

DOI: 10.3785/j.issn.1008-973X.2018.11.014

基于区块链智能合约的物联网数据资产化方法

盛念祖^{1,2}, 李芳¹, 李晓风^{1,2}, 赵赫¹, 周桐^{1,2}

(1. 中国科学院合肥物质科学研究院, 安徽合肥 230031; 2. 中国科学技术大学, 安徽合肥 230026)

摘 要: 使用基于区块链智能合约的物联网数据资产化方法解决物联网系统中个人数据难以确权、数据资产的量化跟踪和价值转移无法高效完成等问题. 借助区块链数字指纹将数据所有权和控制权从设备生产商转移至用户, 为个人数据确权; 通过全生命周期管理和数字签名等技术, 将设备状态和数据哈希值存储至区块链, 保证数据的可靠性; 使用智能合约构建去第三方数据交易平台, 保证数据共享的安全性, 便捷地完成数据变现和数据价值转移. 攻击可能性和攻击成功概率的量化分析结果表明, 区块链智能合约技术可以为数据提供防篡改, 消除数据交易过程中的信任问题. 借助区块链智能合约技术能够初步实现物联网数据的资产化, 促进物联网设备的数据价值转移和共享.

关键词: 区块链; 智能合约; 物联网; 数据资产化; 数据确权

中图分类号: TP 311 **文献标志码:** A **文章编号:** 1008-973X(2018)11-2150-09

Data capitalization method based on blockchain smart contract for Internet of Things

SHENG Nian-zu^{1,2}, LI Fang¹, LI Xiao-feng^{1,2}, ZHAO He¹, ZHOU Tong^{1,2}

(1. Hefei Institutes of Physical Science, Chinese Academy of Sciences, Hefei 230031, China;

2. University of Science and Technology of China, Hefei 230026, China)

Abstract: The Internet of Things data capitalization method based on blockchain smart contract was used as a solution to the challenges when ascertaining data ownership, dealing with the inefficiency of quantification and value transfer of digital asset. The data's right was determined by transferring data ownership and control from the equipment manufacturers to the users, with the aid of digital fingerprints stored in the blockchain. The data reliability was ensured by storing the up-to-date device status and hash of the data into the blockchain, which was achieved by the full lifecycle management and digital signatures. A data trading platform without any third-party was established based on smart contract to guarantee the security of data sharing and accomplish the realization and migration of data's value. The quantitative analysis of attack probability and attack success rate shows that the technology of blockchain smart contract provides data tamper resistance and eliminates the trust crisis in data transaction processes. The proposed method realizes the capitalization of data in the Internet of Things initially and may help promote the data value migration and data sharing in the Internet of Things in future.

Key words: blockchain; smart contract; Internet of Things (IoT); data capitalization; determination of data ownership

随着传感器相关技术的发展, 硬件成本不断降低, 各式各样的物联网设备越来越普及. 物

联网大数据具有极高的潜在价值, 是一种重要的资产. 数据资产具有可控制性、可计量性、可靠性、

收稿日期: 2018-01-26. 网址: www.zjujournals.com/eng/fileup/HTML/201811014.htm

基金项目: 国家自然科学基金资助项目(61602435); 安徽省自然科学基金资助项目(1708085QF153).

作者简介: 盛念祖(1993—), 男, 硕士生, 从事计算机应用研究. orcid.org/0000-0003-0634-3015. E-mail: shengnianzu@163.com

通信联系人: 赵赫, 男, 高级工程师. orcid.org/0000-0003-2646-7306. E-mail: zhaoh@hfcas.ac.cn

可变现性^[1-2]等特征.可控制性是指数据资产具有合法的控制权和使用权,可计量性是指数据资产具有可靠的计量方法,可靠性是指数据资产具有可追溯、可证伪、可分析等特点,可变现性是指数据资产具有转化为经济利益的可能性.目前用户普遍使用的物联网数据存储方案主要为设备生产商提供的云存储服务,云存储服务具有价格低廉、部署方便、易于管理等优点,但是存在如下问题.

1)个人数据难以确权^[3].数据持有者通常是设备生产商而不是用户,用户的同意权、知情权、异议权等权利被剥夺,一般只具有查阅数据的权利.

2)数据可靠性差并且不可证伪.设备生产商对云存储数据库具有绝对控制权,能够篡改用户数据,甚至捏造虚假数据,相关研究机构与设备生产商之间难以达成数据信任关系.

3)数据无法共享,不具备可变现性.用户只能使用设备生产商提供的服务支持,无法将数据有偿分享给其他数据收集者.

4)用户隐私难以得到保护.数据与用户个人信息之间通常存在强绑定关系,有泄露个人信息的风险.隐私泄露风险严重影响了用户数据共享的积极性,破坏了数据资产的变现能力.

