Operations Research and Fuzziology 运筹与模糊学, 2023, 13(3), 1824-1838
Published Online June 2023 in Hans. https://www.hanspub.org/journal/orf
https://doi.org/10.12677/orf.2023.133182

Hans汉斯

基于联盟链的安全公平细粒度数据交易方案

白　瑞，陈玉玲，谭超月，杨宇项

贵州大学公共大数据国家重点实验室、计算机科学与技术学院，贵州　贵阳

收稿日期：2023年4月7日；录用日期：2023年6月12日；发布日期：2023年6月20日

摘　要

针对当前区块链数据交易中数据安全、数据滥用、数据隐私等问题，提出一种基于联盟链与属性基加密的数据交易方案，为满足数据卖家的细粒度访问控制的需求，提高数据价值，利用密文策略属性基加密技术保护待售数据安全，保证只有拥有特定属性的数据买家才可以购买并正确解密数据；针对数据交易中身份隐私保护与监管追溯难以平衡问题，引入智能合约以及群签名技术实现了数据交易的安全可信以及匿名可追溯。最后，对提出的方案进行安全性分析以及仿真实验，结果表明所提方案可以实现安全公平的细粒度数据交易。

关键词

数据交易，联盟链，智能合约，属性基加密，群签名

A Secure and Fair Fine-Grained Data Trading Scheme Based on Consortium Blockchain

Rui Bai, Yuling Chen, Chaoyue Tan, Yuxiang Yang

State Key Laboratory of Public Big Data, College of Computer Science and Technology, Guizhou University, Guiyang Guizhou

Received: Apr. 7th, 2023; accepted: Jun. 12th, 2023; published: Jun. 20th, 2023

Abstract

This paragraph presents a data trading scheme that leverages a consortium blockchain and attribute-based encryption to address various issues such as data security, misuse, and privacy in blockchain data trading. In order to meet the fine-grained access control requirements of data sellers and increase data value, the scheme employs ciphertext-policy attribute-based encryption technology to secure data for sale and ensure that only data buyers with specific attributes can purchase and correctly decrypt the data. Furthermore, to strike a balance between identity priva-

文章引用: 白瑞, 陈玉玲, 谭超月, 杨宇项. 基于联盟链的安全公平细粒度数据交易方案[J]. 运筹与模糊学, 2023, 13(3): 1824-1838. DOI: 10.12677/orf.2023.133182

cy protection and regulatory traceability in data trading, the scheme utilizes smart contracts and group signature technology to achieve secure and anonymous traceability of data trading. Finally, through security analysis and simulation experiments, the effectiveness of the proposed scheme is demonstrated in achieving efficient and fine-grained data trading.

Keywords

Data Trading, Consortium Blockchain, Smart Contracts, Attribute-Based Encryption, Group Signature

Copyright © 2023 by author(s) and Hans Publishers Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/
Open Access

1. 引言

随着信息技术的快速发展，人们对数据获取、数据存储、数据分析的能力不断提高，全球的数据也呈现爆发式增长、海量聚集的特点[1]。数据在商业、工业、农业等领域中的应用越来越广泛，并且正在成为一种可以交换、交易、整合的商品。因此，如何推动大数据产业的创新发展、构建以数据为关键要素的数字经济成为大数据时代面临的一项重大难题。2020 年 4 月，中共中央、国务院发布《关于构建更加完善的要素市场化配置体制机制的意见》，将数据同土地、劳动力、资本、技术等传统生产要素并列，作为一种新型生产要素参与分配。随着交易成为释放生产要素价值的关键环节，数据资源的交换和流通越来越重要，因此人们对此的需求也日益增强。随着互联网公司、移动设备运营商、物联网设备运营商等一些企业或组织囤积了大量的数据，但是由于公司或组织自身技术限制，无法发掘数据的全部价值，因此数据共享成为了打破数据孤岛，实现数据资源共享流通的重要方式。作为一种有偿式数据共享，数据交易可以激励拥有数据的企业或组织主动分享数据，所以大多数公司都愿意将蕴含巨大商业价值的数据进行数据交易。为了深度挖掘大数据的价值，促进数据的商品化，越来越多的数据交易市场已经建立起来，为数据卖家和数据买家之间搭建了桥梁。

目前数据交易已经成为现代数字经济中不可或缺的一部分。然而，数据交易涉及的隐私保护、数据安全、数据滥用等问题[2] [3]已经成为当前数据交易领域面临的重大挑战。尤其是在云计算、物联网和大数据等技术的不断发展下，传统的中心化数据交易平台已经无法满足这些新场景下数据安全与隐私保护的需求。因此，研究如何构建一种安全可靠的数据交易系统是非常必要的。

区块链是以比特币为代表的数字加密货币体系的核心支撑技术[4]，是一种由多个节点通过运行一系列共识算法如工作量证明机制 Pow、股份证明机制 Pos，共同参与维护的分布式数据库，具有去中心化、不可篡改、可追溯等特性，与数据交易十分契合，可以保证数据交易的安全可信。所以国内外学者纷纷对两者结合的数据交易方案展开研究。但是区块链技术在数据交易领域的应用还处于早期发展阶段，应用方案尚未成熟[5]。

Jung 等人[6]针对数据交易中的“二次转卖”问题，提出了一套名为 AccountTrade 的交易协议，为了防止不诚实的转卖行为，提出一个量化数据集的唯一性的指标，然后定义了两个问责模型，且在实验中证明该模型是有效的。Sadiq 等人[7]使用联盟链来构建物联网的信任环境，此外还利用布隆过滤器实现交易数据的快速检索，采用椭圆曲线双线性配对的数字签名方案保证了交易数据的可靠性和完整性，并引入星际文件系统 IPFS 保证物联网数据的长期可用性。Abubaker 等人[8]针对数据交易中数据需求者对数

据质量的担忧问题，将物联网数据货币化，并提出一种基于区块链的物联网数据声誉系统，该系统认为数据质量具有完整性、准确性、可靠性、及时性和完整性，并构建了一套评论系统，将数据的评级和评论永久存储在区块链中，为用户提供数据质量的保证，且使用高级加密标准 AES 加密数据保证数据的完整性，最后设计了仲裁机制，通过提供奖励的方式激励仲裁员参与解决数据提供者与数据需求者的交易纠纷。Chen 等人[9]提出一种基于区块链的不可否认物联网数据交易方案，该方案包括两部分，分别为交易方案和仲裁方案，交易方案分为分治法以及两种承诺法保证高效的数据交易，而仲裁方案则包括链上，链下两部分，若用户对实时的链上的仲裁结果不满，还可以利用离线仲裁作出最终决断，解决了物联网数据交易中可靠性以及及时性的限制。在云计算环境中，Li 等人[10]提出一种基于区块链的分布式数据交易模型为买方、卖方和代理节点构建了信任关系，并且增加了用户交易数据的激励，最后采用双重拍卖机制实现社会福利最大化。虽然基于区块链的数据交易可以实现公平、安全的数据交易，但是由于数据卖家无法控制自身数据出售给哪一类数据买家，导致了数据滥用问题[11]。所以设计一种安全公平，具有细粒度访问控制的数据交易方案是一项重要任务。

