引用格式：齐伊宁，秦宣梅，孙东红.潘鸿运，李琪，黄永峰，王丹丹.面向领域数据安全可信共享的云链融合系统[J].中国传媒大学学报（自然科学版），2022，29(02)：9-18.
文章编号：1673-4793(2022)02-0009-10

面向领域数据安全可信共享的云链融合系统

齐伊宁[1]，秦宣梅[1*]，孙东红[2]，潘鸿运[1]，李琪[2]，黄永峰[1]，王丹丹[3]

（1.清华大学电子系，北京 100084；2.清华大学网络科学与网络空间研究院，北京 100084；3.国家信息中心信息化和产业发展部，北京 100045）

摘要：针对领域大数据存储分散性的问题，引入区块链技术，提出了云计算和区块链结合的领域大数据共享管理的框架模型，设计云链融合的协同机制。在此基础上，建立了基于云链融合机制的共享数据标识编码与解析方法，实现分散数据的统一标识和定位寻址；提出轻量级加密访问控制方法，结合属性基加密机制实现领域数据的“可控可计量”共享访问；提出数据完整性审计方法，实现共享数据的完整性和可用性验证，提高领域数据的共享可信性。根据云链融合的领域大数据的共享管理框架模型以及相关技术，研发了一套面向分散领域大数据的安全可信共享软件系统，对上述模型和方法进行了有效性的验证和性能分析。该系统可以进一步扩展应用到涉及多个云环境的智慧城市评价、产业链协同等行业。

关键词：领域大数据；云计算；区块链；云链融合；数据共享
中图分类号：TP39　　**文献标识码：**A

Cloud-blockchain fusion system for secure and trusted sharing of domain data

QI Yining[1], QIN Xuanmei[1*], SUN Donghong[2], PAN Hongyun[1], LI Qi[2], HUANG Yongfeng[1], WANG Dandan[3]

(1. Department of Electronic Engineering, Tsinghua University, Beijing 100084, China;
2. Institute for Network Science and Cyberspace, Tsinghua University, Beijing 100084, China;
3. Informatization and Industry Development Department, State Information Center, Beijing 100045, China)

Abstract: At present, various domains such as smart city, industrial chain have recorded data of every business stage in different time and space. Big data in these domains is characterized by life cycle, information correlation and content integrity, which make the domain data form a typical chain and has high value. However, the storage of domain data is highly decentralized, which greatly affects the mining and utilization of domain data. To solve this problem, this paper introduces blockchain technology, proposes a framework model combining cloud computing and blockchain for sharing and management of domain data, and designs a collaborative scheme of cloud-blockchain fusion. On this basis, a shared data identification encoding and parsing method based on cloud-blockchain fusion scheme is established to realize unified identification and location addressing of dispersed data. Then, a lightweight access control method combined with attribute-based encryption mechanism is introduced to realize controllable and quantifiable sharing and access of domain data. Afterwards, a data integrity audit method is proposed to

基金项目：国家重点研发计划专项（2018YFB2101501）；国家自然基金委联合基金重点项目（U1836204）；工信部工业互联网创新发展工程专项（TC200H02Y和TC200H02X）

作者简介(*为通讯作者)：齐伊宁（1990-），女，博士研究生，主要从事数据完整性审计和区块链等技术的研究。Email：qyn18@mails.tsinghua.edu.cn
秦宣梅（1993-），女，博士，助理研究员，主要从事区块链和数据安全等技术的研究。Email：qxm17@ tsinghua.org.cn

verify the integrity and availability of shared data, thus improving the credibility of shared domain data. According to the domain big data sharing framework model and related methods of cloud-blockchain fusion scheme, a set of secure and trusted shareware system for decentralized domain big data is developed, and the validity of the above model and methods is verified and performance analysis is conducted. The system can be further extended and applied to industries involving multiple cloud environments such as smart city evaluation and industrial chain collaboration.

Keywords: domain data; cloud computing; blockchain; cloud-blockchain fusion technology; data sharing

1 引言

随着物联网、云计算和数字化技术发展,人类活动、企业生产、社会交往和国家安全等都被精细记录,产生越来越多的数据,形成了各行各业的领域大数据。领域大数据从产生、采集、存储、传输、处理、使用到销毁的全生命周期流转过程中具有明显的关联性[1],这种相互关联领域大数据也称为链条数据。例如,一个城市的智慧城市评价数据涉及到众多业务环节,包括基层采集的数据、分析处理数据、执法数据和决策数据等,这些数据由于管理部门的不同,经常是分散存储管理,有边端存储有云端存储,还有的在上级垂管部门端存储。又例如,食品安全领域大数据记录了某食品从农田到餐桌的整个生命周期中不同阶段的数据,而且,这些不同阶段数据分别存储在不同加工环节的生产企业、不同流通环节的商家企业和不同食品质检环节的管理机构等。如果要实现食品安全的全生命周期监管,就必须整合这些分散存储在不同组织的数据,获得食品安全全生命周期数据的全链条。因此,当前各个领域都记录了领域在不同时空下发生的各个业务环节数据,这些领域大数据具有生命周期性、信息相关性和内容完整性等特征,这些特征使得领域数据具有典型的链条性,以及具有极高利用价值。

由于数据采集和存储等技术、以及数据权属管理等限制,使得领域链条数据具有高度的分散性和低可用性特点。链条数据高分散性是指数据在存储上归属不同平台,在管理上隶属不同机构,在资产上权属不同组织。链条数据低可用性是指相对其本身的高价值,数据分散后隔断了数据相关性和完整性、丢失了许多维度信息,不能充分发挥链条数据的真正价值。因此,如何实现领域链条数据共享是挖掘数据资产核心价值亟待解决的前沿课题。

