文章编号:1671-6833(2024)01-0054-10

基于双区块链的产品溯源系统研究与实现

韩妍妍[1,2], 魏万奇[1], 窦凯丽[3], 张 齐[1]

(1.北京电子科技学院 电子与通信工程系,北京 100070;2.西安电子科技大学 通信工程学院, 陕西 西安 710071;
3.北京电子科技学院 网络空间安全系,北京 100070)

摘 要:现有区块链溯源系统由于区块信息存储容量有限,多采用区块链与云存储相结合的方式,并没有从根本上解决区块链溯源信息存储和数据泄露的问题。针对此问题构建双区块链模式:查询链完成溯源信息的上传,实现溯源系统的基本功能;存储链结合星际文件系统保障数据完整安全。在此基础上,利用改进后的容量证明共识算法保证底层的安全性,在减少能源消耗的同时满足区块链溯源系统的应用需求。测试结果表明:系统上传平均速度可达 80.66 M/s,下载平均速度可达 90.75 M/s,具有良好的上传和下载性能。区块链每秒交易量达到 125.88 笔,单次交易承载量可以满足溯源系统的区块交易需求。

关键词:双区块链; 容量证明; 星际文件系统; 溯源系统; 数据可靠存储

中图分类号: TP311 **文献标志码**:A **doi**:10.13705/j.issn.1671-6833.2024.01.005

随着信息技术的快速普及,溯源系统在追溯产品信息、处理安全事故等方面发挥了重要的作用[1]。近年来,物联网、区块链技术的快速发展为溯源系统的迭代更新带来了新的可能。中国政府早在 2019 年就提出要探索“区块链+”,希望能在商品防伪等领域为人们提供便捷的服务[2]。

中国互联网公司如阿里巴巴、腾讯等也都展开了相关研究,国内外学者都有进一步研究和应用[3]。Li 等[4]在介绍区块链架构模型和关键技术的基础上,设计了一种产品追溯模型,但该模型无法保证源头数据的真实可靠。Baralla 等[5]在一种工厂到家庭(factory to family,F2F)的新型营销模型基础上,运用区块链设计了一种农产品追溯系统,并利用二维码(quick response code,QR code)验证了产品的部分信息,但系统数据访问率较低。张国英等[6]通过智能合约建立起溯源数据模型以记录溯源信息,进而确保溯源信息不可篡改,但是由于区块存储容量有限,没能做到批量数据的存储。刘雅东[7]基于以太坊搭建了一个溯源信息存储平台,虽然提高了感知数据的安全性,但基于工作量证明(proof of work,PoW)的共识机制造成了大量的能源消耗和算力资源浪费。高琰晨[8]针对物流行业成员间的激励契约设计了一种基于区块链的物流信息追溯模型,但该模型并不能满足实际生产的需求。

以上方案虽然利用区块链技术解决了产品追溯问题,但是现有解决方案大都采用单区块链实现产品溯源。单链方案中的大多数也都集中在解决数据的防篡改和去信任化问题上,而忽视了源头数据的真实可靠性。此外,单一的公有链或者私有链也因为工作效率较低而难以满足实际生产的需求。面对单链方案的不足,不少学者开始思考采用双链模式来解决这些问题。Leng 等[9]利用一条区块链存储所有公用数据,另一条区块链记录敏感数据,不仅解决了数据泄露问题,保障了数据安全性,而且提高了数据访问效率。刘家稷等[10]在文献[9]的基础上利用公有链和私有链构建了一种防伪溯源系统,在保证溯源信息真实可靠的同时,做到了以高效率低成本的方式运行。以上解决方案虽然弥补了单链方案的不足,但仍没有解决溯源系统数据大批量存储的问题。而星际文件系统(interplanetary file system,IPFS)作为一种分布式文件系统,为该问题提供了一种可行的解决方案。

收稿日期:2023-07-31;**修订日期**:2023-08-27

基金项目:中央高校基本科研业务费专项资金(328202233);北京高校高精尖学科建设项目(3201023)

作者简介:韩妍妍(1982—),女,黑龙江哈尔滨人,北京电子科技学院副研究员,博士,主要从事秘密共享、信息安全管理、区块链研究,E-mail:hyy@besti.edu.cn。

引用本文:韩妍妍,魏万奇,窦凯丽,等. 基于双区块链的产品溯源系统研究与实现[J]. 郑州大学学报(工学版),2024,45(1):54-63. (HAN Y Y, WEI W Q, DOU K L, et al. Research and implementation of product traceability system based on dual blockchain[J]. Journal of Zhengzhou University (Engineering Science),2024,45(1):54-63.)