针对上述问题，本文提出了一种基于联盟链与属性基加密的数据交易方案。该方案利用联盟链的弱中心化特点，实现了数据交易的分布式和可追溯的特性，同时引入属性基加密[12] [13]与群签名[14]技术，有效保护了数据安全与交易方隐私性。本文方案不仅能够实现高效、安全的数据交易，同时还能够满足不同用户对于数据隐私与数据访问控制的需求。本文的研究成果对于推动数据交易市场的健康发展以及保护用户隐私具有重要意义。

2. 设计目标

2.1. 数据安全性

在数据量较大的情况下，如果直接将数据上传到区块链网络，会造成巨大的存储开销，所以一般将数据存储在云服务器上，但是云服务器是诚实且好奇的[15]，所以需要先将数据加密再上传至云服务器，数据的安全性由加密数据的密码算法保证。

2.2. 细粒度访问控制

在某些场景下，需要考虑数据的分享对象，比如人们更希望将医疗数据只分享给某些医疗机构或医院，就需要数据交易支持更细粒度的访问控制。传统的公钥密码在处理这些问题时会面临密钥管理方面的难题，而且每一次与不同的买家交易时要重新加密数据文件，造成系统开销的增加。密文策略属性基加密是最适合解决该问题的技术，数据卖家通过设计数据访问结构即可满足对数据细粒度访问控制的需求。

2.3. 公平性

在传统中心化数据交易过程中，可能会出现各种不公平的现象，比如数据拥有者与数据交易平台合谋欺骗数据使用者，或者数据使用者与数据交易平台合谋欺骗数据拥有者等，在传统数据交易方案中，难以保证交易的公平性。区块链是一种具有去中心化、防篡改等特点的分布式数据库，智能合约[16] [17]在被引入区块链之后被认为是一种“可信第三方”，可以保证交易“一手交钱一手交货”，但是由于智能合约只能死板的执行预先设定好的脚本，使恶意的参与者有了可乘之机，所以需要引入仲裁机制与智能合约一起保证数据交易的公平性。

2.4. 匿名性

由于区块链具有透明性的特点，存储在区块链上的信息可以被所有节点看到，为了保护用户的隐私，

上链的交易信息需要将交易双方匿名，防止其他用户通过分析特定交易记录来确认数据交易双方的身份，所以当数据交易正确执行时，只有交易双方知道对方的身份。

2.5. 可追溯性

数据交易过程中可能存在各种各样的恶意用户，比如售卖不合数据描述的数据的卖家、意图拿到数据不支付的买家等，这时就需要系统可以根据交易记录追溯到恶意用户的真实身份并对其进行惩罚。

3. 模型架构

3.1. 角色介绍

如图 1 所示，本方案基于联盟链构建，主要包含数据卖家，数据买家，证书机构、云存储服务提供商、属性颁发机构、公共审计员六个实体。

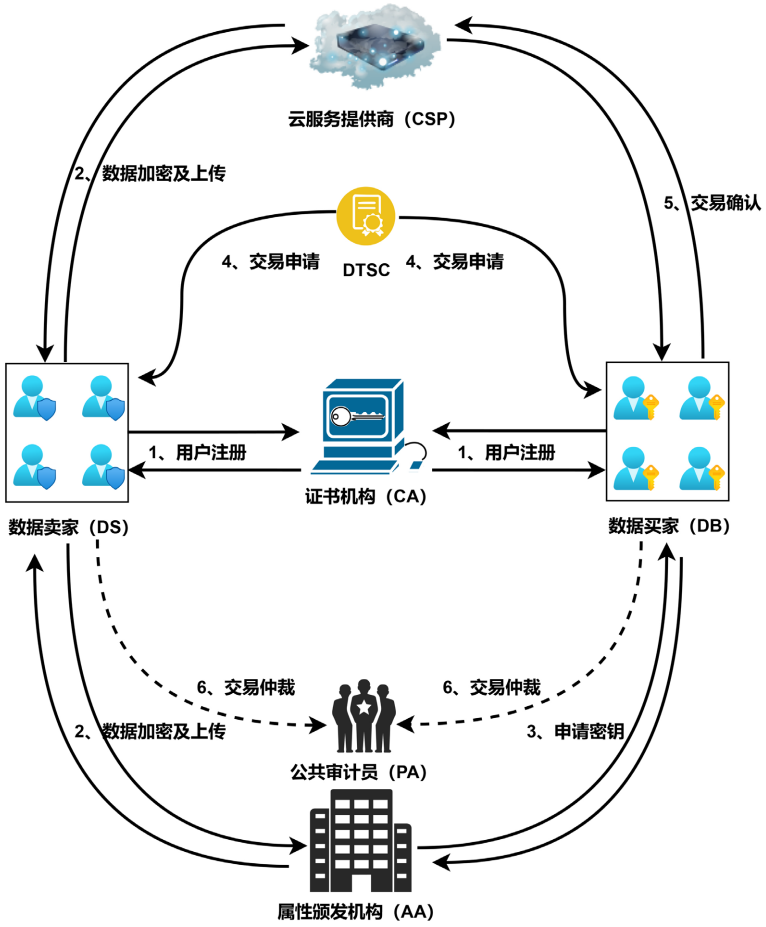

Figure 1. System model diagram
图 1. 系统模型图

数据卖家(Data Sellers, DS)：是数据的提供者，包括但不限于政府及企事业单位，从互联网、物联网等方式收集数据或者自身产生数据，并通过数据交易出售这些数据获得利益。负责加密数据并将其存储到云端。在本方案中，为实现更细粒度的数据访问控制，数据卖家需要为待售数据文件设定属性集，并指定数据访问结构，保证只有拥有被允许访问的属性的数据买家才可以购买并解密相应数据文件。

数据买家(Data Buyers, DB)：是数据的需求者，可以利用数据进行大数据分析、数据挖掘、机器学习模型训练等任务，在区块链上通过数据卖家上传的数据描述和访问属性需求信息选择合适的数据文件。

证书机构(Certificate Authority, CA)：假设证书机构是可信的，负责管理系统中用户的公钥证书，并维护证书与用户公钥之间的映射关系。

云服务提供商(Cloud Service Provider, CSP)：假设云服务提供商是诚实且好奇的，它为数据卖家提供云存储服务，会诚实的执行数据卖家和数据买家的请求，同时也会窥视存储在它上面的数据文件。

属性颁发机构(Attributes Authority, AA)：负责审核数据买家的属性证明并为数据买家颁发对应的属性密钥。

公共审计员(Public Auditor, PA)：负责为数据买家和数据卖家提供仲裁服务，并在交易发生纠纷时，由其上传仲裁结果，打开群签名追溯恶意方的真实身份，解决关于数据正确性和一致性的争议。