区块链是一种无需信任、去中心化的分布式账簿技术,具有透明可信、防篡改、可追溯、高可靠性等特点,有望解决物联网发展中的大数据管理、信任、安全和隐私等关键问题^[4].国内外学者和相关研究机构已经开展了将区块链技术应用到物联网和大数据安全的研究工作.Zyskind等^[5]为实现数据隐私保护和数据共享等功能,使用区块链技术和分布式哈希表(distributed hash table, DHT)存储方法,构建了用户数据权限管理系统.Azaria等^[6]利用智能合约构建分布式管理系统,管理各中心化的电子医学数据库,促进了医学机构间的数据共享.上述研究对如何利用区块链实现数据共享和隐私保护进行了研究探讨,但是没有涉及如何将数据变现,实现数据资产化.Zhang等^[7]提出了一种新型电子商务模式,通过区块链兑换物联网币与物联网设备进行交易,实现去第三方的智能设备和物联网数据交易,但是不能保证数据可靠性,个人数据的价值转移存在局限性.IBM公司在物联网白皮书^[8]中指出,通过利用区块链技术消除信任需求、促进交易处理和设备交互可以实现“去中心化自治物联网”,但是没有提及如

何体现设备数据价值和实现设备数据资产化.

本研究基于区块链技术提出物联网数据资产化方法.通过设备签名传输协议统一上传数据的方式,提供数据价值计量方法,将数据保存在用户本地或去中心化数据库,为个人数据确权;通过设备全生命周期管理合约,为物联网设备提供从出厂开始的全生命周期事件存储^[9],为设备数据提供证伪性,增强数据可靠性;通过物联网数据订单合约,为研究机构等数据需求者提供价值交换的数据收集方法,实现设备数据资产化.

1 技术基础

区块链技术具有分布式对等、链式数据块、防伪造、防篡改、透明度高和可靠性高等典型特征^[4],通过分布式节点验证和共识机制,解决了拜占庭将军问题^[10],无需信任单个节点就可以构建去中心化可信任系统^[11].将区块链技术优势与物联网相融合,有助于保障用户权利,提升数据可靠性,促进数据互通,实现高效、稳定、可靠的物联网数据资产化方法.

智能合约是一系列部署于区块链上的具有状态的链上代码,具有不可篡改、去中心化、自治化等特性.代码和状态信息存储于区块链上,通过交易事件触发并且在所有节点上运行,在所有节点产生共识结果后,将共识结果引起的状态信息变化记录在区块链上.智能合约的代码和状态通过区块链获得,可复制可验证,具有不可篡改、透明、可信等区块链的一般特性;代码在所有节点上共同运行,不存在中心服务器,具有去中心化特性;智能合约根据事先约定好的触发条件及运行机制控制管理智能资产,不需要任何第三方机构的控制,具有自治化的特性.使用智能合约可以消除中间平台等第三方机构,转移、存储及发送以太币及代币等价值物,对链上资产进行管理,实现可编程自动化的资产管理系统^[11-13].

比特币是最早也是最具影响力的区块链技术的应用.比特币区块链的脚本语言在一定程度上能够实现智能合约,但是缺少图灵完备性,无法实现物联网数据资产化方法的设计.新建一个区块链是有风险的,特别是在初期总计算量较小的情况下,恶意计算量居优的攻击者可以自由生成分支,产生恶意数据.稳定的区块链的总计算量非常大,攻击者很难产生计算量优势^[14],风险比