高文涛等[11]基于联盟区块链和 IPFS 技术,提出一种去中心化的音乐共享模型,实现了安全透明的音乐数据存储和共享。冯国富等[12]提出一种区块链和 IPFS 技术相结合的水产品交易溯源模型,在保证交易数据安全的同时,解决了传统水产品交易溯源系统存在的数据共享难、产品追溯难等问题。曾卫民[13]提出一种 IPFS 和区块链技术相结合的溯源方案模型,解决了传统食品供应链隐私泄露、大批量数据存储难等问题。

虽然上述解决方案将区块链与 IPFS 技术相结合,实现了产品的可追溯以及数据的可靠存储与共享。但以上方案大都采用权益证明(proof of stake, PoS)共识机制、代理权益证明(delegated proof of stake,DPoS)共识机制、PoW 共识机制以及实用拜占庭容错(practical Byzantine fault tolerance,PBFT)共识机制。其中,PoS 共识机制和 DPoS 共识机制由于部分主节点掌握着区块记账权,过程十分烦琐,影响了区块链的进一步应用;PBFT 共识机制虽然可以在保持安全性的条件下实现(N-1)/3 的容错性,但是 PBFT 共识机制的时间复杂度为 $O(N^2)$,因而该机制的扩展性较差;PoW 共识机制因为要找到满足一定条件的 Nonce 值,所以需要进行大量的哈希运算,这造成了电力和各种算力资源的浪费。针对这些问题,容量证明(proof of capacity,PoC)共识机制应运而生。PoC 将 PoW 中需要付出的计算资源改为付出一定数量的存储空间,从而减少 PoW 共识工作过程造成的浪费。目前,国内外对 PoC 共识的应用还相对较少。

综合上述分析,本文提出一种基于双区块链的产品溯源方案。利用区块链和星际文件系统实现交易信息的可追溯、溯源文件的可靠存储和共享,利用改进后的 PoC 共识保障溯源交易的透明度和可靠度。

本文所做主要工作如下。

(1)提出一种改进型 PoC 共识机制。针对传统 PoW 共识耗能高的缺点,提出一种改进型 PoC 共识,能够对抗其他 PoC 的多挖操作,功耗远远低于比特币的资源开销。

(2)基于双区块链模式的产品溯源系统设计与实现。系统实现了用户注册登录、溯源信息上传、查询以及溯源文件提取等功能,并且针对传统区块链不适合存储大文件的情况,使用存储链和 IPFS 技术实现了区块链的可靠查询和溯源信息的大批量存储。

(3)双区块链产品溯源系统的测试与分析。对各模块进行功能测试,对 PoC 共识机制的优势通过系统溯源的交易量,文件提取速度、稳定性等方面进行测试与分析,并验证了文件存储准确性。

1　相关技术

1.1　区块链技术

区块链本质上是一个分布式账本,具备去中心化、防篡改、可追溯等特性。区块链中的节点通过竞争打包生成新区块,并按照时间的顺序以链表的形式进行连接组成区块链。其中的每个区块都包含上一区块的哈希值、交易信息和随机数等信息[14]。

1.2　PoC 共识算法

PoC 共识算法整个过程分为两个阶段[15],这两个阶段由证明者和验证者来完成。算法第 1 阶段,验证者生成一个随机大小的文件,然后发送给证明者进行存储,验证者只需存储文件的部分内容;算法第 2 阶段,验证者要求证明者发送一个指定位置的文件片段,证明者为保证能正确回答问题,必须保存整个文件,故而验证者只需要保存一段数据即可。

