基于区块链的双重可验证云存储方案

冯涛[1]，孔繁琪[1]，柳春岩[2]，马蓉[1]，Maher Albettar[3]
（1. 兰州理工大学计算机与通信学院，甘肃 兰州 730050；2. 兰州理工大学经济与管理学院，甘肃 兰州 730050；
3. 康考迪亚大学康考迪亚信息系统工程研究所，蒙特利尔 H3G 1M8）

摘 要：针对工业物联网（IIoT）中，工业设备将数据存储到云端，导致数据易被篡改且无法追踪恶意用户，引起恶性循环的问题，借助区块链的可溯源性和不可篡改性等特点，提出了一种基于区块链的双重可验证云存储方案。首先，使用同态加密技术加密数据并上传至云端，确保传输安全性，实现数据的隐私保护；其次，将聚合密文和上传者信息存储在区块链上，有效避免数据被非法用户篡改的风险，并能对恶意用户进行追溯，实现对云端数据完整性的双重验证；最后，进行安全性分析，证明所提方案比同类方案更加安全可靠。

关键词：区块链；可验证计算；隐私保护；云存储
中图分类号：TP309
文献标识码：A
DOI: 10.11959/j.issn.1000−436x.2021225

Dual verifiable cloud storage scheme based on blockchain

FENG Tao[1], KONG Fanqi[1], LIU Chunyan[2], MA Rong[1], Maher Albettar[3]
1. School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China
2. School of Economics and Management, Lanzhou University of Technology, Lanzhou 730050, China
3. Concordia Institute for Information Systems Engineering, Concordia University, Montréal H3G 1M8, Canada

Abstract: In response to the problem that in the industrial Internet of things (IIoT), industrial equipment stored data in the cloud, causing data to be easily tampered with and unable to track malicious users, causing a vicious circle, with the help of the traceability and immutability of the blockchain, a dual verifiable cloud storage scheme based on the blockchain was proposed. Firstly, homomorphic encryption technology was used to encrypt data and upload it to the cloud , which ensured transmission security and realized data privacy protection. Secondly, the aggregated ciphertext and uploader information were stored on the blockchain, which could effectively avoid the risk of data being tampered with by illegal users, and could trace malicious users to achieve double verification of cloud data integrity. Finally, a safety analysis was carried out to prove that the proposed scheme is safer and more reliable than similar schemes.

Keywords: blockchain, verifiable calculation, privacy protection, cloud storage

1 引言

近年来，工业物联网（IIoT, industrial Internet of things）推动各种智能应用迅速崛起，在制造业、医疗保健、智能能源电网和智能交通系统等领域受到了极大的关注，并成功地融入人们生活和工作[1]。为了减少本地服务器的计算开销和节约存储空间，大量工业设备在业务中产生的实时感知数据和敏感信息需要存储到云端。云存储是云计算的重要分支[2]，它为数据的存储和分析提供了高质量的按需服务，但在网络访问和数据上传过程中，其带来的隐私泄露问题也日趋严重。云服务器或敌手恶意获取和篡改用户的数据，导致敏感信息泄露，且对恶意用户无法追踪，从而导致恶性循环。云存储数据

收稿日期：2021−07−28；**修回日期**：2021−10−08
基金项目：国家自然科学基金资助项目（No.62162039, No.61762060）；甘肃省科技厅重点研发计划基金资助项目（No.20YF3GA016）
Foundation Items: The National Natural Science Foundation of China (No.62162039, No.61762060), The Key Research and Development Program of Gansu Province (No.20YF3GA016)

的机密性、完整性对工业物联网健康发展非常重要。目前，云存储主要存在以下几个问题：1) 数据的隐私和传输安全得不到保护；2) 云存储数据的正确性和完整性得不到有效验证；3) 无法追踪恶意用户，从而导致恶性循环。因此，针对数据处理的可信性和安全性问题，为保障用户查询真实数据，研究一种可验证云存储方案具有重要意义。

在过去的几十年里，许多学术界和工业界研究者已对可验证计算等相关问题进行了广泛而深入的探索。可验证计算方案主要涉及应用安全领域、计算机理论领域和密码学领域[3]，并广泛应用在计费系统、外包计算、电子投票系统等各种场景中。在应用安全领域，大多方案从应用角度出发，以基于审计和各种安全协处理器工具为主；在计算机理论领域，大多方案依赖于复杂的概率可检测证明（PCP, probabilistically checkable proofs）[4]，但由于极高的计算成本，它们很难在实际生活中应用。在密码学领域，主要的密码学工具有同态加密[5]、混淆电路、基于属性的加密、同态签名[6]等。Chaum 等[7]利用盲签名构造了具体协议，其提出的 electronic wallet 模型是利用密码学手段研究可验证计算的开端。随着区块链技术的发展[8]，其可以提供完美的透明度和分布式可验证性，以及利用密码学技术和共识协议来保证网络安全传输，为可验证计算增加了更多的应用场景和功能性，创建了更安全可信的数据共享平台。区块链与可验证计算共同进行数据隐私保护，有利于数据分布式存储，以及对云端数据的完整性进行检验，有效控制数据破坏行为所造成的数据安全问题，也有利于监管部门对外包计算等场景进行有效监管，追踪恶意输入。

近年来，有学者提出了基于区块链的可验证云存储方案，但依然存在以下 3 个问题尚未解决： 1) 在数据传输过程中，保证数据安全性、机密性的同时，没有考虑数据的可用性；2) 方案不能在云存储服务环境下实现整体数据和单个数据的可验证；3) 无法实现对恶意修改和破坏数据用户的追踪。针对上述问题，本文提出了一种基于区块链的双重可验证云存储方案，解决了数据隐私泄露、不可验证、传输不安全和无法追溯恶意用户的问题。

本文主要的研究工作如下。

1) 提出了基于区块链的双重可验证云存储方案系统模型，并分析敌手的恶意行为。通过区块链技术可实现数据的一致存储和难以篡改，增加了上链数据造假的难度和成本，从而可以消除大部分使用者对数据信用的顾虑。本文构造的方案可保证数据的完整性以及计算过程的可靠性和可信度。使用同态加密技术对数据加密，在密文上进行计算，解决了隐私泄露和传输不安全等方面的问题。