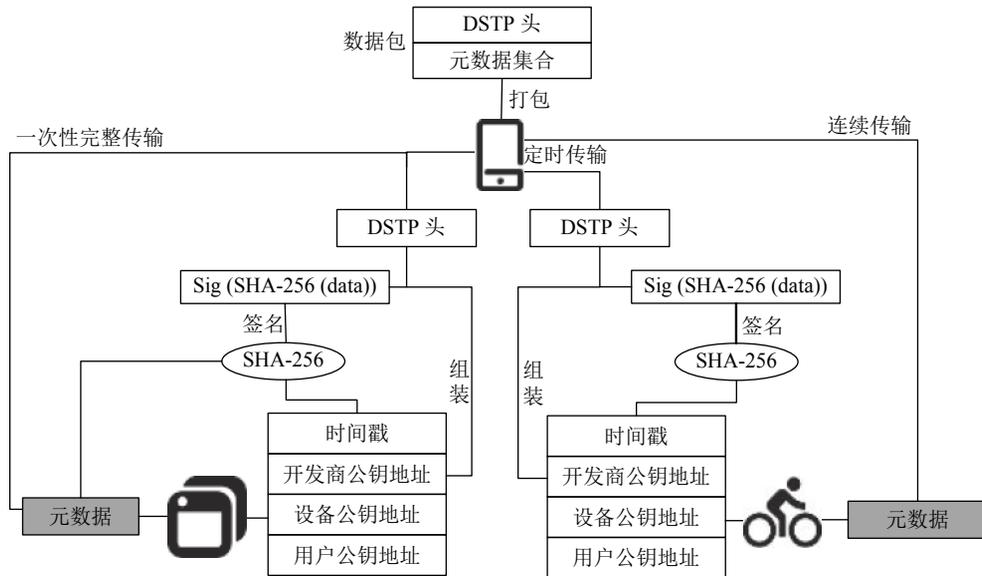


图 2 设备签名传输协议下的数据传输流程

Fig.2 Data transfer flow under device signature transfer protocol

2.2 节中的 Protocol 3 将 Merkle 树的根节点 (MerkleRoot) 保存在设备全生命周期管理合约中. ω 的表达式为

$$\omega = \mu \frac{(\text{cost}_{\text{save}} + \delta \text{cost}_{\text{transaction}})}{\text{pkgPrice}}; \quad \mu > 1. \quad (1)$$

式中: $\text{cost}_{\text{save}}$ 为将 Merkle 树根节点保存到设备全生命周期管理合约中所产生的花费; $\text{cost}_{\text{transaction}}$ 为使用物联网数据订单合约交易时, 数据收集者提交回执所产生的花费; pkgPrice 为单一数据包的价值; μ 和 δ 为系数, 根据数据不同和订单详情动态设置.

2.2 设备全生命周期管理合约

设备全生命周期管理合约 (device life-cycle manage contract, DLMC) 要求设备生产商、用户和设备通过公钥地址来完成出厂、绑定等协议, 并且根据智能合约中存储的数据, 为数据提供真实性验证, 保证数据资产可靠性.

设备出厂后, 设备生产商为设备生成一对公私钥对, 保存在设备中. 调用智能合约的接口, 将 $\text{compound}(\text{pk}_{\text{add}}^{\text{m}}, \text{pk}_{\text{add}}^{\text{d}})$ 二元组保存到智能合约中, 以设备地址为索引生成一条出厂日志. 如 Protocol 1 所示, m 为设备生产商, u 为用户, d 为设备. map 为利用公钥地址进行映射查找操作, pk_{sig} 为数字签名操作, pk_{add} 为公钥地址, $\text{event}_{\text{leave}}$ 为生成的出厂日志, indexed 表示使用该条属性作为索引用于搜索日志.

Protocol 1 Leave Factory

- 1 **Procedure** Leave Factory($m, \text{contract}, d$)
- 2 m executes :
- 3 generate a keyPair $\rightarrow d$
- 4 $\text{pk}_{\text{sig}}^{\text{m}}(\text{pk}_{\text{add}}^{\text{m}}, \text{pk}_{\text{add}}^{\text{d}}) \rightarrow \text{contract}$
- 5 contract executes:
- 6 **if** $\text{map}(\text{pk}_{\text{add}}^{\text{m}}, \text{pk}_{\text{add}}^{\text{d}}) = \phi$ **then**
- 7 add compound $(\text{pk}_{\text{add}}^{\text{m}}, \text{pk}_{\text{add}}^{\text{d}})$
- 8 **end if**
- 9 **return** $\text{event}_{\text{leave}}(\text{pk}_{\text{add}}^{\text{m}}, (\text{indexed})\text{pk}_{\text{add}}^{\text{d}})$
- 10 **end procedure**

用户将智能合约生成的随机数 nonce 和用户公钥地址发送给设备, 设备使用设备私钥对用户公钥地址和随机数 nonce 进行签名. 智能合约验证设备签名后, 改变绑定状态二元组 $\text{compound}(\text{pk}_{\text{add}}^{\text{u}}, \text{pk}_{\text{add}}^{\text{d}})$, 以设备地址为索引生成一条绑定日志 ($\text{event}_{\text{bind}}$), 如 Protocol 2 所示. 需要注意的是, 当设备为健身房、医院等公共场所中的共享设备时, 用户与设备关系更换频繁, 不需要进行绑定, 由设备签名传输协议保证可靠性.