3.2. 系统流程

系统运行流程主要分为全局初始化、用户注册、数据加密及上传、申请密钥、交易申请、交易确认及交易仲裁七个阶段。

1) 全局初始化：在此阶段生成 CA、PA、AA 的公私钥对以及公共参数；

2) 用户注册：DS 和 DB 在加入联盟链时，首先向 CA 提交身份认证申请，然后 CA 在实体身份认证审核通过后为其颁发证书文件，然后将证书加入一个区块链交易并进行广播，待区块链交易被确认后，将被确认的交易 ID 和用户公钥的映射加入到系统的映射关系中；

3) 数据加密及上传：DS 首先采用 CP-ABE 方案对待售数据进行加密，加密完成后通过智能合约将数据发布至联盟链，同时将密文上传至 CSP 处，CSP 返回一个数据存储地址给 DS 保存；

4) 申请密钥：DB 根据访问结构和数据描述选定需求数据后，向 AA 发起密钥申请，AA 在联盟链上查找到对应数据的访问控制策略，并判断 DB 拥有的属性是否符合该数据的访问结构，若符合，则根据 DB 的属性集为 DB 颁发属性密钥；

5) 数据交易：DB 在得到密钥后，向 DS 发起数据交易申请，然后调用智能合约输入相关信息；DS 收到交易申请后，将数据地址用 DB 的公钥加密后通过智能合约发送给 DB；DB 在收到 DS 加密过后的数据地址后，用自己的私钥解密，并到 CSP 处下载数据密文，通过在步骤 4)中得到的属性密钥解密密文。验证数据是否与数据描述一致，其中包括数据大小、数据可用性及数据完整性。在验证无误后，本次数据交易完成，并对交易信息进行群签名发送给背书节点；

6) 交易确认：背书节点收到交易信息后，验证群签名的合法性，验证通过后，将数据交易记录信息上传到区块链进行存储；

7) 交易仲裁：DB 若在步骤 6)中发现数据与数据描述不一致，则向 PA 发起仲裁请求，并提交相关证明。PA 收到该请求后根据证明及交易信息在规定时间内对交易双方进行仲裁，并追踪恶意方的真实身份信息，随后调用智能合约将结果返回至交易合约，交易合约根据仲裁结果对恶意方进行惩罚。

4. 方案构造

4.1. 算法设计

1) $SATDT.SystemSetup(\lambda)\rightarrow(P,MK)$

初始化算法以安全参数 λ 作为输入，输出公共参数 P 和用来进行数据加密的系统主密钥 MK。

2) $SATDT.Register(P,sk_{CA},pk_u)\rightarrow(cer_u,sk_u)$

注册算法以公共参数 P、CA 的私钥 sk_{CA} 以及用户 u 的公钥 pk_u 作为输入，输出用户 u 的数字证书 cer_u

以及用户 u 的私钥 sk_u。

3) $SATDT.KeyGen(P,MK,S)\to(Ask_u)$

密钥生成算法以公共参数 P、主密钥 MK 以及用户 u 的属性集 S 作为输入，输出用户 u 的属性密钥 Ask_u。

4) $SATDT.Encrypt(P,F,\mathbb{A})\to(CT)$

该算法以公共参数 P、数据文件 F 及数据访问结构 $\mathbb{A}$ 作为输入，输出密文 CT。

5) $SATDT.Decrypt(P,CT,Ask_u)\to(For\perp)$

解密算法以公共参数 P、密文 CT、用户 u 的属性密钥 Ask_u 作为输入，当用户 u 的属性集满足访问结构$\mathbb{A}$时输出数据文件 F，否则输出$\perp$。

6) $SATDT.\mathrm{sgn}(P,sk_u,m,aux)\to\sigma$

签名算法以公共参数 P、用户私钥 sk_u、交易信息 m、辅助信息 aux 作为输入，输出一个签名 σ。

7) $SATDT.Verify(P,m,aux,\sigma)\to(1\text{ or }0)$

验证算法以公共参数 P、交易信息 m、辅助信息 aux、签名 σ 为输入，如果签名 σ 由联盟链成员所签，则输出 1，否则输出 0。

8) $SATDT.Trace(P,sk_{pA},m,aux,\sigma)\to(pk_u,cer_u)$

追溯算法以公共参数 P，PA 的私钥 sk_{CA}，交易信息 m，辅助信息 aux 以及一个签名 σ 作为输入，输出恶意用户的公钥 pk_u 和证书 cer_u。

4.2. 智能合约设计

在本小节中，主要介绍本方案用到的两个智能合约数据发布智能合约(DRSC)和数据交易智能合约(DTSC)。

1、数据发布合约的整体流程如算法 1 所示。

Algorithm 1. Data release smart contract
算法 1. 数据发布智能合约

输入：数据描述信息 DT_F，押金 $Deposit_{DS}$
输出：0/1 //返回数据是否发布成功，0 代表失败，1 代表成功

1：dateNow = time.now() // 获取数据发布时间
2：检查输入参数个数以及参数是否合法
3：**if** $Deposit_{DS}\geq p_F$ **then**
4：生成唯一数据标识 DataID
5：idHash = gethash(DataID)//获取数据 ID 的哈希
6：Datalist.add(DT_F, idHash)//将 DT_F 和 idHash 存储到数据列表
7：return 1
8：**else**
9：返还预存押金
10：return 0
11：**end if**

算法 1 展示了 DRSC 的整体流程，该智能合约的输入为经 DS 群签名后的数据描述信息 DT_F，包括 DS 的公钥 pk_{DS}、数据描述 Des_F，数据价格 p_F，访问结构 $\mathbb{A}_F$，还需要向 DRSC 的地址存入押金 $Deposit_{DS}$。

2、数据交易智能合约的整体流程如算法 2 所示。

算法 2 展示了 DTSC 的整体流程，DB 需要向智能合约输入对应数据 ID 的哈希 idHash 以及用 DS 加密的 DB 的公钥 $Enc(pk_{DB})$，并向该智能合约地址预存押金 $Deposit_{DB}$；DS 需要向 DTSC 输入加密后的数据存储地址；若交易存在纠纷，则需要 PA 向智能合约输入仲裁结果 Res。

Algorithm 2. Data trading smart contract
算法 2. 数据交易智能合约

输入： idHash， $Deposit_{DB}$， $Enc(pk_{DB})$， $Enc(addr)$，*finalize*/*arbitration*，*Res*
输出： 0/1 //返回交易是否成功，0 代表失败，1 代表成功