2) 提出的基于区块链的双重可验证云存储方案，可对数据完整性进行有效核验。本文应用具有同态性质的哈希函数，云服务器可以聚合多用户的签名，并对聚合数据进行总体验证，解决了通信负担大、计算效率低等问题。需要追溯上传虚假数据的恶意用户时，进行单独验证，解决了在分布式计算和云存储时对恶意用户的追溯问题，有效避免了恶性循环。

3) 安全性分析表明，本文方案在数据加密和聚合时是一种高效的可验证方案。在保证数据隐私性的前提下，可追溯上传错误数据的恶意用户，比同类方案更加安全可靠。本文方案具有可行性、安全性和有效性的特点。

2　相关工作

在区块链服务网络模型下，研究者构建了结合可验证计算的安全协议，实现数据隐私保护、数据存证、数据核验等多种功能。按照可验证计算参与方数目，可将方案分为一对一可验证计算方案和多方可验证计算方案。

在一对一可验证计算方案中，Wang 等[9]提出了一种基于区块链的可验证数据完整性的个人健康档案共享方案。该方案实现了个人健康档案在共享过程中数据的安全性，提高了患者敏感数据的隐私性，但该方案的计算负担较高，应用场景单一，不具有可扩展性。Guo 等[10]提出了一种新的基于区块链技术的动态单点登录方案，实现了对加密数据的可靠搜索，并通过智能合约设计了加密的核对表，可以在区块链内实现安全的结果验证。该方案虽然提高了安全性，却极大地增加了计算量，降低了验证效率。Guo 等[11]提出了一种轻量级的可验证外包解密的 CP-ABE 协议方案。该方案在解密过程中加入验证算法，利用密文的可验证性来验证用户设备的正确性。然而，该方案不满足公开可验证性，因此无法解决用户与云服务平台因数据正确性所产生的问题。

在多方可验证计算方案中，Dimitriou[12]提出了一种基于区块链的投票系统，这一系统通过随机数发生器令牌来确保抗强迫性和无收据性，并且适用于大规模的选举。区块链的结构确保了可核查性和

可扩展性，从而增加了对选举的可信度，但是不能抗共谋攻击。Wang 等[13]结合区块链技术实现公平支付，解决了服务器返回错误结果的问题，其使用智能合约存储安全索引并执行搜索。这一方案只支持“与”门访问策略，表达不灵活。Dorsala 等[14]设计了一种基于证明的可验证计算公平协议方案，为比特币和以太坊等区块链系统提供了在其网络中生成交易的公开可验证性。该方案大量运用双线性映射导致效率降低，提高了脚本执行的成本。Zhang 等[15]为了实现去中心化的手续费公平支付方案，采用在区块链中临时冻结押金的形式，确保用户只要诚实执行就能得到搜索结果和服务费用。但该方案需要用户端在验证结果正确性的过程中进行大量的签名验证计算，用户开销较大。

基于区块链的可验证计算方案已有一定的研究基础，虽然根据区块链技术与可验证计算技术特征，研究者已提出了多种安全协议方案，但仍未解决云存储中所面临的隐私保护和数据传输安全问题，以及通过整体数据和单个数据验证而实现对恶意用户的追踪问题。

3 预备知识

3.1 同态哈希函数

同态哈希函数[16]满足以下 2 个性质。

1) 同态性。对于任意 2 个数据 m_1、m_2 和实数 w_1、w_2，都有 $H(w_1m_1+w_2m_2)=H(m_1)^{w_1}H(m_2)^{w_2}$。

2) 免碰撞性。攻击者不存在概率多项式算法，能伪造 (m_1,m_2,m_3,w_1,w_2)，并满足 $m_3\neq w_1m_1+w_2m_2$，使 $H(m_3)=H(m_1)^{w_1}+H(m_2)^{w_2}$。

3.2 同态加密

同态加密技术[17]可以在密文上进行特定数学运算，并且解密结果等于对应明文进行运算后的结果。在数据聚合、隐私保护等方面有着重要的应用。Enc 表示加密，Dec 表示解密，$\odot$ 和 $\oplus$ 分别表示在明文和密文上的运算，同态加密满足运算 $\mathrm{Dec}(\mathrm{Enc}(a)\odot\mathrm{Enc}(b))=a\oplus b$。满足同态性质的加密方式统称为同态加密技术。

加密分为对称加密和非对称加密。对称加密是指加密和解密使用同一个密钥，这种方式显著降低了计算开销；非对称加密采用公钥加密、私钥解密，且基于数学困难问题，加解密操作实现复杂度较大，效率较低，对于资源的占用也较多。本文方案使用对称隐私同态加密技术对原始数据进行加密，因为该算法加密速度快，容易实现。

4 方案模型

本节将阐述基于区块链的双重可验证云存储方案的系统模型、威胁模型等内容。

4.1 系统模型

本文提出的基于区块链的双重可验证云存储方案系统模型如图 1 所示。数据拥有者将密文和签名传至云端，上传者信息记录到区块链上，数据使用者可以向挖矿者节点申请验证总体数据和单个数据的完整性，并实现对恶意用户的追溯。其中包括 5 类参与实体：云服务器（CS, cloud server）、数据拥有者（DO, data owner）、数据使用者（DU, data user）、挖矿者节点（MN, miner node）、区块链（blockchain）。各实体角色及所属功能介绍如下。

1) 云服务器。云服务器是第三方不可信的存储服务器，具有强大的计算能力和巨大的存储空间来处理和维护数据所有者的数据。在本文模型中，它是函数计算的具体执行者，主要负责存储、聚合和计算数据拥有者上传的加密文件和签名，然后将计算和验证结果发送给数据使用者。

2) 数据拥有者。数据拥有者拥有原始数据。在数据加密阶段，数据拥有者利用流密钥加密算法对自己的原始数据进行加密；在签名产生阶段，数据拥有者用自己的私钥和随机数对文件生成签名，将密文及签名上传给云服务器进行存储，同时将对应的上传者信息记录到区块链中。

3) 数据使用者。数据使用者作为合法用户，可以向区块链中的挖矿者节点请求对总体数据和单独数据进行完整性验证。Du 获得正确的聚合结果后，向云服务器支付服务费，完成交易。

4) 挖矿者节点。挖矿者节点用于验证总体数据和单个数据，会严格执行本文算法，并对 DO 和 CS 返回验证结果，以确保隐私数据的完整性和正确性。当正在执行验证任务的挖矿者节点出现故障时，新的挖矿者节点从备用节点中递补。

