的生活带来了越来越大的影响,相信在不久的未来,云存储会结合各种应用技术来实现高安全和高效率的验证方案.

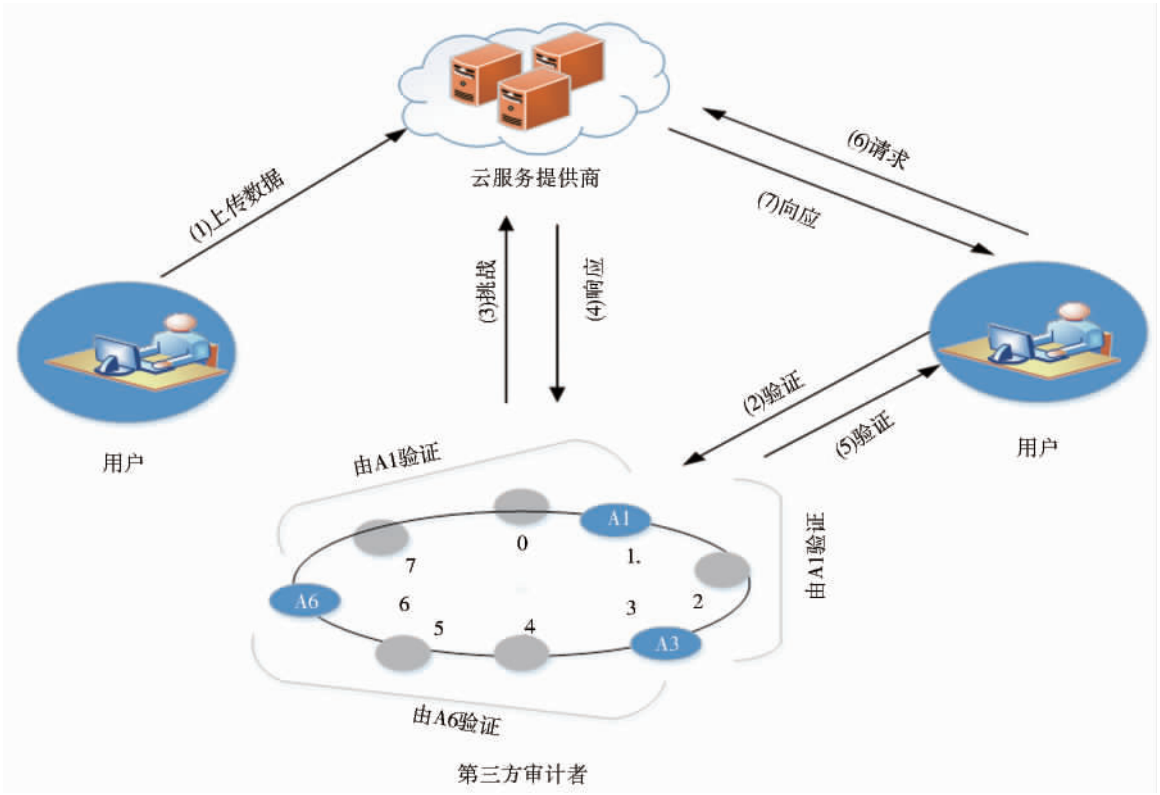


图 12　多个 TPA 的云存储模型

Fig.12　Cloud storage model of multiple TPAs

3　结束语

针对数据持有性证明的模型和审计框架进行了论述,并根据不同用户的需求,对主要的数据持有性证明方案进行了分类归纳,分析了数据持有性证明机制各个应用场景的现状和不足,并对其进行了展望.随着云存储在生活中的大量应用,用户对数据完整性审计方案的效率和安全等要求不断提高,但是现有的审计机制在性能上还未能实现理想中的轻量级,在功能方面各有侧重不能顾全,而且,云存储和区块链技术仍处于探索发展阶段,且云存储技术结合边缘计算和神经网络的智能审计方案也有待探索和研究.

参 考 文 献:

[1] 杨青.50亿条公民信息泄露 京东前员工牵涉其中[EB/OL]. [2017-3-11]. https://www.chinacourt.org/article/detail/2017/03/id/2576962.shtml.

[2] 数安时代 GDCA.企业陷入数据泄露的启示公民信息泄露[EB/OL]. [2019-6-5]. https://m.sohu.com/a/318749838_604699.