1：dateNow = time.now() // 获取交易时间
2：检查输入参数个数以及参数是否合法
3：**if** $Datalist.query(idHash)==1$ **then**//检查数据是否存在
4：**if** $Deposit_{DB} \geq p_F$ **then**//判断押金是否符合条件
5：将 $Enc(addr)$ 发送给 DB
6：DB 验证数据是否符合条件
7：**if** 到达预定时间 **then**
8： $Deposit_{DB}$ to DS
9： $Deposit_{DS}$ to DS
10： return 1
11：**end if**
12：**if** DB 发送 *finalize* **then**
13： $Deposit_{DB}$ to DS
14： $Deposit_{DS}$ to DS
15： return 1
16：**end if**
17：**if** DB 发送 *arbitration* **then**
18：进行链下仲裁
19：PA 发送 Res //0 代表交易继续，1 代表 DS 恶意，2 代表 DB 恶意
20：**if** Res == 0 **then**
21： $Deposit_{DB}$ to DS
22： $Deposit_{DS}$ to DS
23：return 1
24：**end if**
25：**if** Res == 1 **then**
26： $Deposit_{DB}$ to DS
27： $Deposit_{DS}$ to DS
28： return 0
29：**end if**
30：**if** Res == 2 **then**
31： $Deposit_{DB}$ to DB
32： $Deposit_{DS}$ to DB
33： return 0
34：**end if**
35：**end if**
36：**end if**

4.3. 详细构造

全局初始化：

CA 运行 *SATDT.SystemSetup* 算法，并设置 λ 为安全参数，选择阶为素数 p 的循环群 G_0， G_1， G_2，G_T，其中 G_0 是一个双线性群，(G_1, G_2) 是一个双线性群对且存在可计算同构映射 φ，设 g_0， g_1， g_2 分别为群 G_0， G_1， G_2 的生成元且满足 $g_2 = \varphi(g_1)$。假设 $e_1 : G_0 * G_0 \to G_3$ 与 $e_2 : G_1 * G_2 \to G_T$ 为两个双线性映射。构建三个抗碰撞的哈希函数：$H_0 : \{0,1\}^* \to G_0$， $H_1 : \{0,1\}^* \to G_1$， $H_2 : \{0,1\}^* \to \mathbb{Z}_p$，并随机选择两个

参数 $h,u \in G_1$。

在本方案中，证书机构 CA 和公共审计员 PA 作为群管理员，需要为其分别生成公私钥对。

首先随机选择两个正整数分别为 $\alpha,\beta \in \mathbb{Z}_p$，然后计算加密主私钥 MK，

$$MK = \beta, g_0^{\alpha} \tag{1}$$

然后随机选择 $\gamma \in \mathbb{Z}_p$ 并计算 $\omega = g_2^{\gamma}$，则，

$$sk_{CA} = \gamma, pk_{CA} = \omega \tag{2}$$

PA 随机选择两个群 G_1 的生成元 v_1, v_2，并随机选择两个正整数 $k_1, k_2 \in \mathbb{Z}_p$ 满足如下关系，

$$u = v_1^{k_1} = v_2^{k_2} \tag{3}$$

则

$$sk_{PA} = (k_1, k_2), pk_{PA} = u \tag{4}$$

最后，输出公共参数

$$P = (p, G_0, G_1, G_2, G_3, G_T, g_0, g_1, g_2, e_1, e_2, h, u, v_1, v_2, \omega, H_0, H_1, H_2) \tag{5}$$

CA 将公共参数 P 上传到区块链，并将主私钥 MK 通过安全信道发送给 AA。

用户注册：

若用户想要加入联盟链进行数据交易，需要向 CA 发起实体资质审核申请，CA 在实体资质审核通过后会调用 *SATDT.Register* 算法，为用户颁发数字证书。具体来说：

1、首先用户随机选择一个正整数 $y \in \mathbb{Z}_p$，并计算 $Y = h^y$ 并将 Y 发送给 CA，同时需要证明 Y 的知识证明基于 $h: pk_u \{(y): Y = h^y\}$。

2、CA 随机选择一个正整数 $x \in \mathbb{Z}_p$，并设置 $A = \left(g_1 Y^{-1}\right)^{\frac{1}{x+\gamma}}$，其中 γ 为 *CA* 的私钥。然后将 (x, A) 发送给用户作为该用户的成员证书。

3、收到 (x, A) 后，用户检查以下等式是否成立：

$$e_2\left(A, \omega g_2^x\right) = e_2\left(g_1 h^{-y}, g_2\right) \tag{6}$$

若上述等式成立，则接受该成员证书，并将 (x, A) 以及公钥 Y 发送给 CA，用户保存 (A, x, y) 作为其私钥 sk_u。

CA 将该用户的注册信息 (x, A) 和公钥 Y 加入成员列表中。

数据加密及发布：

1、DS 作为数据提供者，在需要出售数据时，首先为自己的待售数据文件设定访问结构 $\mathbb{A}$，然后运行 *SATDT.Encrypt* 算法对数据文件 F 进行加密，具体方法如下：

DS 自顶向下的为访问控制结构中的访问控制树 T 的每一个节点 x 随机选择一个多项式 q_x，q_x 的阶 d_x 为节点 x 的阈值 $k_x - 1$，即 $d_x = k_x - 1$。

具体来说，对于访问控制树 T 的根节点 R，设置根节点的多项式 $q_R(0) = s$，然后选择 d_R 个随机数 $\{a_1, \cdots, a_{d_R}\}$ 来定义 q_R，表示为：

$$q_R = s + a_1 x + \cdots + a_{d_R} x^{d_R} \tag{7}$$

对于 T 中的其他节点 x，令 $q_x(0) = q_{parent(x)}(index(x))$，并随机选择 d_x 个随机数来定义多项式 q_x。其中 $parent(x)$ 表示节点 x 的父节点，$index(x)$ 表示节点 x 在父节点的孩子节点中的序号。假设 Y 为数据访问控制树 T 所有叶子节点的集合，则根据数据文件的访问控制树 T 构造的密文为：

$$CT_F=\left(T,C=g_0{}^{\beta *s},\tilde{C}=F*e_1\left(g_0,g_0\right)^{\alpha *s},\left(\forall y\in Y:C_y=g_0{}^{q_y(0)},C_y'=H_0\left(att\left(y\right)^{q_y(0)}\right)\right)\right) \tag{8}$$

其中 $att\left(y\right)$ 代表与叶子节点 y 相关联的属性。

2、DS 将数据文件 F 的描述 Des_F、数据价格 p、访问结构 $\mathbb{A}$、以及 DS 的公钥 pk_{DS} 通过 $DRSC$ 发布至联盟链并对这些信息进行群签名，并将数据加密密文 CT 发送给 CSP，CSP 返回 DS 一个数据存储地址 $addr_F$。

数据交易：

1、属性密钥申请：DB 在联盟链网络中寻找满足自身需求的数据，并验证 DS 的群签名，然后将数据对应的交易 ID 哈希、自己的属性集 S 及对应属性的证明文件发送给 AA，AA 在存储的联盟链数据列表中查找到对应数据的访问结构，然后验证属性证明文件的合法性及属性集的正确性，随后调用 $SATDT.KeyGen$ 生成 DB 的属性密钥。具体流程如下：AA 选择一个随机数 $r\in\mathbb{Z}_p$，对于属性集合 S 中的元素，分别为其选择一个随机数 $r_j\in\mathbb{Z}_p$，使用式(4)~(8)计算 DB 的属性私钥，

$$Ask_{DB}=\left(D=g_0{}^{\frac{\alpha+r}{\beta}},\left(\forall j\in S:D_j=g_0{}^{r}*H_0\left(j\right)^{r_j},D_j'=g_0{}^{r_j}\right)\right) \tag{9}$$