5) 区块链。区块链主要负责链上存储数据使用者上传的签名和云服务器上传的聚合结果，接收授权数据使用者进行完整性验证。区块链上只需存储加密文件地址，即可追溯加密文件和上传此加密文件的数据拥有者。由于区块链是公开的，所有用户都可以浏览和访问，有效避免数据被非法修改，确保数据安全。

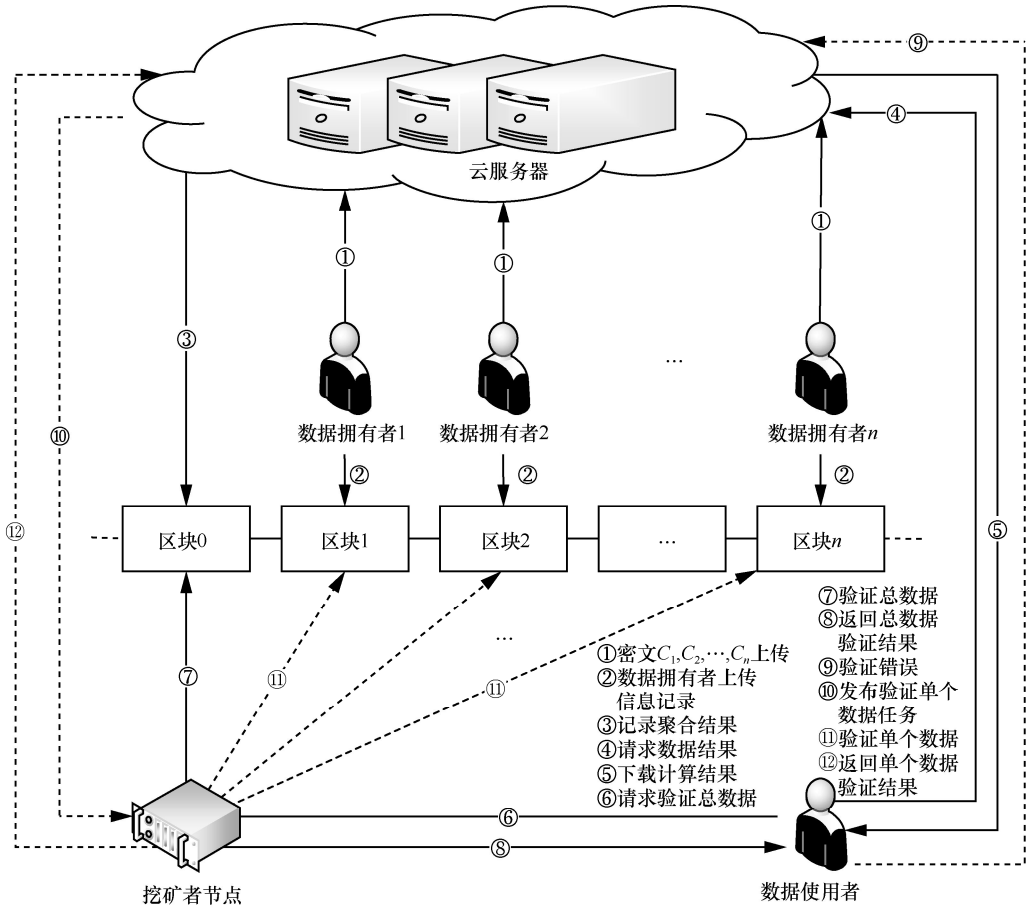

图 1　基于区块链的双重可验证云存储方案系统模型

4.2　威胁模型

本文方案中，云服务器是诚实且好奇的，它会遵守协议以及为用户返回计算和验证结果，但也可能通过恶意手段利用中间结果或最终结果来推测用户的隐私数据，给用户的隐私造成威胁。数据拥有者不是完全可信的，可能出于某些目的伪造或者提供错误的密文，不诚实地将数据上传至云端，或者上传过程中受到敌手恶意篡改、欺骗攻击。区块链具有完全可信性，可以引入区块链帮助数据使用者验证云服务器返回的结果和数据拥有者上传密文的正确性。区块链只用于记录上传者对应信息和聚合结果，不能给区块链上传任何原始数据。挖矿者节点是诚实可信的，会按照本文算法严格执行验证，并将验证结果反馈给数据使用者。

5　方案概述

可验证云存储方案时序如图 2 所示。首先，数据拥有者使用同态加密技术对数据加密和签名，多用户将密文 $C_1,C_2,\cdots,C_v$ 和签名上传至云服务器。云服务器将上传各个密文对应的上传者信息记录到区块 1～区块 n，将密文聚合结果上传到区块 0，其次，数据使用者向云服务器请求数据结果，云服务器返回计算结果。最后，数据使用者向挖矿者节点发布验证总数据任务，挖矿者节点接收验证任务后，在区块链头进行总体结果验证，并返回总数据验证结果。如果数据正确，云服务器就可以得到服务费，完成交易。如果验证数据不正确，则把错误结果返回给云服务器并终止交易。云服务器接收到错误结果反馈后，再次向挖矿者节点发送验证单个数据信息任务，查询上传了错误信息的数据拥有者。挖矿者节点收到验证任务后，分别访问区块 1～区块 n，再将单个数据验证结果返回给云服务器。追溯到的恶意用户将会受到惩罚，并且用户可拒绝支付服务费用。