要有效的实现链条数据共享面临如下3方面的关键技术问题。(1)数据资源确权和可控可计量访问问题。领域数据的高价值性使得数据即资产的观念日益深入人心,数据有效共享需要建立健全数据流通交易规则和技术,在确保数据安全前提下,实现数据授权利用、可控访问以及可计量使用。(2)数据安全与隐私保护问题。数据共享使得数据可能离开用户控制域、数据的安全和隐私等问题寄托于数据使用者的诚信,因此,数据所有者和管理者因为害怕风险和不安全问题不愿意开放数据使用。(3)数据可用性审计问题。数据即使被授权给消费者合法使用,但数据的使用者无法有效判断共享数据是否完整和可用,也会降低数据使用意愿[2]。

由于上述3方面的关键技术问题存在,严重阻碍了领域大数据的安全共享、可信利用和高效挖掘。针对上述问题,本文综合云计算和区块链技术的特点,提出了一种云计算与区块链相融合的领域数据共享平台架构,建立了“原始数据不出域、数据可用不可见”的安全可信共享模型,并结合典型应用场景,研发了相应的应用系统,对云链融合的共享架构模型和技术进行了验证和性能分析。

2 面向领域数据共享的云链融合模型

云计算是采用虚拟化和按需服务等技术实现资源(包括计算、存储、软件和数据等资源)集中式服务模式。在传统的云计算架构中,通常以云端作为大型数据中心和计算中心[3]。但随着第五代移动通信(5G)的到来和物联网(IoT)技术的发展,面对带宽消耗、网络延迟、数据隐私性保护等挑战[4],中心化的云端只处理计算资源需求大、实时性要求不高的计算任务,于是出现了云边端协同的边缘计算架构,边缘计算在一定程度上缓解了带宽和实时性问题[5]。但其数据分布在不同管理域,云边端之间的组织架构不同,给分散的链条数据的共享方式和可信计算带来一些挑战。

区块链(Blockchain)是一种去中心化、不可篡改、可

追溯、多方共同维护的分布式数据库，能够将传统单方维护的仅涉及自己业务的多个孤立数据库整合在一起，分布式地存储在多方共同维护的多个节点上[6]。区块链通过集成对等网络（P2P）协议、非对称加密、共识机制、块链结构等多种技术，解决了数据可信问题[7]。通过运用区块链，无需借助任何第三方可信机构，互不了解、互不信任的多方可实现可信、对等的价值传输[8]。

基于云计算与区块链各自技术特点和优势，本文设计了一种多云（包括云边端）与区块链融合系统的体系架构，实现领域链条数据的安全可信共享。云链融合系统结构如图 1 所示。总体架构划分为 3 层：客户端、区块链及云计算平台；系统通过设计 9种接口和相应接口元语数据来实现客户端-云-链之间的协同和交互。

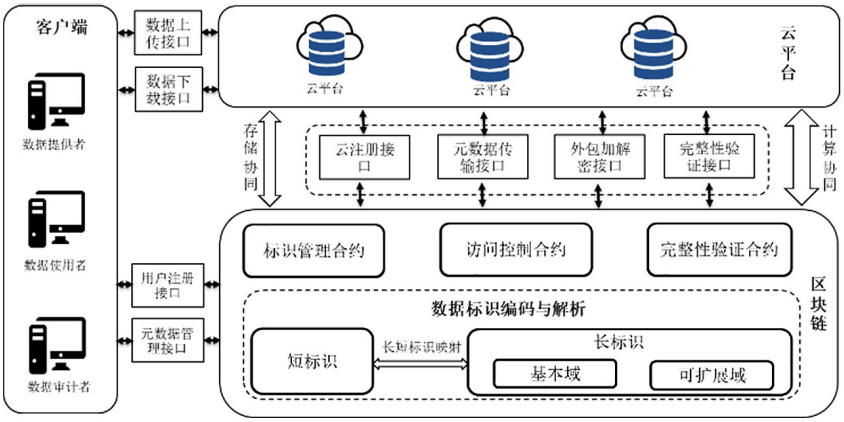


图 1　面向领域数据共享的云链融合系统模型

云计算平台：领域链条数据分散存储在不同机构的云计算平台中。通过云-链协同计算和存储接口承担计算可信度要求高的计算任务和存储管理领域原始数据。

客户端：客户端包括数据提供者、消费者和审计者等。数据提供者通过客户端对数据进行上传、更新、共享等操作，首先提供共享数据密文到云，制定访问策略，计算中间密钥密文，生成访问元数据上传到区块链。数据使用者从区块链获得授权，从云获得数据密文进行解密。为了验证数据的完整性，数据审计者为所要求审计的数据块分别产生相应的随机数，打包为挑战请求上传到区块链进行数据完整性审计。

区块链：不同云计算平台和客户端都是组成区块链的节点，这些节点共同存储、管理和维护云链融合系统的领域大数据。区块链智能合约根据制定的策略抽取备份元数据，编码成区块链全局唯一标识存储在状态数据库中，以便对共享的领域数据进行寻址，区块链中部署的属性令牌管理合约、外包加密合约以及预解密合约负责收集用户属性集、验证用户的属性是否满足属性策略，以及进行外包加密和预解密。最后通过完整性验证合约从分布式账本中提取被审计数据块的元数据，然后与云平台提交的完整性证明做双线性对验证，如果验证通过，则证明数据完整；否则不完整。

云链融合协同机制：云计算与区块链的协同机制是两者融合的基础。本文从3个层面设计了两者协同机制，如图 2 所示，包括：计算协同、存储协同和交互协同等。为此分别设计客户端-云接口、客户端-链接口、云-链接口等 3 类接口方法来实现云链协同机制。具体来说，客户端-云接口包括密文数据上传接口和密文数据下载接口，客户端-链接口包括用户注册接口和元数据管理接口，云-链接口包括云注册接口、元数据传输接口、外包加解密计算接口和完整性元数据传输接口。