[3] 首席知产官.腾讯云丢失数据遭千万索赔,腾讯:抱歉! 13万不能再多了 [EB/OL]. [2018-8-7]. https://baijiahao.baidu.com/s? id=1608142183319333964.

[4] DESWARTE Y, QUISQUATER J J, SAïDANE A. Remote integrity checking[M]//Integrity and Internal Control in Information Systems VI, Boston: Kluwer Academic Publishers,2004: 1-11. DOI:10.1007/1-4020-7901-x_1.

[5] ATENIESE G, BURNS R, CURTMOLA R, et al. Provable data possession at untrusted stores[M]//Proceedings of the 14th ACM Conference on Computer and Communications Security -CCS'07, 2007,New York: ACM Press, 2007: 598-610. DOI:10.1145/1315245.1315318

[6] BONEH D, LYNN B, SHACHAM H. Short signatures from the Weil pairing[M]//Advances in Cryptology — ASIACRYPT 2001, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001: 514-532. DOI:10.1007/3-540-45682-1_30.

[7] GENTRY C, RAMZAN Z. Identity-based aggregate signatures[M]//9th International Conference on Theory and Practice of Public-Key Cryptography, New York: Springer-Verlag, 2006.

[8] ZHAO J N, XU C X, LI F G, et al. Identity-based public verification with privacy-preserving for data storage security in cloud computing[J]. IEICE Trans Fundamentals, 2013, E96,A(12): 2709-2716. DOI:10.1587/transfun.e96.a.2709.

[9] LI Y N, YU Y, MIN G Y, et al. Fuzzy identity-based data integrity auditing for reliable cloud storage systems[J]. IEEE Trans Dependable Secure Comput, 2019, 16(1): 72-83. DOI:10.1109/TDSC.2017.2662216.

[10] WANG X D, JIAO W Z, YANG H, et al. Algebraic signature based data possession checking method with cloud storage[Z]. 11th International Conference on Prognostics and System Health Management (PHM-2020 Jinan), 2020, Jinan, China. DOI:10.1109/PHM-Jinan48558.2020.00010.

[11] ATENIESE G, DI PIETRO R, MANCINI L V, et al. Scalable and efficient provable data possession[Z].The 4th International Conference on Security and Privacy in Communication Netowrks, Istanbul, Turkey, 2008. DOI: 10.1145/1460877.1460889.

[12] ERWAY C C, KÜPÇÜ A, PAPAMANTHOU C, et al. Dynamic provable data possession[J]. ACM Trans Inf Syst Secur, 2015, 17(4): 1-29. DOI:10.1145/2699909.

[13] WANG Q, WANG C, LI J, et al. Enabling public verifiability and data dynamics for storage security in cloud computing[M]//Computer Security -ESORICS 2009, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009: 355-370. DOI: 10.1007/978-3-642-04444-1_22.

[14] ZHU Y, AHN G J, HU H X, et al. Dy-namic audit services for outsourced storage in clouds[J]. IEEE Trans on Services Computing, 2013,6(2): 227-238. DOI: 10.1109/TSC.2011.51.

[15] TIAN H, CHEN Y X, CHANG C C, et al. Dynamic-hash-table based public auditing for secure cloud storage[J]. IEEE Trans Serv Comput, 2017, 10(5): 701-714. DOI:10.1109/tsc.2015.2512589.

[16] JIN H, JIANG H, ZHOU K. Dynamic and public auditing with fair arbitration for cloud data[J]. IEEE Trans Cloud Comput, 2018, 6(3): 680-693. DOI:10.1109/TCC.2016.2525998.

[17] WANG C, WANG Q, REN K, et al. Privacy-preserving public auditing for data storage security in cloud computing[C]//Proceedings IEEE INFOCOM, San Diego, USA, 2010: 1-9. DOI:10.1109/INFCOM.2010.5462173.

[18] WANG C, CHOW S S M, WANG Q, et al. Privacy-preserving public auditing for secure cloud storage[J]. IEEE Trans Comput, 2013, 62(2): 362-375. DOI:10.1109/TC.2011.245.

[19] WANG B Y, LI B C, LI H. Oruta: privacy-preserving public auditing for shared data in the cloud[J]. IEEE Trans Cloud Comput, 2014, 2(1): 43-56. DOI:10.1109/TCC.2014.2299807.