6　算法构造

本节对具体的基于区块链的双重可验证云存

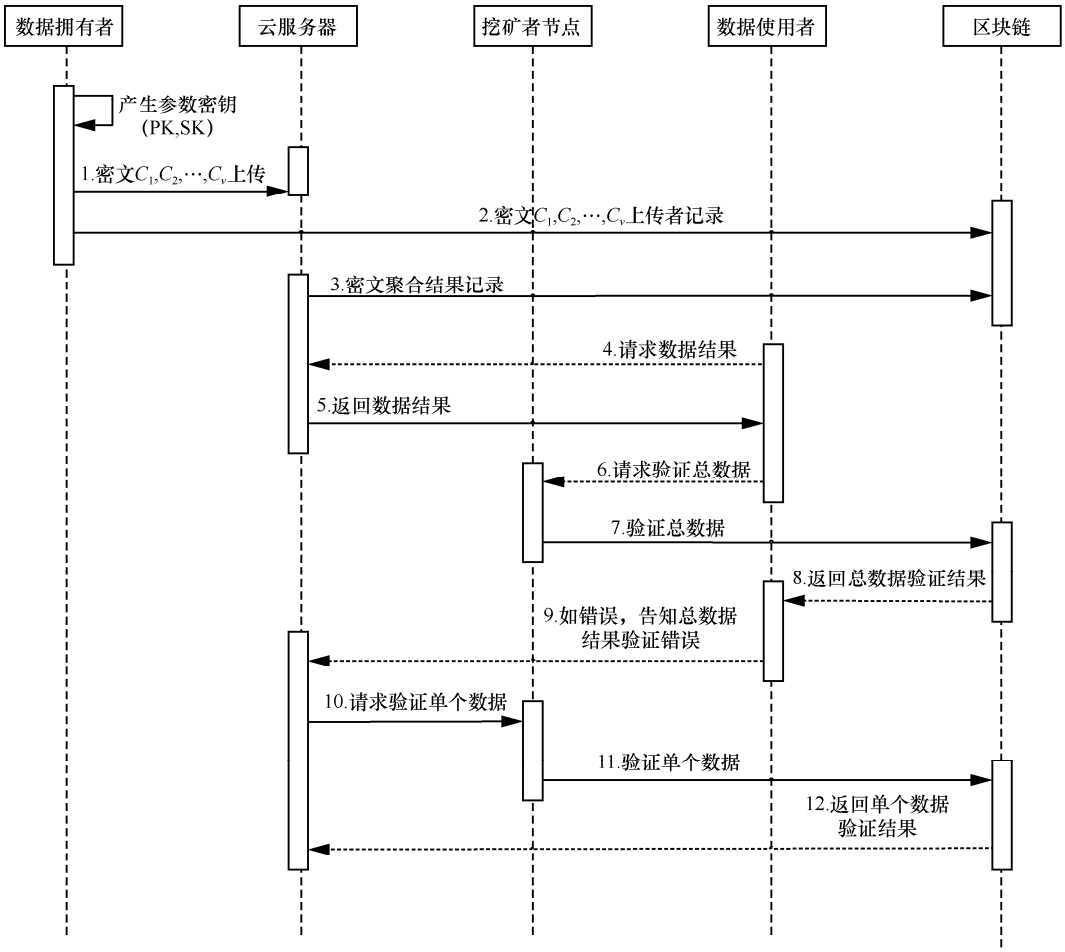

图 2 可验证云存储方案时序

储方案的算法进行详细介绍。

6.1 数据加解密聚合算法

6.1.1 数据准备阶段

本文方案使用流密钥加密算法对原始数据进行同态加密，因为该算法加密速度快，容易实现，并且在本文中每个数据使用者与云服务器不共享相同的密钥，因此安全性较高。

1) 密钥分配。在密钥分配阶段，系统为每个数据使用者广播由基于密钥长度可变的流加密算法簇产生的密钥 kn 。

2) 计数器 rt 。为了使云服务器能检验数据拥有者上传的是否是最新数据，每个数据拥有者产生一个计数 rt ，并对其进行初始化，来保证数据的实时性和抵御重放攻击能力。

3) 数据加密。数据拥有者 L 将原始数据 m_L 进行加密后，再上传至云服务器，保证了数据的隐私性，其加密算法为

$$c_L = \mathrm{Enc}(m_L) = m_L + \mathrm{rt}k_L \tag{1}$$

6.1.2 云服务器聚合密文

1) 密文聚合处理阶段。云服务器将 V 个数据拥有者上传的密文通过加密算法进行聚合。由于采用的是加法同态加密，因此不需要对密文进行解密，可以直接对上传数据进行相关运算。这既降低了计算开销，也增加了方案的安全性，减少了隐私泄露的风险。其聚合函数为

$$\begin{aligned}&\mathrm{CAGG} = c_1 + c_2 + \cdots + c_V = \\ &\mathrm{Enc}(m_1) + \mathrm{Enc}(m_2) + \cdots + \mathrm{Enc}(m_v)\end{aligned} \tag{2}$$

2) 数据解密阶段。云服务器收到数据拥有者上传的密文后，对其进行解密处理，其解密函数为

$$\mathrm{Dec}(\mathrm{CAGG}) = \mathrm{CAGG} - \mathrm{rt}K = m_1 + m_2 + \cdots + m_v \tag{3}$$

6.2 总体验证

在可验证计算方案中，单独验证每个用户的密文会导致过多的通信量，所以需要云服务器将每个用户的密文和签名聚合，先进行总体验证工作，具体算法描述如下。

6.2.1 系统建立阶段（Setup）

设系统中一共有 V 个数据拥有者，每个数据拥有者将自己数据加密后存储在云服务器上，聚合数据 $F=\{m_1,m_2,\cdots,m_V\}$。其中，每个数据拥有者 l 的数据又分为 n 个数据块，即 $F_l=\{m_{l,1},m_{l,2},\cdots,m_{l,n}\}$。

步骤 1 密钥生成 KeyGen(1^k)。以安全参数 k 作为输入，数据拥有者 l 随机选择 2 个素数 p' 和 q'。令 $p=2p'+1$ 和 $q=2q'+1$，满足 p 和 q 也是大素数，保证 p 和 q 长度相等，计算 $N_l=pq$ 和 $\varphi(N_l)=(p-1)(q-1)$；随机选择整数 $e_l<\varphi(N_l)$，满足 $\gcd(e_l,\varphi(N_l))=1$，并计算整数 d_l 满足 $d_l\equiv e_l^{-1}(\mathrm{mod}\varphi(N_i))$，安全地销毁 p、q、$\varphi(N_l)$。选择安全的同态哈希函数 $H(\cdot)Z_{N_l}^*\to Z_{N_l}^*$。则用户的公钥 $\mathrm{pk}_l=(N_l,e_l)$，私钥 $\mathrm{sk}_l=(d_l)$。

步骤 2 签名生成 SignGen(F_l,sk_l)。数据拥有者选择数据 F_l 的标识符 $\mathrm{name}_l\in Z_{N_l}^*$，对每个 $i\in\{1,2,\cdots,n\}$，用户计算签名如下

$$\sigma_{l,i}=(H(\mathrm{name}_l\|i)H(m_{l,i}))^{d_l}\ \mathrm{mod}N_l \tag{4}$$