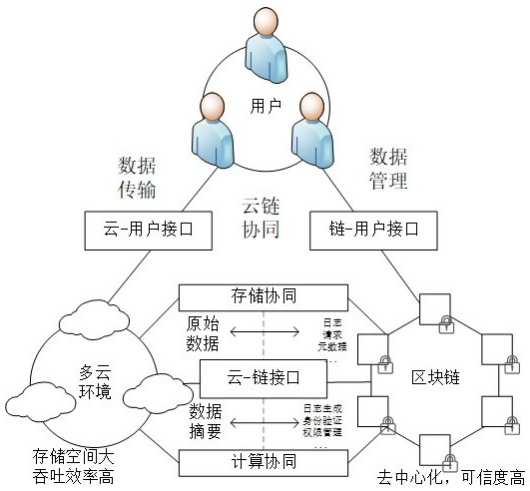


图 2　云链融合机制及其接口方式

云链存储协同对云链融合架构下的数据进行了分工存储、合作管理。其中,云平台以文件的形式存储用户的具体数据,而区块链只存储用于数据管理的元数据、用户请求等抽象数据。云链存储协同方法还对日志进行了分级管理,其中,记录用户对数据进行读写操作的操作日志存储在区块链中,而记录用户对数据进行查询操作的查询日志存储在用户本地。通过云链存储协同方法,注册在区块链中的管理元数据对分布在多云环境中的原始数据等进行操作管理和安全防护,重点保障了云上数据和链上记录的一致性。

云链计算协同对云链融合架构下的计算任务进行了分工。其中,云平台使用哈希算法为用户上传的数据计算数据摘要,作为数据标识;区块链承担了数据查找、日志生成、身份验证和权限管理等计算开销小但可信度要求高的计算任务。由于每个数据读写操作都对应一个区块链中的交易,因此其数据操作日志生成的效率与区块链交易的吞吐量相关,在海量高速大数据的环境中建议采用性能较好的联盟区块链如Hyperledger Fabric进行部署,以满足大多数场景的需求。云链计算协同方法还利用区块链的事件监听机制,在云端监听区块链中的用户请求/元数据是否正确,以确定云中的操作是否可以执行。云链协同计算提高了云计算的安全性,也为区块链减轻了计算负担。

云链交互协同是云链融合架构中的标准化数据管理接口,为用户和云平台参与云链协同的存储和计算提供了支撑。云链协同接口包括云平台和用户之间的接口("云-用户接口")、区块链和用户之间的接口("链-用户接口")和云平台和区块链之间的接口("云-链接口")。其中,按照接口中封装的底层函数的不同,"云-用户接口"又可分为数据传输接口和传输建立接口两个子类型,"链-用户接口"和"云-链接口"又可分为交易接口、事件接口和监听接口三个子类型。

3 云链融合机制下的领域数据安全可信共享系统的实现

基于上述云链融合机制,本文采用Hyperledger Fabric联盟链和Hadoop等开源代码,实现了一套面向领域数据安全可信共享的云链融合系统。在系统中,重点设计了云链融合的共享数据标识编码与解析协议,云链融合机制下的访问控制方法和数据完整性审计方法。

3.1 共享数据的标识编码与解析协议

在云链融合机制下,为了统一管理存储在不同云上的共享数据,首先需要制定一套标识编码规则,通过标识完成对共享数据的唯一标记、对共享数据信息的记录和维护。相应地,为了实现共享数据寻址,需要制定一套标识解析方法,按照统一的解析步骤完成对标识的解析,最终定位共享数据在云端的存储位置。

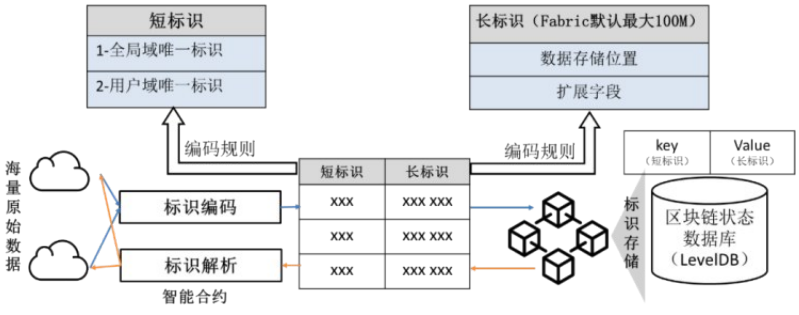


图3 云链融合的共享数据标识编码结构

共享数据的标识编码为二级结构,如图3所示,包含短标识和长标识。以短标识为键、长标识为值生成键值对,存储在Fabric状态数据库中。短标识包括全局域唯一标识和用户域唯一标识两个部分,用户域唯一标识用于增加命名自主性,构建的短标识用于在区块链网络中对数据进行唯一标记。长标识为描述数据的所有元数据的合集,基本的元数据包括:数据在云中存储位置(URL)、扩展字段。其中,扩展字段可为空,可以方便用户进行扩展。由于长标识中的字段都可以更新,当数据物理位置发生变化时,只需要更新长标识中存储的URL,就能实现用原有的短标识对新的物理位置进行寻址,使得标识编码具有数据物理位置可迁移的特性。长短标识的键-值对存储形式为快速寻址提供了保障。

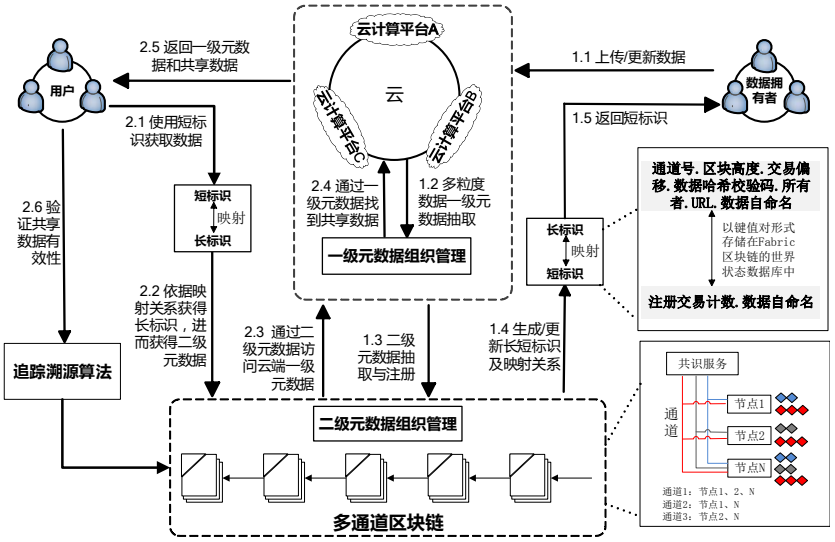


图 4 共享数据的标识编码与解析协议

共享数据标识注册和解析协议流程如图 4 所示。标识注册者发起注册请求，区块链依据标识编码规则，自动编码生成全局唯一的二级标识，将短标识返回给注册者。当标识注册者使用短标识为参数，向区块链发起数据查询请求，智能合约对二级标识分级解析，返回长标识用于对云端共享数据的寻址。