[20] PATIL J M, CHAUDHARI S S. Efficient privacy preserving and dynamic public auditing for storage cloud[Z]. International Conference on Nascent Technologies in Engineering (ICNTE) Navi Mumbai, India, 2019. DOI: 10.1109/ICNTE44896.2019.8945817.

[21] KUMAR A. A novel privacy preserving HMAC algorithm based on homomorphic encryption and auditing for cloud[Z]. Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 2020. DOI: 10.1109/I-SMAC49090.2020.9243340.

[22] CURTMOLA R, KHAN O, BURNS R, et al. MR-PDP: multiple-replica provable data possession[Z].The 28th International Conference on Distributed Computing Systems, Beijing, China, 2008. DOI:10.1109/ICDCS.2008.68.

[23] MERKLE R C . A certified digital signature[M]// Conference on the Theory and Application of Cryptology,New York:Springer, 1989:218-238.

[24] LIU C, RANJAN R, YANG C, et al. MuR-DPA: top-down levelled multi-replica merkle hash tree based secure public auditing for dynamic big data storage on cloud[J].IEEE Transactions on Computers,2015,64(9): 2609-2622. DOI: 10.1109/TC.2014.2375190.

[25] GUO W, QIN S J, GAO F, et al. Dynamic proof of data possession and replication with tree sharing and batch verification in the cloud[J]. IEEE Trans Serv Comput, 2020,9:1-1. DOI:10.1109/TSC.2020.3022812.

[26] ZHAO Y L, LI S Q. Dynamic flexible multiple-replica provable data possession in cloud[Z].17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 2020. DOI:10.1109/ICCWAMTIP51612.2020.9317378.

[27] QI Y, XIN T, HUANG Y . Enabling efficient verification of dynamic data possession and batch updating in cloud storage[J]. KSII Trans Internet Inf Syst, 2018, 12(6): 2429-2449. DOI:10.3837/tiis.2018.06.001.

[28] GUO W, ZHANG H, QIN S J, et al. Outsourced dynamic provable data possession with batch update for secure cloud storage[J]. Future Gener Comput Syst, 2019, 95: 309-322. DOI:10.1016/j.future.2019.01.009.

[29] WANG B Y, LI B C, LI H. Panda: public auditing for shared data with efficient user revocation in the cloud[J]. IEEE Trans Serv Comput, 2015, 8(1): 92-106. DOI:10.1109/TSC.2013.2295611.

[30] TRUEMAN T E, NARAYANASAMY P. Ensuring privacy and data freshness for public auditing of shared data in cloud[Z]. IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), Bangalore, India, 2015. DOI:10.1109/CCEM.2015.36.

[31] YANG G Y, YU J, SHEN W T, et al. Enabling public auditing for shared data in cloud storage supporting identity privacy and traceability[J]. J Syst Softw, 2016, 113: 130-139. DOI:10.1016/j.jss.2015.11.044.

[32] MASOOD R, PANDEY N, RANA Q P. DHT-PDP: a distributed hash table based provable data possession mechanism in cloud storage[Z]. 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2020. DOI:10.1109/ICRITO48877.2020.9198019.

[33] YUAN J W, YU S C. Public integrity auditing for dynamic data sharing with multiuser modification[J]. IEEE Trans Inf Forensics Secur, 2015, 10(8): 1717-1726. DOI:10.1109/TIFS.2015.2423264.

[34] ZHU Y, HU Z X, WANG H X, et al. A collaborative framework for privacy protection in online social networks[Z]. The 6th International ICST Conference on Collaborative Computing: Networking, Applications, Worksharing, Chicago, USA, 2010. DOI:10.4108/icst.collaboratecom.2010.52.

[35] ZHU Y, HU H X, AHN G J, et al. Collaborative integrity verification in hybrid clouds[Z].The 7th International Conference on Collaborative Computing: Networking, Applications and Worksharing, Orlando, USA, 2011. DOI:10.4108/icst.collaboratecom.2011.247089.

[36] ZHU Y, WANG H X, HU Z X, et al. Efficient provable data possession for hybrid clouds[Z].The 17th ACM conference on Computer and communications security-CCS'10, Chicago, USA,2010. DOI:10.1145/1866307.1866421.

[37] XU Y, REN J, ZHANG Y, et al. Blockchain empowered arbitrable data auditing scheme for network storage as a service[J]. IEEE Trans Serv Comput, 2020, 13(2): 289-300. DOI:10.1109/TSC.2019.2953033.

(责任编辑:孟素兰)