用 σ_l 表示签名的集合

$$\Phi_l=\{\sigma_{l,1},\sigma_{l,2},\cdots,\sigma_{l,i}\} \tag{5}$$

为了保证 name_l 的完整性，用户计算 $t_l=\mathrm{name}_l\|\mathrm{Sign}_{\mathrm{sk}_l}(\mathrm{name}_l)$ 作为文件 F_l 的标签，$\mathrm{Sign}_{\mathrm{sk}_l}(\mathrm{name}_l)$ 是在私钥 sk_l 下的签名，用户将 F_l 和 (Φ_l,t_l) 发送给云服务器。

6.2.2 挑战阶段（Challenge）

挖矿者节点检索文件标签 t_l，并使用 pk_l 验证签名 t_l，如果验证失败，则终止交易；如果验证成功，则恢复其 name_l。

假设挖矿者节点需要验证 V 个数据拥有者上传数据的完整性，其随机生成一个含有 c 个元素的子集 $I=\{s_j\}_{1\leqslant j\leqslant c}$ 且 $s_1\leqslant s_2\leqslant\cdots\leqslant s_c$，对每个 $i\in I$，云服务器随机选择 $v_i\in Z_P^*$，生成挑战消息 $\mathrm{chal}=\{i,v_i\}_{i\in I}$，并将其发送给云服务器和区块链。

6.2.3 证据生成阶段（ProofGen）

ProofGen($\{m_{l,i}\}_{i\in I},\Phi_l,\mathrm{chal},\mathrm{pk}_l$)。首先，云服务器收到挑战 $\mathrm{chal}=\{i,v_i\}_{i\in I}$，选择数据 $\{m_{l,i}\}_{i\in I}$，发送 $v_i m_{l,i}$ 给云服务器。云服务器随机选择 $r_l\in Z_p^*$，即 $\mu_l'=\sum_{i\in I}v_i m_{l,i}$，为了盲化 μ_l'，令 $\mu_l=\mu_l'+r_l$。令 $Y=H\left(\sum_{l=1}^{V}r_l\right)$，$\mu=H\left(\sum_{i=1}^{V}\mu_l\right)$。其次，云服务器计算聚合签名 $\sigma_l=\prod_{i=s_1}^{s_c}\sigma_{l,i}^{v_i}\mathrm{mod}N_l$，给挖矿者节点和区块链发送存储完整性的证据 $\{\mu,\{\sigma_l\}_{1\leqslant l\leqslant V},Y\}$。

6.2.4 数据验证阶段（Verification）

ProofVeriy(Proof,pk_l)。挖矿者节点收到云服务器发送过来的 $\mathrm{Proof}=\{\mu,\{\sigma_l\}_{1\leqslant l\leqslant V},Y\}$，并验证式(6)。

$$Y\prod_{i=1}^{V}\sigma_l^{e_l}=\mu\prod_{i=l}^{V}\left(\prod_{i=s_l}^{s_c}\left(H(\mathrm{name}_l\|i)^{v_i}\right)\mathrm{mod}N_l\right) \tag{6}$$

如果式(6)成立，挖矿者节点返回 TRUE；否则，返回 FALSE，并进行单独验证，追溯恶意用户。

为验证数据拥有者存储数据的完整性，式(6)的正确性验证如下。

$$Y\prod_{i=1}^{V}\sigma_l^{e_l}=$$

$$H\left(\sum_{l=1}^{V}r_l\right)\prod_{i=1}^{V}\left(\prod_{i=s_1}^{s_c}\left(H(\mathrm{name}_l\|i)^{v_i}\right)H(m_{l,i})^{v_i}\mathrm{mod}N_l\right)=$$

$$\prod_{i=1}^{V}\left(\prod_{i=s_1}^{s_c}\left(H(\mathrm{name}_l\|i)^{v_i}\right)\mathrm{mod}N_l\right)\prod_{i=1}^{V}\left(\prod_{i=s_1}^{s_c}H(m_{l,i})^{v_i}\right)\prod_{i=1}^{V}H(r_l)=$$

$$\prod_{i=1}^{X}\left(\prod_{i=s_1}^{s_c}\left(H(\mathrm{name}_l\|i)^{v_i}\right)\mathrm{mod}N_l\right)\prod_{i=1}^{X}\left(\prod_{i=s_1}^{s_c}H(m_{l,i})^{v_i}H(r_l)\right)=$$

$$\prod_{l=1}^{X}\left(\prod_{i=s_1}^{s_c}\left(H(\mathrm{name}_l\|i)^{v_i}\right)\mathrm{mod}N_l\right)\prod_{l=1}^{X}\left(H\left(\sum_{i=1}^{s_c}m_{l,i}v_i+r_l\right)\right)=$$

$$\mu\prod_{l=1}^{X}\left(\prod_{i=s_1}^{s_c}\left(H(\mathrm{name}_l\|i)^{v_i}\right)\mathrm{mod}N_l\right)$$

总体验证步骤如下。

1) MN 检索文件标签 t_l 并验证其签名。

2) MN 生成随机挑战 $\mathrm{chal}=\{(i,v_i)\}_{i\in I}\xrightarrow{\{(i,v_i)\}_{i\in I}}$ CS。

3) CS 对每个用户 $l(1\leqslant l\leqslant V)$ 计算 μ_l'、σ_l，选择 $r_l\in Z_p^*$。

4) CS 计算 $Y=H\left(\sum_{l=1}^{V}r_l\right)$，$\mu_l=\mu_l'+r_l$，$\mu=H\left(\sum_{i=1}^{V}\mu_l\right)$。

5) CS $\xrightarrow{\{\mu,\{\sigma_l\}_{1\leqslant l\leqslant V},Y\}}$ MN，验证式(6)。

6.3 单个验证

6.3.1 系统建立阶段（Setup）

设数据拥有者 L 的数据有 n 个数据块，为

$F'=\{m_{L,1},m_{L,2},\cdots,m_{L,n}\}$。系统首先执行以下算法生成密钥对和系统参数。

1) KeyGen(1^k)。输入安全参数 k，数据拥有者按 5.2.1 节的方法生成公钥 $\mathrm{pk}=(N,e)$，私钥 $\mathrm{sk}=d$。