1.3　IPFS 技术

IPFS 通过引入基于内容的寻址方式,采用去中心化的对等网络进行信息交换,是一种分布式的数据存储系统,可广泛应用于数据共享和数据传播领域,并确保数据的真实性[16]。

2　系统原型设计

2.1　系统溯源流程

系统溯源流程如图 1 所示。其中,自身回溯链(my retrospective blockchain,MyRTP)作为查询链便于客户进行相关溯源信息的查询;自身存储链(my storage blockchain, MySTO)作为存储链进行溯源信息文件的存储,并做权限下载,确保数据信息的安全流通。

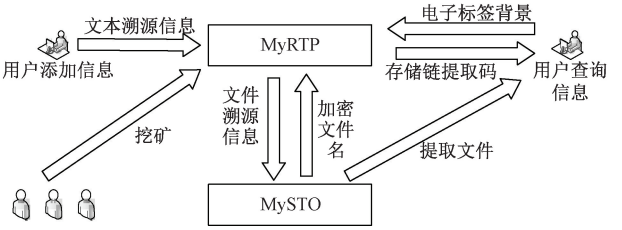

图 1　溯源系统流程示意图

Figure 1　Schematic diagram of traceability system

2.2　系统功能模型

系统功能模型如图 2 所示。其中,用户模块实现了用户注册与登录功能;交易模块实现了溯源数据上链和区块同步管理;挖矿模块包含 P 盘扫盘和爆块验证,实现了溯源数据的交易信息确认,并打包到区块链上产生虚拟币;存储模块实现了交易信息上链和溯源信息 IPFS 存储及下载;P2P 模块实现了

区块链中的节点同步和管理;查询模块实现了产品溯源和交易信息查询的功能。

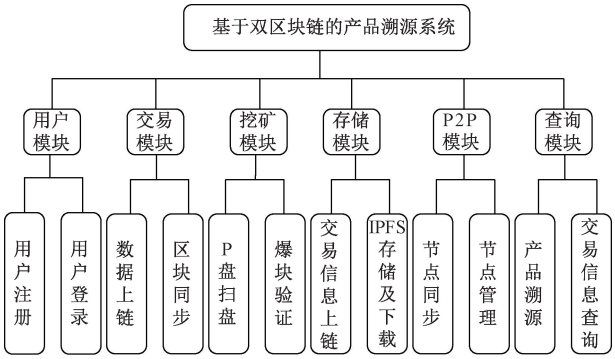

图 2 溯源系统功能模型

Figure 2 Functional model of traceability system

2.3 改进型 PoC 共识机制实现

2.3.1 PoC 共识机制实现

PoC 共识机制的实现主要分为生成 Plot 文件和区块锻造两个过程。

(1)生成 Plot 文件。Plot 文件是由一系列哈希运算结果排列组合而成的数据阵列。构成该数据阵列的基本单元被称为元胞(Nonce Cell)。Nonce Cell 的生成需要使用矿工 8 Byte 的数字账户 ID 地址拼接一个 8 Byte 大小的随机数种子(Nonce Number),构成一个大小为 16 Byte 初始种子(Initial Seed),之后对初始种子进行 1 次 Shabal256()计算,便可得到第 1 个哈希结果#8 191,如图 3 所示。

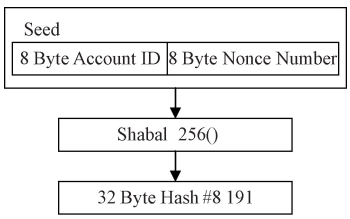

图 3 第一个哈希结果#8 191

Figure 3 The first Hash result #8 191

将第 1 个 Hash 结果#8 191 添加到初始种子之前可得到新的种子:#8 191+*Initial Seed*。随后进行 Shabal256()运算,得到第二个哈希结果#8 190;按照此种方式一直计算下去,最终可得到最后的哈希结果即 *Final Hash*。

如果在生成新种子的过程当中,种子长度超过了 4 096 Byte,此时仅保留种子长度的后 4 096 Byte。之后,再将利用该种子生成的 *Final Hash* 与先前计算得到的 8 192 个哈希结果做异或运算。