3.2　云链融合机制下的访问控制方法

为了实现领域数据的可控可计量的安全共享，本文提出一种云链融合机制下的轻量级加解密的属性基加密访问控制方法。具体方法是：利用区块链承担外包计算任务，保证计算结果的可靠性；基于提出的外包加密算法，数据所有者端只需轻量级加密计算就可以加密生成密文；基于提出的外包解密算法，数据使用者端只需轻量级解密计算就可以解密获得明文。根据上述实现原理，论文提出了云链融合机制下的访问控制协议。

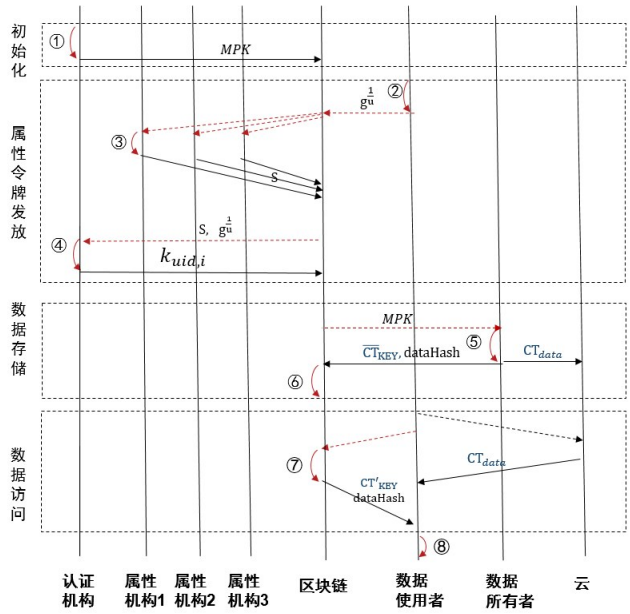


图 5　云链融合机制下的访问控制协议

访问协议流程如图 5 所示，整个协议分为 4 个阶段：系统初始化、基于智能合约的属性令牌发放、云链融合机制下的共享数据加密存储和基于智能合约的数据解密访问。每个阶段分为链上和链下两个部分计算，表示

为 Onchain.[alg_name]和 Offchain.[alg_name],其中 alg_name 是定义的算法名称。具体协议过程如下。

(1)系统初始化

该阶段通过部署智能合约和设置一组参数(MSK,MPK)来初始化系统。链下生成参数,将公共参数上传至区块链。

第1步:证书机构(Certificate Authority, CA)首先部署初始化合约、属性令牌管理合约、外包加密合约以及预解密合约到区块链。CA 链下执行 Offchain.Setup(1^{λ})算法,随机选取 $a_1,a_2,b_1,b_2 \in \mathbb{Z}_p^*$, $d_1,d_2,d_3 \in \mathbb{Z}_p^*$ 生成公共参数 MPK = $\left(h,g,H_1 = h^{a_1},H_2 = h^{a_2},T_1 = e(g,h)^{d_1a_1 + d_3},T_2 = e(g,h)^{d_2a_2 + d_3}\right)$,主密钥 MSK = $(a_1,a_2,b_1,b_2,d_1,d_2,d_3)$。

第2步:提交 SetPubKey_tx 交易请求设置公共参数,交易格式如公式(1):

$$\text{Onchain.SetPubKey\_tx} = (\text{'SetPubKey'}, \text{MPK}) \quad (1)$$

其中,SetPubKey 表示智能合约中定义的公共参数设置函数的函数名,MPK 为公共参数。网络上的任意节点可以通过调用合约,触发 GetPubKeyTx 交易查询公共参数。

(2)属性令牌发放

用户节点加入到区块链网络,首先选取 $u \in \mathbb{Z}_p^*$ 作为密钥,预先生成一个共享密钥 $g^{\frac{1}{u}}$,向区块链发起属性令牌请求。不同属性机构 AAs 收到用户请求,在一段时间内,通过智能合约发放各自管理域内的属性,生成用户属性集 S。用户属性包括:类型,地域编号,时间,位置等。

证书机构 CA 从区块链获取用户公钥 upk 和属性集 S,生成用户属性令牌上传至区块链。

第1步:证书机构 CA 链下执行 Offchain.GenToken(MSK,S,$g^{\frac{1}{u}}$)算法生成属性令牌 k。该算法输入主密钥 MSK,用户属性集 S 和用户公钥 $g^{\frac{1}{u}}$,输出用户的属性令牌 k。属性机构 AA 计算属性令牌的过程如下:

选取随机数 $r_1,r_2 \in \mathbb{Z}_p^*$,按公式(2)计算:

$$k_0 = \left(h^{b_1r_1},h^{b_2r_2},h^{r_1 + r_2}\right) \quad (2)$$

$\forall y \in S$,选取 $\sigma_y \in \mathbb{Z}_p^*$,计算 $k_y = (k_{y,1},k_{y,2},g^{-\sigma_y})$,其中,$k_{y,1},k_{y,2}$ 的具体计算如公式(3):

$$k_{y,t \in \{1,2\}} = \mathcal{H}(y1t)^{\frac{b_1r_1}{a_i}} \cdot \mathcal{H}(y2t)^{\frac{b_2r_2}{a_i}} \cdot \mathcal{H}(y3t)^{\frac{r_1 + r_2}{a_i}} \cdot g^{\frac{\sigma_y}{a_i}} \quad (3)$$