Protocol 2 User Bind to the Device

- 1 **Procedure** BindDevice($u, \text{contract}, d$)
- 2 u executes :
- 3 $\text{pk}_{\text{add}}^{\text{u}} \rightarrow d$
- 4 d execute :
- 5 $\text{compound}(\text{pk}_{\text{add}}^{\text{m}}, \text{pk}_{\text{add}}^{\text{d}}) \rightarrow \text{contract}$
- 6 contract executes:
- 7 **if** $\text{map}(\text{pk}_{\text{add}}^{\text{m}}, \text{pk}_{\text{add}}^{\text{d}}) \neq \phi$ **then**

```

8      generate a nonce → d
9      end if
10     d executes :
11     pksigd(pkaddu, nonce) → contract
12     contract executes:
13     change compound(pkaddd, pkaddu)
14     return eventbind(pkaddu, (indexed)pkaddd)
15 end procedure

```

用户获取一定量的数据后,从智能合约中取出上一次数据的哈希值和当前哈希编号,对二者和当前数据包列表(Pkg)的Merkle树根节点进行哈希,得出当前哈希值,保存到智能合约中.智能合约会保存此次哈希值和区块链时间戳,如Protocol 3所示,compound.pk_{add}^u为设备绑定日志中与设备公钥对应的用户公钥,MerklRoot_{Pkg}为数据包列表的Merkle树根节点.根据数据包中的哈希编号可以从智能合约中获取当前数据包的哈希值(hash_{Pkg})和上一次的哈希值(hash_{lastPkg}),验证当前数据包是否被篡改.

Protocol 3 Save Data Hash

```

1 Procedure SaveHash(u, contract)
2   u executes :
3   generate a MerklRootPkg
4   pksigu(pkaddm, pkaddd) → contract
5   contract executes:
6   if map(pkaddm, pkaddd) ⇒ compound.pkaddu
7     = pkaddu then
8     (hashlastPkg, Index) → u
9   end if
10  u executes :
11  SHA-256(hashlastPkg, Index, MerklRootPkg)
12     → hashPkg
13  hashPkg → contract
14  return Index
15 end procedure

```

回收并且销毁设备后,建议设备生产商访问智能合约,将设备状态设置为已失效,以设备私钥地址为索引生成一条销毁日志.

2.3 物联网数据订单合约

设备签名传输协议能够转移数据所有权,提供统一数据格式,实现数据资产的可控制性和可计量性.设备全生命周期管理合约通过存储设备出厂后的全生命周期证明,为数据可靠性提供背书.基于前两者,本方法能够实现物联网数据订单合约(Internet of Things data order contract, IoT-

DOC),为数据提供可变现性,具体流程如图3、4所示.

如图3所示,数据收集者生成一对以太坊公私钥对,以此公钥地址在IoT-DOC上进行注册;当数据收集者需要某种设备所产生的数据时,在IoT-DOC上发布一条订单信息.如图3(c)所示为物联网数据订单合约的地址和已存入的定金数量.如图4所示,用户获取到设备后,首先生成一对以太坊公私钥对;然后与设备进行绑定;用户以拥有的设备查看可提交的订单,并且向该订单提供的地址发送数据包,由于数据较大,一般不通过智能合约进行传递,而是通过http协议上传.

如图5所示,设备出厂时,设备开发商会在智能合约中注册设备,生成出厂日志;用户绑定设备时,在智能合约中生成绑定日志;数据收集者在获得数据后,根据出厂日志和绑定日志验证数据来源,根据数据DSTP头和DLMC合约验证数据的可靠性,在确定数据可靠性后,向IoT-DOC合约发送回执,确认收到数据.只有当数据收集者向合约发送回执后,合约才会将数据定金转移到用户账户.此时合约会产生以设备地址为索引的提交日志,日志信息如图5(c)所示.如图4(d)所示,用户在收到日志提示后,获知数据收集者已经履行合约并且完成交易,可以进行下一笔数据传输交易.

物联网数据资产化方法使用mist钱包模仿订单提交流程,使用以太坊提供的web3.js库和web3j库访问智能合约.web3.js库是JavaScript库,

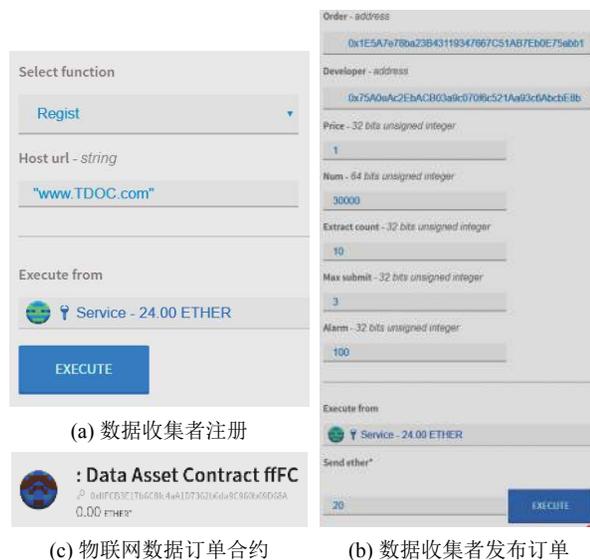


图3 物联网数据订单合约操作流程

Fig.3 Operation of Internet of Things data order contract



图 4 物联网数据资产化方法的移动客户端实现
Fig.4 Implementation of capitalization method of IoT data on user mobile client

主要用于在服务器接收数据并完成数据验证后,向区块链提交收到数据的回执. web3j 库是 Java 库,主要用于在 Android 设备访问智能合约.