最后将 Ask_{DB} 以安全的方式发送给 DB。

2、数据交易：DB 得到 Ask_{DB} 后，发起数据交易请求，并调用数据交易智能合约 DTSC，DB 向智能合约的地址存入正确的押金，并输入自己的公钥和对应的数据 ID 的哈希，DS 将数据存储地址用 DB 的公钥加密后得到 $Enc_{pk_{DB}}\left(addr_F\right)$，然后发送至智能合约 DTSC，DTSC 将其发送给 DB。

3、数据解密：DB 收到 $Enc_{pk_{DB}}\left(addr_F\right)$ 后，验证 DS 的签名，随后用自己的私钥解密得到数据地址 $addr_F$，然后使用该地址在 CSP 处下载密文，并执行 $SATDT.Decrypt$ 进行解密，具体解密步骤如下：

对于访问控制树 T 中的每一个节点 x，如果 x 为叶子节点，则令 $i=att\left(x\right)$。

如果 $i\in S$，则计算：

$$DN_x=\frac{e_1\left(D_i,C_x\right)}{e_1\left(D_i^{'},C_x^{'}\right)}=\frac{e_1\left(g_0{}^{r}*H_0\left(i\right)^{r_i},g_0{}^{q_x(0)}\right)}{e_1\left(g_0{}^{r_i},H_0\left(i\right)^{q_x(0)}\right)}=e_1\left(g_0,g_0\right)^{r*q_x(0)} \tag{10}$$

如果 $i\notin S$，则 $DN_x=\perp$。

如果 x 为非叶子节点，则对于节点 x 的所有孩子节点 z，按照步骤 1 的方法计算 $DN_z=e\left(g,g\right)^{r*q_z(0)}$。令 S_x 为一个大小为 k_x 的任意的孩子节点 z 的集合，并且 $DN_z\neq\perp$。然后计算

$$DN_x=\prod_{z\in S_x}DN_z^{\Delta_{i,S_x'}(0)}=\prod_{z\in S_x}\left(e_1\left(g_0,g_0\right)^{r*q_{parent(x)}\left(index(x)\right)}\right)^{\Delta_{i,S_x'}(0)}=e\left(g_0,g_0\right)^{r*q_x(0)} \tag{11}$$

其中 $i=index\left(z\right),S_x'=\left\{index\left(z\right):z\in S_x\right\},\Delta_{i,S_x'}\left(0\right)=\prod_{\Delta j\in S_x',j\neq i}\frac{-j}{i-j}$。

最终可以得到 $DN_R=e_1\left(g_0,g_0\right)^{r*q_R(0)}$。其中 $q_R\left(0\right)=s$。

由以上结果进行如下计算计算可解密得到数据文件 F：

$$F=\frac{\tilde{C}}{e_1\left(C,D\right)/DN_R}=\frac{F*e\left(g_0,g_0\right)^{\alpha *s}}{e_1\left(g_0{}^{\beta *s},g_0{}^{\frac{\alpha+r}{\beta}}\right)/e_1\left(g_0,g_0\right)^{r*q_R(0)}}=\frac{F*e_1\left(g_0,g_0\right)^{\alpha *s}}{e_1\left(g_0,g_0\right)^{\alpha *s}} \tag{12}$$

4、数据验证：得到数据文件 F 后，DB 验证该数据文件的正确性和一致性，主要包括数据大小、可用性和完整性三个方面，若检查无误，则发送 $finalize$ 给 DTSC 代表本次交易结束，智能合约生成一份交

易记录信息 $DT_F=\left(Hash\left(pk_{DB}\parallel pk_{DS}\right),TradeDate,Enc_{pk_{DB}}\left(addr_F\right),oth\right)$，其中 *TradeDate* 代表交易日期，*oth* 代表其他相关信息，最后双方调用 *SATDT.Sign* 对 DT_F 进行群签名并发送至背书节点进行上链存储，DRSC 和 DTSC 将双方押金归还。群签名具体步骤为：选择一个特定的 $aux\in\{0,1\}^*$ 对 DT_F 进行签署，并计算 $u_0=H_1\left(aux\right)$，随机选取两个数 μ,ξ，计算 $l_1=v_1^{\mu},l_2=v_2^{\xi},l_3=A_{DS}\cdot u^{\mu+\xi},l_4=u_0^{x_{DS}},\delta_1=x_{DS}\cdot\mu,\delta_2=x_{DS}\cdot\xi$。随后，对交易信息 DT_F 执行非交互零知识证明 *Sok*Π 如下，则 DT_F 的可链接群签名可以表示为 $\sigma_{DT_F}=\left(l_1,l_2,l_3,l_4,\Pi\right)$。

$$Sok\Pi\left\{\begin{pmatrix}\mu,\xi\\x_{DS},y_{DS}\\\delta_1,\delta_2\end{pmatrix}:V\begin{pmatrix}l_1=v_1^{\mu}\\l_2=v_2^{\xi}\\1_{G_1}=l_1^{x_{DS}}v_1^{\delta_1}\\1_{G_1}=l_2^{x_{DS}}v_2^{\delta_2}\\\dfrac{e_2\left(g_1,g_2\right)}{e_2\left(l_3,\omega\right)}=e_2\left(u,\omega\right)^{-\mu-\xi}e_2\left(l_3,g_2\right)^{x_{DS}}e_2\left(u,g_2\right)^{-\delta_1-\delta_2}e_2\left(h,g_2\right)^{y_{DS}}\\l_4=u_0^{x_{DS}}\end{pmatrix}\right\}DT_F$$

若数据与描述不符，则可向 PA 发起交易仲裁申请，DTSC 暂停，等待 PA 提交仲裁结果，若恶意方为 DB，则将 DB 的押金转至 DS 账户作为补偿；若恶意方为 DS，则将 DS 押金转至 DB 账户作为补偿；若证据不足，则将 DB 的押金作为购买数据的钱支付给 DS。