然后选取 $\sigma' \in \mathbb{Z}_p^*$,计算 $k' = (k_1',k_2',k_3')$,具体计算如公式(4)和(5):

$$k'_{t \in \{1,2\}} = g^{\frac{1}{u} \cdot d_i} \cdot \mathcal{H}(011t)^{\frac{b_1r_1}{a_i}} \cdot \mathcal{H}(012t)^{\frac{b_2r_2}{a_i}} \cdot \mathcal{H}(013t)^{\frac{r_1 + r_2}{a_i}} \cdot g^{\frac{\sigma'}{a_i}}, \quad (4)$$

$$k_3' = g^{d_3} \cdot g^{-\sigma'} \quad (5)$$

则属性机构 *AA* 生成的用户 *uid* 的属性令牌为 $k = (k_0,\{k_y\}_{y \in S},k')$。

第2步:链上发起 SetToken_tx 交易请求设置用户属性令牌,可以表示为下式(6):

$$\text{Onchain.SetToken\_tx} = (\text{'SetToken'}, \text{UserId}, \text{k}) \quad (6)$$

其中,SetToken 表示智能合约中定义的属性令牌设置函数的函数名,UserId 表示用户身份标识,k 表示用户的属性令牌。可以特别指定 SetToken 函数只能由 CA 调用。交易执行完成后,属性授权过程就会被安全地记录在区块链上。

(3)共享数据加密存储

本文采用混合加密方法,即利用公共参数对原数据进行对称加密,利用属性基加密机制加密对称密钥,然后,将生成的数据密文上传至云,密钥密文上传至区块链。密钥密文的生成和上链过程具体描述如下:

第1步:数据提供者发起公钥获取交易 GetPubKey_tx,调用初始化合约的 GetPubKey 函数,从区块链获取公共参数 MPK。然后,执行 Offchain.PreEncrypt(MPK,(M,π),KEY)算法对对称密钥 KEY 进行属性加密。该算法输入公共参数 MPK,属性访问结构(M,π)和要加密的对称密钥 KEY,输出中间密钥密文 $\overline{CT}_{KEY}$。具体计算过程如下:

随机选取 $s_1,s_2,k \in \mathbb{Z}_p^*$,计算 $ct_0 = \left(H_1^{s_1 - k},H_2^{s_2 - k},h^{s_1 + s_2},H_1^{s_1},H_2^{s_2}\right)$, $ct' = T_1^{s_1} \cdot T_2^{s_2} \cdot KEY$, $param = \left(s_1 - k,^{\circ}s_2 - k\right)$。假设 M 是 m × n 的矩阵,函数 π 将 M 的每一行映射到属性,$\text{I} = \left\{\text{i} \middle| \text{i} \in \{1,...,m\},\pi(\text{i}) \in S\right\}$ 表示矩阵 M 中属性集合 S 表示的行。中间密钥密文生成为 $\overline{CT}_{KEY} = \left(ct_0,ct',M_{ij},param\right)$。

第2步:数据提供者发起外包加密交易,表示为 $\text{Onchain.OutEnc\_tx} = (\text{'OutEnc'}, \text{Index}, \overline{\text{CT}}_{\text{KEY}}, \text{dataHash})$,其中,OutEnc 表示智能合约中定义的外包加密算法的函数名,Index 表示元数据索引,$\overline{CT}_{KEY}$ 表示中间密钥密文,dataHash 表示原数据密文哈希值,该交易生成最终密钥密文 CT_{KEY} 作为元数据记录到区块链上。最终

密钥密文 CT_{KEY} 的具体计算过程如下：

令 $CT_{KEY}=(ct_0,ct_1,\ldots,ct_m,ct')$，其中，$ct_{i\in I}=(ct_{i,1},ct_{i,2},ct_{i,3})$，具体的计算如公式(7)：

$$ct_{i,l\in\{1,2,3\}}=\mathcal{H}(\pi(i)l1)^{s_1-k}\cdot\mathcal{H}(\pi(i)l2)^{s_2-k}\cdot\prod_{j=1}^{n}\left[\mathcal{H}(0jl1)^{s_1-k}\cdot\mathcal{H}(0jl2)^{s_2-k}\right]^{M_{i,j}} \quad (7)$$

其中，$M_{i,j}$ 表示矩阵 M 的第(i,j)个元素。

最后，输出密钥密文 $CT_{KEY}=(ct_0,ct_1,\ldots,ct_m,ct')$，记录在区块链上。

(4)共享数据解密访问

数据使用者访问数据，发起访问交易，从区块链获取中间密文，链下执行最终解密。

第 1 步：链上预解密交易格式为 Onchain.PreDec_tx = ('PreDec', Index,UserId)，其中，PreDec 表示智能合约中定义的预解密算法的函数名，Index 表示元数据索引，UserId 表示用户身份标识。用户发起 PreDec_tx 交易请求，触发预解密合约中的 PreDec 算法，该算法首先获取预加密合约生成的管理元数据(CT_{KEY},dataHash)，从属性令牌管理合约获取用户属性令牌 k，然后执行预解密，生成中间密文 CT'_{KEY}。

如果和属性集 S 相关的属性令牌 k 满足 CT_{KEY} 中的策略(M,π)，则存在常数集$\{\gamma_i\}_{i\in I}$对矩阵 M 的第 i 行，满足$\sum_{i\in I}\gamma_i M_i=(1,0,\ldots,0)$。然后，按照公式(8)-(10)计算：

$$num=ct'\cdot e\left(\prod_{i\in I}ct_{i,1}^{\gamma_i},k_{0,1}\right)\cdot e\left(\prod_{i\in I}ct_{i,2}^{\gamma_i},k_{0,2}\right)\cdot e\left(\prod_{i\in I}ct_{i,3}^{\gamma_i},k_{0,3}\right) \quad (8)$$

$$den=e\left(k_1'\cdot\prod_{i\in I}k_{\pi(i),1}^{\gamma_i},ct_{0,1}\right)\cdot e\left(k_2'\cdot\prod_{i\in I}k_{\pi(i),2}^{\gamma_i},ct_{0,2}\right)\cdot e\left(k_3'\cdot\prod_{i\in I}k_{\pi(i),3}^{\gamma_i},ct_{0,3}\right) \quad (9)$$