3 攻击手段分析

3.1 盗取私钥

盗取私钥指攻击者通过获取用户、设备生产商、数据收集者、设备的私钥,伪造四者的身份. 用户、设备生产商、数据收集者的私钥由三者分别保存,被盗取的可能性比较小. 如果私钥被盗取,用户可以尽快转移已有资产并更换公私钥对以避免损失;设备生产商需要公布私钥被盗取声明,避免新的出厂日志被视为合法;数据收集者



图 5 智能合约生成的设备生命周期日志及交易日志

Fig.5 Device life-cycle log and transaction log generated by smart contract

可以尽快终止数据收集订单,回收剩余定金,并更换公私钥对. 设备私钥保存在设备硬件中,可以在硬件中加入读保护机制避免私钥被盗取.

3.2 伪造签名

攻击者通过揭示签名者私钥等方法伪造签名,使用伪造的签名获取利益. 当 ECDSA 私钥位数分别为 160、224、256 bit 时,使用每秒 1 百万条指令的计算机去破解,破解时间分别为 10^{12} 、 10^{24} 、 10^{28} a. 这表明破解 256 位的 ECDSA 私钥需要巨大的计算量^[18]. 即使使用一台 1000 万美元造价的专业计算椭圆曲线对数的机器,也需要一个月的时间才能计算出素数阶为 2^{120} 的椭圆曲线离散对数^[15],本区块链网络签名算法使用的椭圆曲线算法素数阶略小于 2^{256} ,几乎不可能通过硬件计算.

3.3 数据篡改

攻击者篡改数据后,根据数据的篡改结果修改区块链上保存的数据哈希值,期望以虚假数据通过数据收集者的数据可靠性验证.

区块链技术使用工作量证明 (proof-of-work, PoW) 共识机制保证数据一致性. 每一个合法的区块都需要补增一个随机值使区块哈希值拥有一定数量的前缀 0. 随着前缀 0 的数量提升,矿工找到随机值的难度也会呈指数上升,寻找随机值所花费的 CPU 算力称为工作量. 最先找到随机值的区块会加入区块链成为最新的区块,拥有最多区块的链包含工作量最大,称为主链. 前缀 0 的数量随着全网算力的提升而增加,保证找到随机值的速

度为某一个预定的平均数^[19].

常见的区块链攻击手段为双重支付攻击^[20]和自私挖矿攻击. 攻击者的节点算力超过 50% 全网算力时可以实现双重支付攻击, 篡改最近的区块并控制未来的区块; 攻击者的算力大于 25% 全网算力时可以实现自私挖矿攻击, 大概率获取新区块奖励并破坏挖矿的公平性^[21].

本系统面临的主要威胁不是双重支付攻击和自私挖矿攻击, 而是区块数据被篡改. 假设攻击者试图篡改最新区块前的第 h 块区块的数据, 攻击者必须修改区块哈希值, 并且重新计算往后所有区块的哈希值. 假设当前全网诚实节点算力为每秒 p 次哈希值计算, 当前计算难度下区块哈希值含有 g 个前缀二进制 0. 攻击者是新加入的算力, 算力大小为每秒 q 次哈希值计算. 为攻击者主要在计算过去的区块不会对新区块的产生速度造成影响, 所以新区块哈希值的计算难度不会增加. 为了简化计算, 假设没有新的节点参加, 每秒中诚实节点获得新区块的概率为 $p/2^g$, 攻击者获得新区块的概率为 $q/2^g$. 攻击者与诚实节点的初始高度差 $z_0 = h$, 设 z_i 为第 i 秒的高度差, 高度差 z_{i+1} 每秒变化的可能性分为 3 种情况.

事件 X_1 : 攻击者没有生成区块, 诚实节点生成区块, z_i 加 1, 概率 $P_1 = \frac{p}{2^g} \left(1 - \frac{q}{2^g}\right)$.

事件 X_2 : 攻击者生成区块, 诚实节点没有生成区块, z_i 减 1, 概率 $P_2 = \frac{q}{2^g} \left(1 - \frac{p}{2^g}\right)$.

事件 X_3 : 两者都生成区块或都没有生成区块时, z_i 不变, 概率 $P_3 = 1 - P_1 - P_2$.