交易确认：

背书节点在收到经双方签名后的交易信息 DT_F 后，首先验证 DB 和 DS 签名的合法性，其次验证该交易的合法性，最后调用 *SATDT.Verify* 算法验证双方群签名的有效性，对带有特定辅助信息 $aux\in\{0,1\}^*$ 的交易信息群签名的有效性步骤为，计算 $u_0=H_1\left(aux\right)$。然后生成：

$$\tilde{a}_1=v_1^{z_{\mu}}\cdot l_1^{c}$$

$$\tilde{a}_2=v_2^{z_{\xi}}\cdot l_2^{c}$$

$$\tilde{a}_3=e_2\left(u,\omega\right)^{-z_{\mu}-z_{\xi}}\cdot e_2\left(l_3,g_2\right)^{z_{x_{DS}}}\cdot e_2\left(u,g_2\right)^{-z_{\delta_1}-z_{\delta_2}}\cdot e_2\left(h,g_2\right)^{z_{yDS}}\cdot\frac{e_2\left(g_1,g_2\right)^{c}}{e_2\left(l_3,\omega\right)}$$

$$\tilde{a}_4=l_1^{z_{x_{DS}}}\cdot v_1^{z_{\delta_1}}$$

$$\tilde{a}_5=l_2^{z_{x_{DS}}}\cdot v_2^{z_{\delta_2}}$$

$$\tilde{a}_6=l_4^{c}\cdot u_0^{z_{x_{DS}}}$$

计算 $\tilde{c}=H_1\left(DT_F,l_1,l_2,l_3,l_4,\tilde{a}_1,\tilde{a}_2,\tilde{a}_3,\tilde{a}_4,\tilde{a}_5,\tilde{a}_6\right)$ 并检查 $c=\tilde{c}$ 是否成立，如果成立则输出 1，不成立则输出 0。

在上述过程中用到的 $c,z_{x_{DS}},z_{\delta_1}$ 等相关参数是由用户在实例化关于交易信息 DT_F 的知识签名，*Sok*Π 的过程中产生的，实例化过程如下：

用户随机选择 $\left\{w_{\mu},w_{\xi},w_x,w_y,w_{\delta_1},w_{\delta_2}\in\mathbb{Z}_p\right\}$，并计算：

$$a_1=v_1^{w_{\mu}}$$

$$a_2=v_2^{w_{\xi}}$$

$$a_3=e_2\left(u,\omega\right)^{-w_{\mu}-w_{\xi}}\cdot e_2\left(l_3,g_2\right)^{w_x}\cdot e_2\left(u,g_2\right)^{-w_{\delta_1}-w_{\delta_2}}\cdot e_2\left(h,g_2\right)^{w_y}$$

$$a_4 = l_1^{w_x} \cdot v_1^{-w_{\delta_1}}$$

$$a_5 = l_2^{w_x} \cdot v_2^{-w_{\delta_2}}$$

$$a_6 = u_0^{w_x}$$

然后计算 $c = H_1\left(DT_F, l_1, l_2, l_3, l_4, a_1, a_2, a_3, a_4, a_5, a_6\right)$，随后计算：

$$z_\mu = w_\mu - c\mu$$

$$z_\xi = w_\xi - c\xi$$

$$z_x = w_x - cx$$

$$z_y = w_y - cy$$

$$z_{\delta_1} = w_{\delta_1} - c\delta_1$$

$$z_{\delta_2} = w_{\delta_2} - c\delta_2$$

最终输出 $\Pi = \left(\mathrm{c}, z_\mu, z_\xi, z_x, z_y, z_{\delta_1}, z_{\delta_2}\right)$。

验证者只需要计算以下等式即可验证 Π：

$$a_1 l_1^c = v_1^{z_\mu}$$

$$a_2 l_2^c = v_1^{z_\xi}$$

$$a_3 \cdot \frac{e_2\left(g_1, g_2\right)^c}{e_2\left(l_3, \omega\right)} = e_2\left(u, \omega\right)^{-z_\mu - z_\xi} \cdot e_2\left(l_3, g_2\right)^{z_x} \cdot e_2\left(u, g_2\right)^{-z_{\delta_1} - z_{\delta_2}} \cdot e_2\left(h, g_2\right)^{z_y}$$

$$a_4 = l_1^{z_x} \cdot v_1^{-z_{\delta_1}}$$

$$a_5 = l_2^{z_x} \cdot v_2^{-z_{\delta_2}}$$

$$a_6 l_4^c = u_0^{w_x}$$

如果上述等式全部成立则输出 1，否则输出 0。

交易仲裁：

若在数据交易阶段，DS 和 DB 关于数据的完整性和一致性出现了纠纷，DB 会向 PA 提交交易仲裁申请，PA 根据 DB 和 DS 提交的证据判断是哪一方是恶意的，并公布结果。

在交易仲裁阶段，仲裁发起者收到仲裁请求后，对相关信息进行取证并判断，最后反馈仲裁结果并运行 *SATDT.Trace* 算法揭秘恶意方真实身份。具体流程如下：

1) 仲裁申请：本方案中，由于 DS 只需要被动的等待 DB 购买数据，只要发送了加密的数据地址，一定可以得到数据金额，所以交易仲裁发起者为 DB。DB 在解密后发现数据存在数据大小、可用性、完整性与数据描述不相符、数据重复率高等情况，则可向 PA 发起仲裁请求；

2) 仲裁合法性判断：PA 收到 DB 发送的仲裁请求后，首先判断 DB 身份是否合规，然后判断交易是否合法，决定是否继续进行交易仲裁；

3) 交易取证：若上述步骤全部通过，则向 DS、DB 进行取证，DB 提交有关数据的错误证明及交易信息，DS 提交反驳证明及交易信息，然后 PA 根据双方提交的证据及公开得到的信息等进行比对；

4) 仲裁判决：如果 PA 认为 DB 提交的材料证据不足以认定数据存在问题，则交易继续进行；若 PA 认定 DB 提交的材料是虚假的，则认定 DB 为恶意方，对 DB 的群签名进行追溯，使用 PA 的私钥通过公式(13)可以计算得到 DB 的公钥：

$$\frac{l_3}{l_1^{k_1}\cdot l_2^{k_2}} = A_{DB} \tag{13}$$

若 DB 证据充足，则 PA 认定 DS 是恶意的，对 DS 的群签名进行追溯，使用 PA 的私钥通过上述公式可以计算得到 DS 的公钥；

5) 惩罚阶段：在得到恶意方的公钥后，将恶意方的公钥发送给 CA，由于在进行数据交易时双方都支付了押金，所以 CA 会没收恶意方押金并将其转给诚实方作为补偿。然后 CA 通过比对注册表可以获得用户在注册时填写的身份信息和对应的数字证书，CA 公布恶意方真实身份并将其注册信息移除注册表。

5. 分析与评估

5.1. 安全性分析

数据安全性：对数据安全的挑战主要来自于恶意数据买家的串通攻击，具体来说，数据买家为了获取更多的数据，可能会串通其他拥有不同属性的买家来解密数据，恶意买家之间会将拿到的属性密钥 ASK 集中起来解密数据。对于串通攻击，在 *SATDT.KeyGen* 算法中，AA 使用不同的随机数为数据买家生成属性密钥，就算数据买家将各自的属性密钥结合在一起，也不能解密额外密文。另一方面，由于 CSP 是诚实且好奇的，他会对存储在 Cloud 上的数据感兴趣，并试图解密，但是他只拥有密文 CT 和系统公共参数 P，而解密密文的条件之一则是需要获取 $e_1(g_0,g_0)^{\alpha s}$，从而计算 $\frac{\tilde{C}}{e_1(C,D)/DN_R}$ 得到数据文件 F，然而如果没有正确的 ASK 则无法解密数据，由于要得到数据密文，必须通过智能合约与 DS 交易得到数据存储地址，所以其他具有相同属性的 DB 就算得到属性密钥也无法得到密文。