2) SignGen(F',sk)。数据拥有者随机选择数据 $F'=\{m_{L,1},m_{L,2},\cdots,m_{L,n}\}$ 的标识符 $\mathrm{name}\in Z_N^*$，对每个数据块 $m_{L,n}$，$i\in\{1,2,\cdots,n\}$ 计算签名

$$\sigma_{L,i}=(H(\mathrm{name}\|i)H(m_{L,i}))^d \tag{7}$$

用 σ'_L 表示签名的集合为

$$\Phi'_L=\{\sigma_{L,1},\sigma_{L,2},\cdots,\sigma_{L,n}\} \tag{8}$$

为了保证 name 的完整性，用户计算 $t=\mathrm{name}\|\mathrm{Sign}_{\mathrm{sk}}(\mathrm{name})$ 作为文件 F' 的标签，$\mathrm{Sign}_{\mathrm{sk}}(\mathrm{name})$ 是在私钥 d 下的签名。

假设挖矿者节点知道 F' 的块数 n，用户将 F' 和 (Φ'_L,t) 发送给云服务器。

6.3.2 挑战阶段（Challenge）

挖矿者节点检索文件标签 t，并使用 pk 验证签名 t，如果验证失败，则终止交易；如果验证成功，则恢复其 name。

为检查 $F'=\{m_{L,1},m_{L,2},\cdots,m_{L,n}\}$ 的完整性，用户向挖矿者节点发送验证请求。挖矿者节点收到验证请求后，随机生成一个含有 c 个元素的子集 $I=\{s_j\}_{1\leqslant j\leqslant c}$ 且 $s_1\leqslant s_2\leqslant\cdots\leqslant s_c$，对每个 $i\in I$，挖矿者节点随机选择 $v_i\in Z_P^*$，生成挑战消息 $\mathrm{chal}=\{i,v_i\}_{i\in I}$，并将其发送给云服务器。

6.3.3 证据生成阶段（ProofGen）

ProofGen($\{m_{L,i}\}_{i\in I},\Phi'_L,\mathrm{chal},\mathrm{pk}$)。收到挑战 $\mathrm{chal}=\{i,v_i\}_{i\in I}$ 后，云服务器选择数据 $\{m_{L,i}\}_{i\in I}$，然后发送 $v_i m_{L,i}$ 给云服务器。云服务器随机选择 $r\in Z_p$，计算 $Y=H(r)$，并汇总各数据拥有者发送的数据，即 $\mu'=\sum_{i\in I}v_i m_{L,i}$。为了盲化 μ'，令 $\mu=\mu'+r$。

同时，云服务器计算聚合签名 $\sigma'_L=\prod_{i=s_1}^{s_c}\sigma_{L,i}^{v_i}\mathrm{mod}N$，然后将 $\{\mu,\sigma'_L,Y\}$ 作为存储完整性的证据发送给挖矿者节点。

6.3.4 验证阶段（Verification）

ProofVeriy(Proof,pk)。收到云服务器发送来的 $\mathrm{Proof}=\{\mu,\sigma'_L,Y\}$，挖矿者节点验证式(9)。

$$Y(\sigma'_L)^e=\prod_{i=s_1}^{s_c}\left(H(\mathrm{name}\|i)^{v_i}\right)H(\mu)\mathrm{mod}N \tag{9}$$

如果式(9)成立，挖矿者节点返回 TRUE；否则，返回 FALSE，并找出恶意用户。

为验证数据拥有者存储数据的完整性，式(9)的正确性验证如下

$$Y(\sigma'_L)^e=H(r)\left(\prod_{i=s_1}^{s_c}\sigma_{L,i}{}^{v_i}\mathrm{mod}N\right)^e=$$

$$H(r)\left(\prod_{i=s_1}^{s_c}(H(\mathrm{name}\|i)H(m_{L,i}))^{v_i d}\right)^e\mathrm{mod}N=$$

$$H(r)\left(\prod_{i=s_1}^{s_c}(H(\mathrm{name}\|i)^{v_i})\right)\left(\prod_{i=s_1}^{s_c}(H(m_{L,i}))^{v_i}\right)\mathrm{mod}N=$$

$$H(r)\prod_{i=s_1}^{s_c}(H(\mathrm{name}\|i)^{v_i})H\left(\sum_{i=s_1}^{s_c}v_i m_{L,i}\right)\mathrm{mod}N=$$

$$\prod_{i=s_1}^{s_c}(H(\mathrm{name}\|i)^{v_i})H(\mu)\mathrm{mod}N$$

单个验证步骤如下。

1) MN 检索文件标签 t 并验证其签名。

2) MN 生成随机挑战

$\mathrm{chal}=\{(i,v_i)\}_{i\in I}\xrightarrow{\{(i,v_i)\}_{i\in I}}\mathrm{CS}$。

3) CS 计算

$\mu'=\sum_{i\in I}v_i m_{L,i}$，$\sigma'_L=\prod_{i=s_1}^{s_c}\sigma_{L,i}^{v_i}\mathrm{mod}N$。

4) CS 随机选择 $r\in Z_p$，计算 $Y=H(r)$。

5) CS 计算 $\mu=\mu'+r\xrightarrow{\{\mu,\sigma'_L,Y\}}\mathrm{MN}$。

6) MN 验证式(9)。

7 安全性分析与性能评估

7.1 安全性分析

1) 内容隐私性

本文方案中数据信息存储时，同态加密的密文具有可操作性，不同的数据拥有者通过密码技术使用不同的公钥对数据信息进行同态加密处理后存储在云服务器，加强了数据的传输及存储安全性。并将聚合密文和上传者信息存储在区块中。上述操作进一步保证了共享内容的隐私性。

2) 可验证和可追溯性

现有的可验证云存储方案由于缺少可信第三方，存在参与者共谋篡改数据的安全隐患。本文方案利用区块链的去中心化为可验证计算提供了一个

可信环境，可以有效避免第三方机构或恶意用户私自篡改数据的风险。利用区块链的不可篡改性和可溯源性，对数据进行同态哈希后将签名分布式存储在区块链上，数据使用者能够在保证安全的前提下，通过挖矿者节点向区块链发出对云端数据进行双重验证的请求，最后得到审查结果，并通过签名信息对恶意用户进行追溯和惩罚。