$$mid_{KEY}=\frac{num}{den} \quad (10)$$

生成的 mid_{KEY} 为中间密文。该交易执行完成后，被作为访问记录存在区块链上。这样，用户的访问过程被记录在区块链上，为数据所有者生成访问报告。

第 2 步：数据使用者链下执行最终解密，从区块链获得返回的中间密文结果，利用私钥 u 解密对称密钥 KEY。对称密钥可以按公式(11)计算：

$$ct'\cdot mid_{KEY}{}^{u}=KEY \quad (11)$$

最后，用对称密钥 KEY 解密出原数据，dataHash 用于验证云提供的原数据的完整性。

通过以上 4 个步骤，在云链融合系统中，在云上存储原数据密文，基于智能合约进行外包加密和预解密计算，数据所有者端只需轻量级加密计算就可以加密生成密文，同时，数据使用者端只需轻量级解密计算就可以解密获得明文，从而实现了基于智能合约的共享数据轻量级访问控制方法。同时，通过不同粒度(如数据块、对象和文件等)共享数据加密和访问控制策略，就可以实现共享的可计量访问。

3.3　共享数据的完整性审计方法

验证共享数据可用性和完整性是云链融合共享系统的数据可信利用基础。本文提出了一种云链融合机制下的云数据完整性审计方法。该方法有别传统可信第三方审计模型[10]，将云数据完整性审计的整个流程转移到区块链上通过智能合约执行，提升数据审计结果可信和过程效率。基本原理是采用智能合约的背书机制防止个别第三方审计者篡改审计结果，能够有效利用区块链的不可篡改性抵抗过去的审计方法所无力抵抗的合谋攻击；将审计中起到关键性作用的审计凭据元数据存储在区块链分布式账本中，能够合理利用分布式账本的可靠性解决被用作审计凭据的元数据的可信性问题；设计了云数据完整性审计记录在区块链中的存储形式，形成不可篡改的审计日志。云链融合机制下的云数据安全审计的协议过程设计如下 4 个阶段。(说明：下文中 $\mathbb{G}_1,\mathbb{G}_2$ 代表具有素数阶 p 的乘法循环群，g 为 $\mathbb{G}_2$ 的生成元，$H(\cdot)$ 为抗碰撞加密哈希函数)。

(1)准备阶段

第 1 步，密钥生成算法：$KeyGen(1^{\lambda})\rightarrow(pk,sk)$

用户根据安全参数 λ 获取系统中使用的公私钥对，λ 为所使用的安全素数长度。用户随机选择元素 $x\leftarrow Z_p,u\leftarrow\mathbb{G}_1$，计算 $w\leftarrow u^x,v\leftarrow g^x$。用户私钥为 $sk=(x)$，公钥为 $pk=(u,v,w,g)$。

第 2 步，标签生成算法：$TagGen(sk,pk,F)\rightarrow\Phi$

用来对用户数据文件进行分块，并生成元数据。给定用户文件 F，将用户数据文件分为 n 块，表示为 $F=\{m_1,m_2,\ldots,m_n\}$，对每一数据块可按公式(12)计算其标签：

$$\sigma_i\leftarrow\left(u^{m_i}\right)^x \quad (12)$$

生成 $\Phi=\{\sigma_i\}_{1\le i\le n}$ 作为元数据。

(2)挑战阶段

挑战生成算法：$GenChal(F)\rightarrow chal$

审计者从 F 的 n 个数据块中选择索引子集 $I=\{i_1,i_2,\ldots,i_c\}$，对子集中的每个数据块索引 $i\in I$，选择一

个随机系数$\lambda_i \leftarrow \mathbb{Z}_p$,构建挑战信息$chal = \left\{\left(i,\lambda_i\right)\right\}_{i \in I}$。

(3)响应阶段

完整性证明生成算法:$ProofGen(chal, F, \Phi, pk) \rightarrow P$

云服务器收到挑战信息$chal$后,随机选择系数$r \leftarrow Z_p$,并计算公式(13)和(14):

$$\mu = \sum_{i \in I} \lambda_i m_i + r \tag{13}$$

$$\sigma = w^r \tag{14}$$

最终得到完整证明$P = \{\mu,\sigma\}$并将结果以交易提案形式发回审计者。

(4)证明阶段

审计验证过程:$ProofCheck(P,\Phi) \rightarrow \{true,false\}$

审计者从区块链账本中提取元数据$\Phi = \left\{\sigma_i\right\}_{1 \le i \le n}$,并进行双线性对检验$e\left(\sigma \cdot \prod_{i \in I} \sigma_i^{\lambda_i}, g\right) = e\left(u^{\mu}, v\right)$,那么根据双线性对的原理,验证通过,背书结果输出true,反之输出false。

4 系统的性能测试与分析

根据本文提出的云链融合的共享数据系统模型和实现技术,研发了一套基于Hyperledger Fabric和公共云平台的实验原型系统,该系统通过Hyperledger Fabric区块链智能合约为用户提供安全可信的共享数据管理服务,包括:用户管理、数据存储、数据查询、数据共享、数据审计、属性基加密访问控制等。下面测试分析了该原型系统在共享数据注册和检索、访问控制、完整性审计方面的性能。

4.1 实验环境和方法

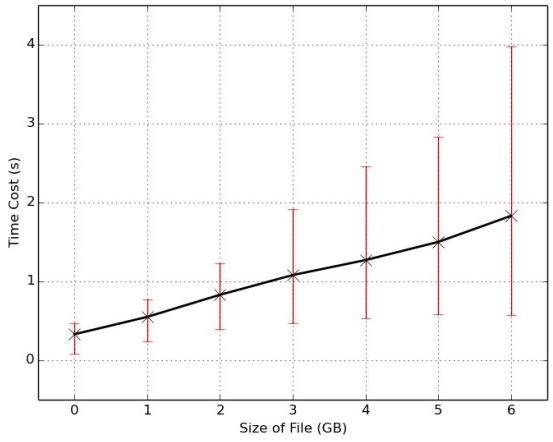
实验选择阿里和亚马逊云等公共云为多源数据共享环境,区块链智能合约的测试使用Hyperledger Caliper[11],每项性能测试重复100次取平均值;链下程序的测试使用测试代码进行,每项测试重复10次取平均值。