交易安全性：交易安全性主要分为两方面，交易公平和隐私安全。在进行数据交易时，恶意的数据卖家可能会发布与数据描述不符的数据，当数据买家得到与数据描述不符的数据时会破坏交易的公平性；恶意的数据买家可能会存在侥幸心理，提交虚假的证明来欺骗 PA，干扰 PA 正确判断；恶意的数据买家也可能在得到数据地址后用不正当的方式结束智能合约，从而逃避支付。针对前两种恶意行为，本方案设计了仲裁机制来保证方案的交易公平，数据买家若对数据正确性和一致性存疑，则可向 PA 提出仲裁请求并提交对应证据，诚实的 PA 会综合证据和公开信息进行决断。针对第三种恶意行为，数据交易智能合约设置了规定时间，在规定时间内若买家不主动发送结束交易标识或进入仲裁阶段，则智能合约自动执行支付。针对以上恶意参与方，PA 会使用 *SATDT.Trace* 算法追踪恶意方的公钥信息并揭露其真实身份，并取消其交易资格。区块链透明性的特点使得数据交易双方的身份信息会泄露，所以本方案采用群签名，使数据买家和数据卖家各自对不含隐私的交易记录信息进行群签名后再将由背书节点将其上传存储，除交易双方外，其他节点不能通过分析交易记录来得到某笔交易的交易者。

5.2. 性能分析

本章进行实验仿真的环境为：操作系统 Windows 11 家庭中文版；处理器 12th Gen Intel (R) Core (TM) i7-12700H2.30 GHz；RAM 16.0 GB。并采用 HyperLedgerFabric (v 2.2.0)作为联盟链的测试环境，采用 Raft 共识机制，联盟链设置一个排序节点 orderer 和两个组织 org0 和 org1，每一个组织下对应一个节点 peer0，本小节在上述环境下采用 Caliper 工具对联盟链的吞吐量进行测试。

在部署智能合约后，联盟链的吞吐量如表 1 所示，系统对查询数据函数 QueryData 最大响应时间为 0.12 s，最小响应时间为 0.00 s，平均每秒处理的交易数为 171.5；上传数据函数 UpdateData 最大响应时间为 0.75 s，最小响应时间为 0.01 s，平均每秒处理的交易数为 109.2；数据交易函数 TradingData 最大响应时间为 0.23 s，最小响应时间为 0.01 s，平均每秒处理的交易数为 118.5。

Table 1. Consortium through put
表 1. 联盟链吞吐量

功能	成功数量	发送速率	最大延迟	最小延迟	吞吐量
QueryData	5086	171.6	0.12	0.00	171.5
UpdateData	3243	109.3	0.75	0.01	109.2
DataTrading	3514	118.6	0.23	0.01	118.5

本节在文献[18]的基础上基于 JPBC 库使用 Java 语言实现了方案中数据加密相关算法，测试了当访问树叶子节点和数据买家拥有属性分别为 1~5 时的密钥生成和加解密效率。如图 2 所示，设置访问树叶子节点数量为 5 时，当 DB 的属性数量增加时，密钥生成时间和数据解密时间也相应增长，初始化和加密时间基本不变，当属性数量为 5 时，密钥生成的时间达到 126.4 ms，解密时间达到 71.1 ms。由于基于访问树策略的 CP-ABE 方案加密算法只与访问树叶子节点有关，如图 3，设置 DB 属性数量固定，当访问树的叶子节点增加时，相应的加密时间也随之上升，而初始化、密钥生成和解密时间基本不变。

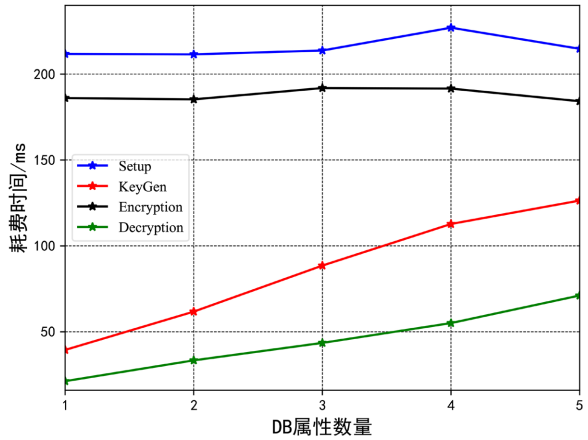

Figure 2. Time cost of each algorithm for attribute base encryption (number of access tree leaf nodes is 5)
图 2. 属性基加密各算法耗费时间(访问树叶子节点数量为 5)

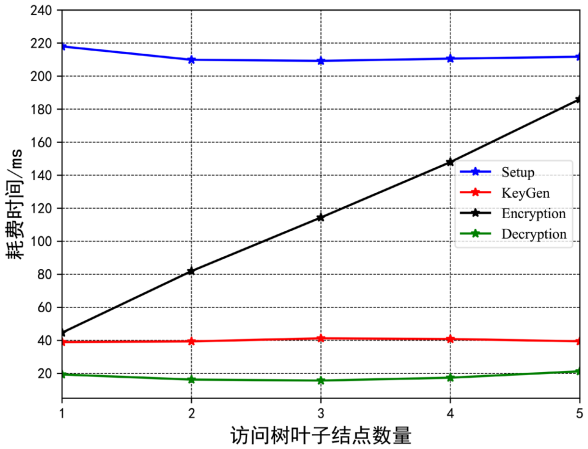

Figure 3. Time cost of each algorithm for attribute-based encryption (number of DB attributes fixed at 1)
图 3. 属性基加密各算法耗费时间(DB 属性数量固定为 1)

5.3. 方案对比

本小节从去中心化、公平性、匿名性、可追溯性和访问控制五方面将本方案与其他数据交易方案进

行对比，对比结果如下表所示。

如表 2 所示，其中的文献都利用了区块链解决了传统数据交易的中心化问题，文献[19]设计了一种具有隐私性的数据交易协议，保证了交易双方的权益，实现了隐私性，但是没有考虑交易的公平性和可监管性并且不能保证数据的细粒度访问控制。同样基于区块链去中心化的特性，文献[20]在数据交易中设计了仲裁机制，集成了一个确定性公钥加密算法和智能合约来保证数据的安全性和交易的公平性，但在匿名性、可追溯性等方面存在不足；在文献[19]的基础上，文献[21]利用联盟链和群签名实现了恶意行为的可监管的特性，但是仍未实现数据的细粒度访问控制。本方案不仅实现了去中心化、匿名性、公平性和可追溯性，而且实现了数据交易的细粒度访问控制，可以在保证数据安全的同时，有效地控制数据的访问权限，提高数据的价值。