3) 抗共谋攻击

本文提出的基于区块链的双重可验证云存储方案可以避免共谋。区块链通过密码学技术为可验证计算提供了一个可信任的环境，云服务器将数据拥有者的聚合密文结果存储在链上，有效防止了参与者之间共谋，使各个参与者诚实地执行协议。同时，被追溯到的共谋者会因违反协议被扣除高额的担保押金，而遵守合约比共谋能获得更高的利益，也消除了参与者共谋的动机。

4) 可扩展性

区块链是一个去中心化的账本，上链的信息共识生成区块都拥有时间戳可溯源且无法篡改。但由于区块存储有限，不能将所有的信息完整的存储在区块中，为了保证存储数据正确性的快速验证，本文方案将完整密文存储在云服务器中，链上区块记录上传者信息和存储聚合结果便于验证和追溯，也解决了区块存储容量有限的问题。

7.2 性能评估

1) 计算代价

本文方案中的计算代价主要由生成验证阶段和数据验证阶段产生。计算代价如表 1 所示，其中，V 为验证的数据拥有者个数，M 为在 G_1 上的乘法，E 为指数，P 为双线性映射，H 为单项哈希函数。

表 1　　计算代价

验证类型	生成验证阶段	数据验证阶段
总体验证	$2H+(2Vc-1)M+VcE$	$VcH+(Vc+V)M+(Vc+V)E$
单个验证	$1H+(2c-1)M+cE$	$(c+1)H+(c+1)M+(c+1)E$

2) 通信量

本文方案在验证过程中需传送 $\{\mu,\{\sigma_l\}_{1\leqslant l\leqslant V},Y\}$，文献[18]方案需传送 $\{\{\sigma_l,\mu_l\}_{1\leqslant l\leqslant V},R\}$，本文方案降低了云服务器与挖矿节点间的通信量。随着验证数据块数目增多，本文方案算法也大大降低了验证每块的平均通信量。

3) 验证算法的有效性

本文的验证算法基于概率模型。设被篡改了 b 块，每次验证 n 块数据中的 c 块，随机变量 X 表示检测到数据被篡改的块数，则 $P\{X\geqslant 1\}\in\left[1-\left(\frac{n-b}{n}\right)^c,1-\left(\frac{n-b-c+1}{n-c+1}\right)^c\right]$，验证一次检测到数据被篡改的概率至少为 $1-\left(\frac{n-b}{n}\right)^c$。设随机变量 Z 表示验证 a 次检测到数据被篡改的块数，则 $P\{Z\geqslant 1\}\geqslant 1-\left(\frac{b-c}{n}\right)^{ac}$，共挑战了 $w=ac$ 块，即至少有 $1-\left(\frac{b-c}{n}\right)^w$ 的概率检测到数据被篡改。

假设 n 块数据中有 0.7%的数据被篡改，即 $b=0.007n$，$1-\left(\frac{n-b}{n}\right)^w=1-0.993^w$。同理得 $b=0.01n$ 和 $b=0.09n$。验证算法的有效性如图 3 所示，$P\{Z\geqslant 1\}$ 与 n 无关。当 $c=n$，即验证每一块数据时，本文算法也可以 100%证明数据完备性。

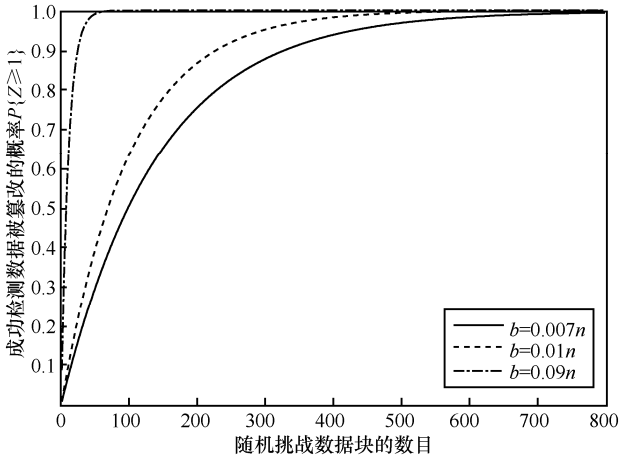

图 3　验证算法的有效性

通过对现有的可验证计算方案进行对比分析发现，文献[9, 12]方案均是基于区块链进行可验证计算研究，不需要依赖可信第三方，但都不具有可追溯性。文献[9]方案实现了个人健康档案共享过程中数据的安全性，提高了患者敏感数据的隐私性，但该方案的计算负担较高，不具有可扩展性；文献[12]提出了一种基于区块链的投票系统，系统中区块链的附加结构确保了普遍的可验证性和可扩展性，从而最大限度地减少了对可信第三方的依赖，并且适用于大规模的选举，但是不能抗共谋攻击。

文献[19-20]方案均实现了数据可验证性。Buccafurri 等[19]提出了一种在云环境中验证查询结

表 2 方案对比

方案	云存储	可验证性	隐私性	可扩展性	抗共谋	区块链技术	可追溯
文献[9]方案	×	√	√	×	—	√	×
文献[12]方案	×	√	√	√	×	√	×
文献[19]方案	√	√	√	—	×	×	×
文献[20]方案	√	√	√	×	—	×	×
文献[21]方案	√	×	√	—	—	×	×
本文方案	√	√	√	√	√	√	√

果的确定性方法，该方法实现了用户对结果的完整性的验证。数据所有者验证第一个标记的时间值，计算每个元素的 MAC 属性，将其与云服务器返回的值进行比较，以便验证从第一个标记开始的每个链接。该方案在一定程度上减小了插入空间和时间的复杂度。武朵朵等[20]提出了一种针对矩阵乘积的可验证外包计算方案。但文献[19-20]方案都不具有可扩展性和可追溯性。

文献[21]提出了一个隐私保护框架，将患者健康记录存储在云辅助环境中，其性能基于适应度、隐私性和实用性等性能指标，但不能对数据正确性进行验证并追溯恶意用户。

本文方案首先保证了数据机密性和传输安全性，实现了数据的隐私保护；其次利用区块链的不可篡改性和可溯源性，实现了云端数据完整性的双重验证，有效避免了数据被非法用户篡改的风险，并可对恶意用户进行追溯。方案对比如表 2 所示。

8 结束语

区块链可以提供极高的透明度、分布式可验证性和不可篡改性。利用区块链技术和可验证计算技术，可对上传到云端的数据的完整性进行检验，有效避免了数据被非法用户篡改的风险。基于区块链上述功能，本文方案采用同态加密技术对数据信息进行加密，保证数据机密性和传输的安全性，实现数据的隐私保护；另外，使用同态哈希函数，数据使用者能够在保证安全前提下可对云端数据进行双重验证，并可对恶意用户进行追溯。区块链的不可篡改、去中心化及匿名性等特点与可验证计算结合，可以在计费系统、医疗敏感信息共享等领域得到很好的具体应用。将区块链与可验证计算相结合，应用于具体实际场景中解决现存在的关键问题，是下一步研究工作的重点。

参考文献：

[1] YU Y, LIU S M, YEOH P L, et al. LayerChain: a hierarchical edge-cloud blockchain for large-scale low-delay industrial Internet of Things applications[J]. IEEE Transactions on Industrial Informatics, 2021, 17(7): 5077-5086.