4.2 共享数据注册效率

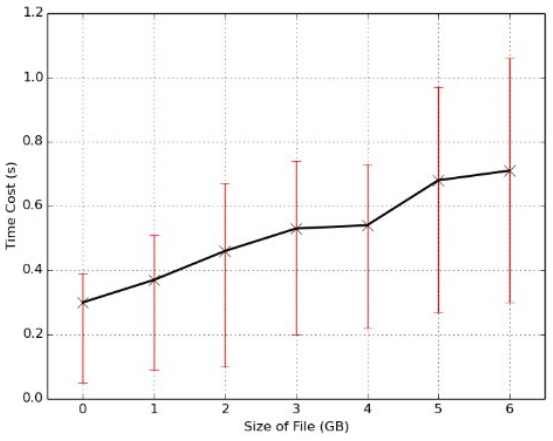
共享数据注册指不同云计算的拟定共享数据通过智能合约在云链融合系统的区块链链表中登记共享数据元信息的过程。实验测试系统中共享数据注册的效率,图6(a)中横坐标为数据大小,纵坐标为时间开销。黑色"x"点为平均时间,上下的红色"-"点表示最大值和最小值。实验结果显示,共享数据注册的时间开销与数据大小呈正相关。由于审计过程需要将数据分块,并将数据块标签存在区块链上,随着数据量的增加,数据长记录中的数据标签会增长,因此使得整个数据记录的长度增加,从而需要更多的时间开销。平均每增大1GB数据,数据注册的时间开销会增加0.25秒。由于区块链中交易排序存在随机性以及节点间通信时存在网络波动,相同大小数据发布的时间开销在一定范围内波动,并随着交易大小的增加波动的范围有所增加。

4.3 共享数据检索性能

共享数据检索指智能合约在区块链上查询数据记录。实验测试平台中数据查询的效率。图6(b)中横坐标为数据大小,纵坐标为时间开销。黑色"x"点为平均时间,上下的红色"-"点表示最大值和最小值。实验结果显示,数据查询的时间开销与数据大小呈正相关。随着数据量的增加,数据长记录中的数据标签会增长,因此使得整个数据记录的长度增加,因此需要更多的时间开销。平均每增大1GB数据,数据查询的时间开销会增加0.07秒。



(a)共享数据注册时间



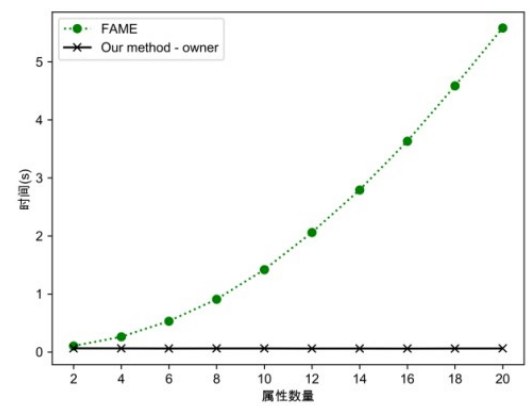
(b)共享数据检索性能

图6 共享数据注册和检索性能测试

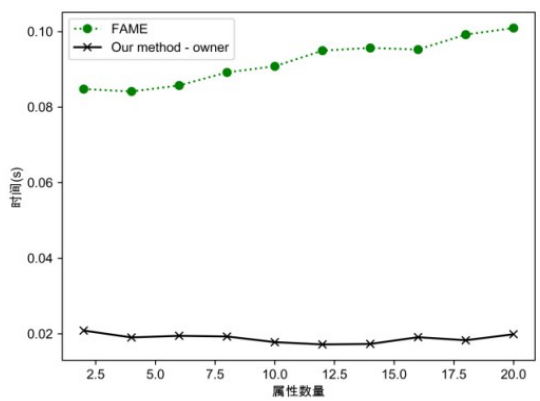
4.4　共享数据访问控制性能

数据访问控制方法主要划分为预加密、外包加密、预解密、最终解密4个计算任务，其中外包加密和预解密计算由智能合约完成，预加密和最终解密由客户端本地完成。为了评估轻量级加解密性能，实验对比FAME[9]方法测试客户端加解密效率。FAME是一种快速的属性基加密方案。提出的访问控制方法在云链融合的架构下基于区块链实现了加解密外包计算。实验采用256-bit BN椭圆曲线生成密钥，基于gofe[12]实现FAME方法。类似于FAME方法，实验中的加密算法采用最复杂的访问策略，即表示为 "$attr_1$ AND $attr_2$ AND ⋯ AND $attr_n$"，这样就需要具有所有n个属性才能解密。

实验首先测试了不同属性数目下客户端链下加解密计算性能，如图7(a)所示，随着属性数目从2开始线性增长至20，FAME方法中客户端的加密时间呈线性增长趋势，提出的方法客户端加密时间基本维持一个常数。解密阶段客户端的计算效率如图7(b)所示，FAME方法和提出方法的解密时间与属性数目无关，提出方法的解密时间约为18.8 ms。由于提出方法基于区块链进行外包计算，有效降低了链下客户端的加解密计算复杂度。



(a)预加密过程响应时间



(b)最终解密过程响应时间

图7　共享数据访问控制的链下计算性能

为了测试交易和智能合约的性能，实验采用区块链性能测试工具Caliper测试区块链交易延迟性能指标。图8展示了并发交易数目固定时属性数量对交易的平均响应时间的影响。属性数量从2增长到20，交易响应时间随着交易数量的增长呈现线性增长趋势。由于预加密函数计算任务资源消耗最大，其平均响应时间最长。虽然共识机制会导致外包计算产生一定的响应延迟，但文献[13]表明在主流的部署配置中，Fabric可以实现超过每秒3500个交易的吞吐量，交易延迟低于1秒，可以很好地扩展到100个以上对等节点。