Table 2. Scheme comparison results
表 2. 方案对比结果

方案	去中心化	公平性	匿名性	可追溯性	访问控制
Li 等[19]	√	×	√	×	×
Li 等[20]	√	√	×	×	×
Zhu 等[21]	√	√	√	√	×
本方案	√	√	√	√	√

6. 总结

由于传统数据交易无法保证公平性且存在单点故障问题，目前对于数据交易的研究大多基于区块链技术，但是联盟链的隐私性与数据交易要求的可监管性并不完全兼容，且目前层出不穷的数据滥用问题要求数据交易方案支持数据细粒度的访问控制，所以本方案结合联盟链技术与密文策略属性基加密实现了数据的细粒度访问控制，保证只有符合条件的数据买家才可以进行购买和解密，并利用群签名技术对交易双方的身份进行保护，联盟链其他交易节点只能验证签名的正确性，除 PA 和 CA 外的所有节点均无法从交易记录信息中获取交易双方的身份信息，保证了方案的匿名性和可追溯性。通过与其他方案的分析对比，本方案实现了数据交易的数据安全性、公平性、匿名性、可追溯性以及数据细粒度访问控制。

基金项目

国家自然科学基金(62202118，61962009)；贵州省教育厅自然科学研究科技拔尖人才项目(黔教技[2022]073 号)；山东省自然科学基金项目(ZR2021MF086)。

参考文献

[1] Muschalle, A., Stahl, F., Löser, A. and Vossen, G. (2013) Pricing Approaches for Data Markets. In: Castellanos, M., Dayal, U. and Rundensteiner, E.A., Eds., *Enabling Real-Time Business Intelligence*. *BIRTE* 2012. *Lecture Notes in Business Information Processing*, Vol. 154, Springer, Berlin, 129-144. https://doi.org/10.1007/978-3-642-39872-8_10

[2] 刘峰, 杨杰, 齐佳音. 区块链密码学隐私保护技术综述[J]. 网络与信息安全学报, 2022, 8(4): 29-44.

[3] 江东, 袁野, 张小伟, 王国仁. 数据定价与交易研究综述[J]. 软件学报, 2023, 34(3): 1396-1424.

[4] 袁勇, 王飞跃. 区块链技术发展现状与展望[J]. 自动化学报, 2016, 42(4): 481-494.

[5] 曹萌, 于洋, 梁英, 史红周. 基于区块链的大数据交易关键技术与发展趋势[J]. 计算机科学, 2021, 48(S2): 184-190.

[6] Jung, T., Li, X.-Y., Huang, W., Qiao, Z., Qian, J., Chen, L., Han, J. and Hou, J. (2019) AccountTrade: Accountability

against Dishonest Big Data Buyers and Sellers. *IEEE Transactions on Information Forensics and Security*, **14**, 223-234. https://doi.org/10.1109/TIFS.2018.2848657

[7] Sadiq, A., Javaid, N., Samuel, O., Khalid, A., Haider, N. and Imran, M. (2020) Efficient Data Trading and Storage in Internet of Vehicles using Consortium Blockchain. 2020 *International Wireless Communications and Mobile Computing* (*IWCMC*), Limassol, 15-19 June 2020, 2143-2148. https://doi.org/10.1109/IWCMC48107.2020.9148188

[8] Abubaker, Z., Khan, A.U., Almogren, A., Abbas, S., Javaid, A., Radwan, A. and Javaid, N. (2022) Trustful Data Trading through Monetizing IoT Data Using BlockChain Based Review System. *Concurrency and Computation: Practice and Experience*, **34**, e6739. https://doi.org/10.1002/cpe.6739

[9] Chen, F., Wang, J., Jiang, C., Xiang, T. and Yang, Y. (2022) Blockchain Based Non-Repudiable IoT Data Trading: Simpler, Faster, and Cheaper. *IEEE INFOCOM* 2022—*IEEE Conference on Computer Communications*, London, 2-5 May 2022, 1958-1967. https://doi.org/10.1109/INFOCOM48880.2022.9796857

[10] Li, C., Liang, S.Y., Zhang, J., Wang, Q. and Luo, Y. (2022) Blockchain-Based Data Trading in Edge-Cloud Computing Environment. *Information Processing & Management*, **59**, Article ID: 102786. https://doi.org/10.1016/j.ipm.2021.102786

[11] 杨洸. 数字媒体时代的数据滥用: 成因、影响与对策[J]. 中国出版, 2020(12): 3-8.

[12] Sahai, A. and Waters, B. (2005) Fuzzy Identity-Based Encryption. In: Cramer, R., Ed., *Advances in Cryptology—EUROCRYPT* 2005. *EUROCRYPT* 2005. *Lecture Notes in Computer Science*, Vol. 3494, Springer, Berlin, 457-473. https://doi.org/10.1007/11426639_27

[13] Goyal, V., Pandey, O., Sahai, A. and Waters, B. (2006) Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data. *Proceedings of the* 13*th ACM Conference on Computer and Communications Security*, Alexandria, 30 October-3 November 2006, 89-98. https://doi.org/10.1145/1180405.1180418

[14] Chaum, D. and van Heyst, E. (1991) Group Signatures. In: Davies, D.W., Ed., *Advances in Cryptology—EUROCRYPT* '91. *EUROCRYPT* 1991. *Lecture Notes in Computer Science*, Vol. 547, Springer, Berlin, 257-265. https://doi.org/10.1007/3-540-46416-6_22

[15] Li, R., Shen, C., He, H., Gu, X., Xu, Z. and Xu, C.-Z. (2018) A Lightweight Secure Data Sharing Scheme for Mobile Cloud Computing. *IEEE Transactions on Cloud Computing*, **6**, 344-357. https://doi.org/10.1109/TCC.2017.2649685

[16] Szabo, N. (1997) Formalizing and Securing Relationships on Public Networks. *First Monday*, **2**. https://doi.org/10.5210/fm.v2i9.548

[17] 贺海武, 延安, 陈泽华. 基于区块链的智能合约技术与应用综述[J]. 计算机研究与发展, 2018, 55(11): 2452-2466.

[18] Bethencourt, J., Sahai, A. and Waters, B. (2007) Ciphertext-Policy Attribute-Based Encryption. 2007 *IEEE Symposium on Security and Privacy* (*SP* '07), Berkeley, 20-23 May 2007, 321-334. https://doi.org/10.1109/SP.2007.11

[19] 李姝, 赵培培, 于金刚, 王海汀. 基于区块链的数据交易平台的研究与设计[J]. 小型微型计算机系统, 2021, 42(5): 1109-1114.

[20] Li, T., Li, D. and Wang, M. (2022) Blockchain-Based Fair and Decentralized Data Trading Model. *The Computer Journal*, **65**, 2133-2145. https://doi.org/10.1093/comjnl/bxab050

[21] 朱自强, 姚中原, 祝卫华, 赵海鸿, 潘长风, 斯雪明. 匿名可追溯的区块链数据交易方案[J]. 应用科学学报, 2022, 40(4): 653-665.