每秒钟高度差 z_{i+1} 的变化概率分布符合多项分布:

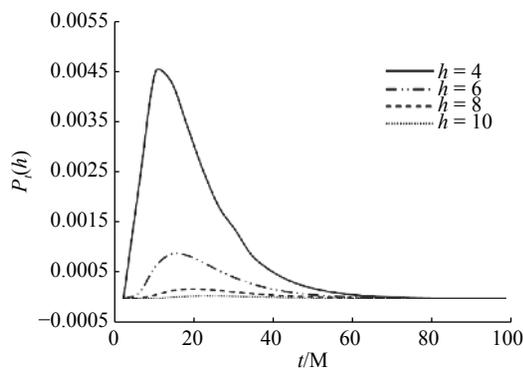
$$z_{i+1} = \begin{cases} z_i + 1, & \left(P_1 = \frac{p}{2^g} \left(1 - \frac{q}{2^g}\right)\right); \\ z_i - 1, & \left(P_2 = \frac{q}{2^g} \left(1 - \frac{p}{2^g}\right)\right); \\ z_i, & (P_3 = 1 - P_1 - P_2). \end{cases} \quad (2)$$

当 $z_{i+1} = -1$ 时, 攻击者成功追上了诚实节点, 可以发布区块链, 数据篡改成功. 在 t 秒内, 会出现 t 次高度变化事件, 设 n 为 X_1 发生的次数; 当篡改成功时, X_2 至少发生 $(n+h+1)$ 次, 设 j 为事件 X_2 实际发生次数与最少发生次数的差值, 则 X_2 实际发生次数为 $(n+h+1+j)$; X_3 发生的次数基于前两者发生的次数, 为 $(t-2n-h-1-j)$, 其中 $n \in [0, (t-1-h)/2]$, $j \in [0, t-2n-h-1]$. 在 t 秒内, 攻击者追上诚实节点的概率为

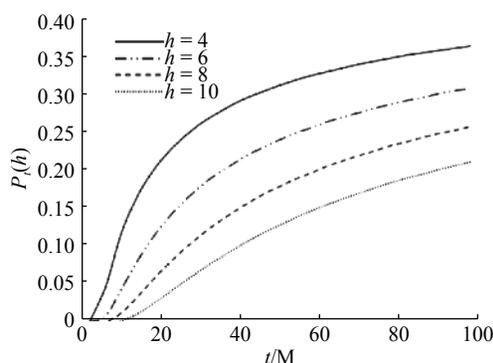
$$P_t(h) = \sum_{n=0}^{n_{\max}} \sum_{j=0}^{j_{\max}} \left(\frac{t!}{n!(h+n+1+j)!(t-2n-h-1-j)!} \cdot P_1^n P_2^{(h+n+1+j)} P_3^{(t-2n-h-1-j)} \right). \quad (3)$$

根据式 (3) 作图得到攻击者篡改区块数据成功概率, 如图 6 所示, 纵坐标为成功概率 $P_t(h)$, 横坐标为时间 t , 单位为 M , M 为诚实节点新生成一个区块的平均时间, 计算方法为 $M = 2^g/p$.

由图 6 可知, 攻击者成功的概率随着篡改区块深度 h 的增加越来越低. 当攻击者的算力小于诚实节点时, 攻击者需要先计算 h 个区块的哈希值, 篡改成功概率先逐渐增加后逐渐减小, 直至接近 0; 当攻击者的算力大于或等于诚实节点时, 篡改成功的概率会逐渐增加, 但是增加幅度越来越小. 攻击者的算力与诚实节点算力相等时, 篡改区块成功率较高, 在一段时间后, 成功概率逼近 35%; 攻击者算力为诚实节点算力的 50% 时, 篡改前 4 个区块的最大篡改成功概率小于 0.5%. 比特币和以太坊的算力强大, 攻击者试图达到与诚实节点相同的算力水平几乎不可能; 即使有能力达到该算力水平, 获得区块挖掘的奖励也远远高于篡改数据的收益^[22].



(a) 攻击者算力为诚实节点算力 50% 情况下



(b) 攻击者算力为诚实节点算力相等情况下

图 6 攻击者篡改区块数据成功概率

Fig. 6 Success probability of tampering blockchain by attackers

3.4 数据造假

数据造假指用户改变设备时钟和设备时间戳,期望短时间内生成大量虚假数据以牟取利益.用户在DLMC合约中保存设备数据哈希值时,也保存了区块时间戳,数据收集者可以对比区块时间戳与数据时间戳,验证用户是否伪造数据.