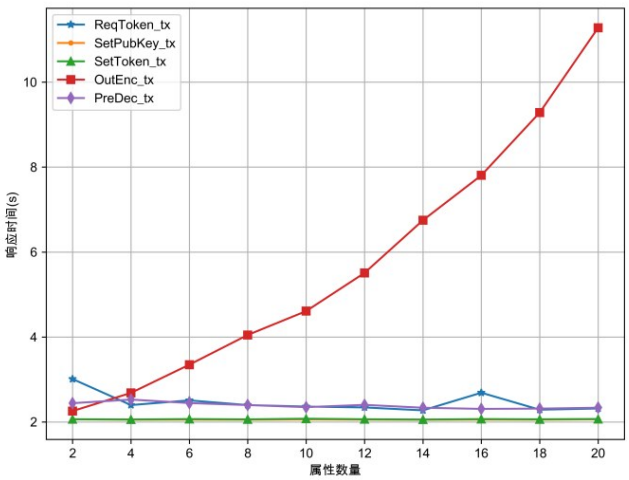


图8　共享数据访问控制的链上计算性能

4.5　共享数据审计性能

数据审计过程包括生成挑战、生成证明和完整性验证3个子过程。在云链融合的数据共享系统中，生成挑战和完整性验证由区块链的智能合约完成，生成证明由云服务完成。实验测试系统的共享数据审计效率。图9中横坐标为审计的数据块数，纵坐标为时间开销。蓝色“x”点表示生成挑战的时间开销，绿色圆点表示生成证明的时间开销，黑色“*”点表示完整性验证的时间开销。实验结果显示，随着数据审计块数的增加，3个子过程操作的时间开销都有所增加。但是由于3个子过程的计算复杂度不同，时间开销增长的速度有所不同。其中生成挑战部分计算复杂度最低，因此时间开销相对稳定；完整性验证部分计算复杂度最高，且在区块链上实现，因此时间开销的增长最为明显，且相对其他二者，其时间开销的增长较不稳定。

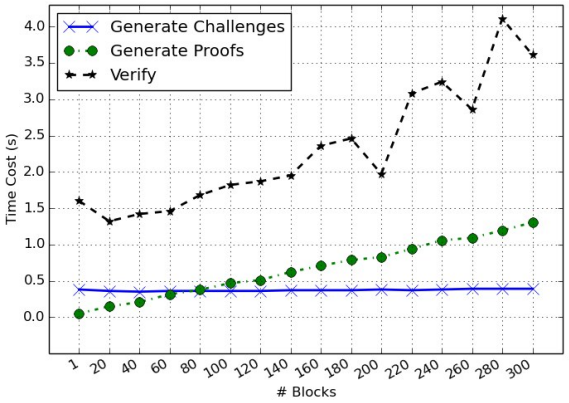


图9 共享数据审计性能

5 小结

各个领域存量数据分散在不同云存储系统中,考虑数据所有权和隐私等问题,未来增量数据势必继续分散在各个机构的数据中心。因此,领域数据的分散性是其固有特性。为了实现这些分散领域大数据的安全和可信共享,本文提出了一种区块链和云计算的融合机制来管理这些分散数据,支撑"原始数据不出域、数据可用不可见"的领域数据共享模式。通过设计共享数据标识编码和解析协议,实现共享数据的定位和寻址;通过设计一种基于区块链智能合约机制的轻量级属性基加密访问控制算法,实现共享数据的"可控可计算";通过建立一种基于云链融合机制的数据完整性审计算法,保障数据的可信共享。以上述模型和算法为基础,本文研发了相应实际原型系统,测试分析了其性能,验证了模型和算法的可行性和先进性。

参考文献(References):

[1] Xia X, Chen F, He Q, et al. Online collaborative data caching in edge computing[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(2): 281-294.

[2] Guo H, Zhang Z, Xu J, et al. Accountable proxy re-encryption for secure data sharing[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 18(1): 145-159.

[3] Abdulqadir H R, Zeebaree S R M, Shukur, H M, et al. A study of moving from cloud computing to fog computing [J].Qubahan Academic Journal 2021, 2 (1): 60-70.

[4] Wu H, Deng S, Li W, et al. Mobility-aware service selection in mobile edge computing systems[C]. IEEE International Conference on Web Services (ICWS), 2019:201-208.

[5] WangY, WangW, LiuD, et al. Enabling edge-cloud video analytics for robotics applications[J]. IEEE Transactions on Cloud Computing (Early Access), 2021:1-1.

[6] Deepa N, Pham Q V, Nguyen D C, et al. A survey on blockchain for big data: approaches, opportunities, and future directions[J]. Future Generation Computer Systems, 2022, 131: 209-226.

[7] 张奥, 白晓颖. 区块链隐私保护研究与实践综述[J]. 软件学报, 2020, 31(05): 1406-1434.

[8] Meiklejohn S, Pomarole M, Jordan G, et al. A fistful of bitcoins: characterizing payments among men with no names [C]. Proceedings of the 2013 Internet Measurement Conference (IMC 2013), 2013:127-140.

[9] Agrawal S, Melissa C. FAME: fast attribute-based message encryption[C]. Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, 2017:665-682.

[10] Ateniese G, Burns R, CurtmolaR, et al. Provable Data Possession at Untrusted Stores[C]. CCS '07: Proceedings of the 14th ACM conference on Computer and communications security, 2007: 598-609.

[11] Hyperledger Caliper [DB/OL]. [2022-03-18] https://hyperledger.github.io/caliper/

[12] fentec-project/gofe(Public)[DB/OL]. [2022-03-18]https://github.com/fentec-project/gofe

[13] Androulaki E, Barger A, Bortnikov V, et al. Hyperledger fabric: a distributed operating system for permissioned blockchains[C]. EuroSys '18: Proceedings of the Thirteenth EuroSys Conference, 2018: 1-15.

编辑:王谦,王雨田