[2] LI H T, GUO F, WANG L L, et al. A blockchain-based public auditing protocol with self-certified public keys for cloud data[J]. Security and Communication Networks, 2021, 2021: 1-10.

[3] 薛锐, 吴迎, 刘牧华, 等.可验证计算研究进展[J].中国科学:信息科学, 2015, 45(11): 1370-1388.
XUE R, WU Y, LIU M H, et al. Progress in verifiable computation[J]. Scientia Sinica (Informationis), 2015, 45(11):1370-1388.

[4] NIGAM R, PATHAK R K, KUMAR A, et al. PCP framework to expose malware in devices[C]//Proceedings of 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC). Piscataway: IEEE Press, 2020: 1-6.

[5] 杨亚涛, 赵阳, 张卷美, 等. 同态密码理论与应用进展[J]. 电子与信息学报, 2021(2): 475-487.
YANG Y T, ZHAO Y, ZHANG J M, et al. Recent development of theory and application on homomorphic encryption[J]. Journal of Electronics & Information Technology, 2021(2): 475-487.

[6] CHANG J Y, JI Y Y, SHAO B L, et al. Certificateless homomorphic signature scheme for network coding[J]. IEEE/ACM Transactions on Networking, 2020, 28(6): 2615-2628.

[7] CHAUM D, PEDERSEN T. Wallet databases with observers[C]//Proceedings of Annual International Cryptology Conference. Berlin: Springer, 1993: 89-105.

[8] LONE A H, NAAZ R. Demystifying cryptography behind blockchains and a vision for post-quantum blockchains[C]//Proceedings of 2020 IEEE International Conference for Innovation in Technology (INOCON). Piscataway: IEEE Press, 2020: 1-6.

[9] WANG S P, ZHANG D, ZHANG Y L. Blockchain-based personal health records sharing scheme with data integrity verifiable[J]. IEEE Access, 2019, 7: 102887-102901.

[10] GUO Y, ZHANG C, JIA X H. Verifiable and forward-secure encrypted search using blockchain techniques[C]//Proceedings of ICC 2020 - 2020 IEEE International Conference on Communications (ICC). Pis-

cataway: IEEE Press, 2020: 1-7.

[11] GUO R, ZHUANG C Y, SHI H X, et al. A lightweight verifiable outsourced decryption of attribute-based encryption scheme for blockchain-enabled wireless body area network in fog computing[J]. International Journal of Distributed Sensor Networks, 2020, 16(2): 155014772090679.

[12] DIMITRIOU T. Efficient, coercion-free and universally verifiable blockchain-based voting[J]. Computer Networks, 2020, 174: 107234.

[13] WANG S P, ZHANG Y L, ZHANG Y L. A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems[J]. IEEE Access, 2018, 6: 38437-38450.

[14] DORSALA M R, SASTRY V N, CHAPRAM S. Fair protocols for verifiable computations using bitcoin and Ethereum[C]//Proceedings of 2018 IEEE 11th International Conference on Cloud Computing (CLOUD). Piscataway: IEEE Press, 2018: 786-793.

[15] ZHANG Y H, DENG R H, SHU J G, et al. TKSE: trustworthy keyword search over encrypted data with two-side verifiability via blockchain[J]. IEEE Access, 2018, 6: 31077-31087.

[16] YAO H L, WANG C F, HAI B, et al. Homomorphic hash and blockchain based authentication key exchange protocol for strangers[C]// Proceedings of 2018 Sixth International Conference on Advanced Cloud and Big Data (CBD). Piscataway: IEEE Press, 2018: 243-248.

[17] REN W, TONG X, DU J, et al. Privacy-preserving using homomorphic encryption in Mobile IoT systems[J]. Computer Communications, 2021, 165: 105-111.

[18] WANG C, CHOW S S M, WANG Q, et al. Privacy-preserving public auditing for secure cloud storage[J]. IEEE Transactions on Computers, 2013, 62(2): 362-375.

[19] BUCCAFURRI F, LAX G, NICOLAZZ S, et al. Range query integrity in cloud data streams with efficient insertion[C]//International Conference on Cryptology and Network Security. Cham: Springer International Publishing, 2016: 719-724.

[20] 武朵朵, 来齐齐, 杨波. 矩阵乘积的高效可验证安全外包计算[J]. 密码学报, 2017, 4(4): 322-332.
WU D D, LAI Q Q, YANG B. Efficient, verifiable and secure outsourcing of matrix multiplication[J]. Journal of Cryptologic Research, 2017, 4(4): 322-332.

[21] SATHYA A, RAJA S K S. Privacy preservation-based access control intelligence for cloud data storage in smart healthcare infrastructure[J]. Wireless Personal Communications, 2021, 118(4): 3595-3614.

[作者简介]

冯涛（1970− ），男，甘肃临洮人，博士，兰州理工大学研究员、博士生导师，主要研究方向为网络与信息安全、区块链、工业互联网等。

孔繁琪（1994− ），女，满族，辽宁沈阳人，兰州理工大学硕士生，主要研究方向为网络与信息安全、区块链、可验证计算等。

柳春岩（1971− ），女，甘肃永靖人，兰州理工大学硕士生导师，主要研究方向为供应链管理等。

马蓉（1992− ），女，甘肃兰州人，兰州理工大学博士生，主要研究方向为工业物联网中的安全多方计算和隐私保护等。

Maher Albettar（1989− ），男，康考迪亚大学教授，主要研究方向为系统安全集成、建模、物联网和云计算。