3.5 设备造假

设备造假指设备生产商不生产设备,只产生设备公私钥对,利用设备公私钥上传伪造数据.设备生产商作为设备生产者,是具有实体的企业,对应区块链上的出厂日志上有相应的实体生产记录.数据收集者可以查阅相关设备生产商生产记录和信誉记录,或者通过检查设备出厂日志和绑定日志来排查数据,判断该设备生产商是否进行设备造假.

3.6 订单欺骗

订单欺骗指数据收集者接受数据后,不向IoT-DOC提交回执,达到收集数据又不产生支付的目的;或者数据收集者自己给自己提交回执,造成交易数量比较多的假象.订单欺骗不可行的原因是订单中有比较多的数据相关信息,如提交数据总数、提交人数等,订单欺骗成本比较高;用户在交易结果没有达到预期时,不会继续分享数据,数据收集者只能收集到用户第一次上传的数据,订单欺骗得不偿失.

4 方法可行性讨论

基于区块链智能合约的物联网数据资产化方法旨在实现数据资产化,保护个人用户权益.下面从数据确权、保障数据可靠性、提供数据变现、保护用户隐私等4个角度,讨论该方法能否达到要求.

4.1 所有权

设备签名传输协议将数据收集在用户手中,保证用户的数据所有权,实现数据资产可控制性,为个人数据确权.用户可以将数据保存在本地,也可以加密后存储在分布式数据库或者其他数据库.处理数据时,用户可以选择无偿发送给服务提供商,获取服务提供商提供的数据分析服务,也可以通过物联网数据订单合约,将数据有偿发送给数据收集者,实现数据变现.

4.2 可靠性

如果用户选择无偿分享数据,用户不需通过DLMC合约将设备信息和数据包哈希值保存在区块链,研究机构可以通过验证设备签名,保证基

本的数据可靠性.如果用户选择有偿分享数据进行数据变现,必须将设备信息及数据包哈希值保存到区块链,通过区块链验证数据的哈希值和时间戳,充分保证数据可靠性.

4.3 共享性

数据通过物联网数据订单合约实现有偿分享,便于用户将手中数据变现,实现数据价值.用户可以通过出售数据获取利益,激发用户生成数据和分享数据的动力;相关研究机构可以通过发布订单获取数据,扩大研究数据范围和数据量.

4.4 隐私性

数据中不包含用户个人信息(例如手机号、邮箱等).数据收集者收集到数据后,只能根据数据中的用户公钥地址判断数据来源是否一致,不能获知用户的隐私信息.对于不安全或者已经泄露个人信息的公钥地址,用户可以更换公钥地址避免个人信息进一步泄露.设备不同时,用户可以选择使用不同的公私钥对,避免公钥和个人信息关联.此外,对于数据归属一致性要求(如健康设备数据要求归属一致才具有分析价值)不高的设备,用户可以通过定期更换公私钥对的方法增强隐私性.

5 结 语

本研究基于区块链智能合约技术、数字签名技术,提出以设备签名传输协议、设备全生命周期管理合约、物联网数据订单合约三者为核心的数据生产、管理、变现方法,初步实现物联网数据的资产化.本文方法能够保护用户的隐私权和数据所有权,维护数据真实性,体现数据本身所具有的价值,构建无需第三方机构保证的安全可信数据交易平台.用户、设备生产商与数据收集者达成良好连接,有利于促进三者之间的数据分享,为各大研究机构提供更加广泛的数据.未来的工作将致力于利用闪电网络技术^[23]、plasma可扩容自主智能合约^[24]及联盟链^[25]等方法进一步降低数据共享成本,建立分布式数据仓库,提供分布式数据分析服务.

参考文献 (References):

- [1] 齐爱民, 盘佳. 数据权、数据主权的的确立与大数据保护的基本原则[J]. 苏州大学学报: 哲学社会科学版, 2015(1): 64-70.
QI Ai-min, PAN Jia. Data right, the establishment of data sovereignty and the basic principle of big data protection[J]. *Journal of Soochow University*:

- Philosophy and Social Science Edition**, 2015(1): 64–70.
- [2] 中国资产评估协会. 中国资产评估准则: 2005[M]. 北京: 经济科学出版社, 2005: 41–45.
- [3] 彭云. 大数据环境下数据确权问题研究[J]. 现代电信科技, 2016, 46(5): 17–20.
PENG Yun. Research on authenticating data rights in big data environment[J]. **Modern Science and Technology of Telecommunications**, 2016, 46(5): 17–20.
- [4] 中国电子技术标准化研究院. 中国区块链与物联网融合创新应用蓝皮书[R/OL]. (2017-09-13)[2017-12-20]. <http://www.cesi.ac.cn/images/editor/20170913/20170913145041632.pdf>.
- [5] ZYSKIND G, NATHAN O, PENTLAND A. Decentralizing privacy: using blockchain to protect personal data [C] // **IEEE Security and Privacy Workshops**. San Jose: IEEE, 2015: 180–184.
- [6] AZARIA A, EKBLAW A, VIEIRA T, et al. MedRec: using blockchain for medical data access and permission management [C] // **International Conference on Open and Big Data**. Vienna: IEEE, 2016: 25–30.
- [7] ZHANG Y, WEN J. An IoT electric business model based on the protocol of bitcoin[C] // **International Conference on Intelligence in Next Generation Networks**. Paris: IEEE, 2015: 184–191.
- [8] IBM Institute for Business Value. Device democracy-saving the future of the Internet of Things [R/OL]. (2017-10-02)[2017-12-20]. <http://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=PM&subtypeXB&htmlfidGBE03620USEN>.
- [9] 赵赫, 李晓风, 占礼葵, 等. 基于区块链技术的采样机器人数据保护方法[J]. 华中科技大学学报: 自然科学版, 2015, 43(s1): 216–219.
ZHAO He, LI Xiao-feng, ZHAN Li-kui, et al. Data integrity protection method for microorganism sampling robots based on blockchain technology [J]. **Journal of Huazhong University of Science and Technology: Natural Science Edition**, 2015, 43(s1): 216–219.
- [10] LAMPORT L. The Byzantine generals problem [J]. **ACM Transactions on Programming Languages and Systems**, 1982, 4(3): 382–401.
- [11] 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.
YUAN Yong, WANG Fei-Yue. Blockchain: the state of the art and future trends [J]. **Acta Automatica Sinica**, 2016, 42(4): 481–494.
- [12] 钱卫宁, 邵奇峰, 朱燕超, 等. 区块链与可信数据管理: 问题与方法[J]. 软件学报, 2018, 29(1): 150–159.
QIAN Wei-ning, SHAO Qi-feng, ZHU Yan-chao, et al. Research problems and methods in blockchain and trusted data management [J]. **Journal of Software**, 2018, 29(1): 150–159.
- [13] Ethereum White Paper. A next-generation smart contract and decentralized application platform [R/OL].(2015-11-12)[2017-12-20]. <https://github.com/ethereum/wiki/wiki/WhitePaper>.
- [14] CONOSCENTI M, VETRÒ A, MARTIN J C D. Blockchain for the Internet of Things: a systematic literature review [C] // **Computer Systems and Applications**. Agadir: IEEE, 2017: 2161–5330.
- [15] JOHNSON D, MENEZES A, VANSTONE S. The elliptic curve digital signature algorithm (ECDSA) [J]. **International Journal of Information Security**, 2001, 1(1): 36–63.
- [16] U.S. Department of Commerce. Secure hash standard-federal information processing standards publication 180–4 [S/OL].[S. 1.]: Federal Information Processing Standards Publication, 2012: 21–23[2017-12-20]. <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf>.
- [17] MERKLE R C. A digital signature based on a conventional encryption function [C] // **A Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology**. Santa Barbara: CRYPTO, 1987, 293, 369–378.
- [18] KHALIQUE A, SINGH K, SOOD S. Implementation of elliptic curve digital signature algorithm [J]. **International Journal of Computer Applications**, 2011, 2(2): 21–27.
- [19] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system[EB/OL].(2018)[2017-12-20]. <https://bitcoin.org/bitcoin.pdf>.
- [20] TSCHORSCH F, SCHEUERMANN B. Bitcoin and beyond: a technical survey on decentralized digital currencies [J]. **IEEE Communications Surveys and Tutorials**, 2016, 18(3): 2084–2123.
- [21] EYAL I, SIRER E G. Majority is not enough: bitcoin mining is vulnerable [C]// **International Conference on Financial Cryptography and Data Security**. Christ Church: International Financial Cryptography Association, 2014: 436–454.
- [22] ANTONOPOULOS A M. Mastering Bitcoin[M/OL]. [S.1.]: O'Reilly Media, 2015: 210–218 [2017-12-20]. <http://chimera.labs.oreilly.com/books/1234000001802/index.html>.
- [23] POON J, DRYJA T. The bitcoin lightning network: scalable off-chain instant payments[EB/OL]. [2016-01-14]. <https://lightning.network/lightning-network-paper.pdf>.
- [24] POON J, BUTERIN V. Plasma: scalable autonomous smart contracts[EB/OL]. [2017-08-11]. <http://plasma.io/plasma.pdf>.
- [25] KANG J, YU R, HUANG X, et al. Enabling localized peer-to-peer electricity trading among plug-in hybrid electric vehicles using consortium blockchains [J]. **IEEE Transactions on Industrial Informatics**, 2017, 13(6): 3154–3164.