将异或运算后得到的 8 192 个哈希值按照顺序,每两个合并为一个 Scoop 并填入到 Nonce Cell 中,最终将得到一个包含 4 096 个 Scoop 的 256 KB 的 Nonce Cell,如图 4 所示。通过不断生成 Nonce Cell,并对这些 Nonce Cell 做优化后进行排列,最后写入 Plot 文件,直至写满该文件。

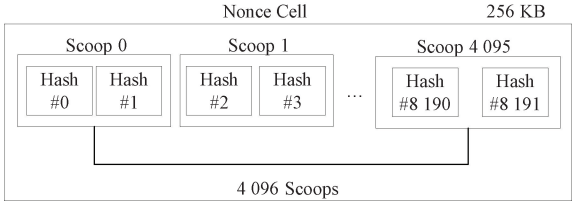

图 4 生成 Nonce Cell

Figure 4 Generating the Nonce Cell

(2)区块锻造。矿工在生成好 Plot 文件后,即可获取区块的打包和记账权,以完成溯源交易信息的确认。用户在提交溯源信息以后,矿工将在广播中接收到交易信息,并将该交易信息打包生成一个待出块的区块。矿工在进行挖矿获取记账权时,首先要通过钱包获取挖矿的基本信息,包括签名(Generation Signature)、目标(Base Target)和下一区块高度(Height)。矿工将签名和区块高度进行组合作为种子,并做 Shabal256()运算,得到生成哈希。然后对该哈希值做取模运算,取模数字是每个 Nonce 中 Scoop 的数量,也就是 4 096。取模运算以后将得到一个 4 096 以内的数字,该数字就是 *Scoop Number*。

挖矿的过程即检索并取出 Plot 文件中所有对应 *Scoop Number* 中的 Scoop 数据,并将该数据(64 位)和 Generation Signature(64 位)进行组合作为一个新的种子,然后对其进行 Shabal256()运算,从而得到 *Target*;之后将 *Target* 除以代表系统难度的参数值 *BaseTarget*,可得到区块倒计时(*Deadline*,8 位),如图 5 所示。

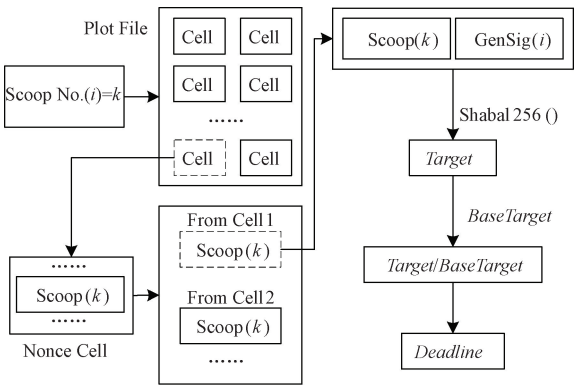

图 5 Deadline 生成

Figure 5 Deadline generation

Deadline 表示自上个区块出块以后,产生新的区块需要等待的时间。在该时间未结束前,新产生的区块是不合法的,并不会被区块链网络所接受。之后,所有矿工将自己找到的 *Deadline* 广播至区块

链网络,进而找到最小的 *Deadline*,此过程即为扫盘过程。当上一区块距离当前经过的时间已经达到或者超过了自己找到的 *Deadline*,并且没有其他矿工提交过更小的 *Deadline*,则该矿工将获得当前区块的打包权和记账权。

2.3.2　PoC 共识机制改进

在生成 Plot 文件时,通过不断生成 Nonce Cell,并对其优化排序,再写入 Plot,最终写满整个文件。由于 *Nonce* 值生成后可能存在 P 盘文件雷同的问题,因此本文在 PoC 共识机制的基础上进行了改进。

在生成 Plot 文件的过程中,*Nonce* 值生成后,通过将 Nonce Cell 做置换处理来避免 P 盘文件雷同的问题。具体为将先前的 8 192 个散列值与最终散列值逐一异或,并保存 8 192 个异或散列值,如图 6 所示。

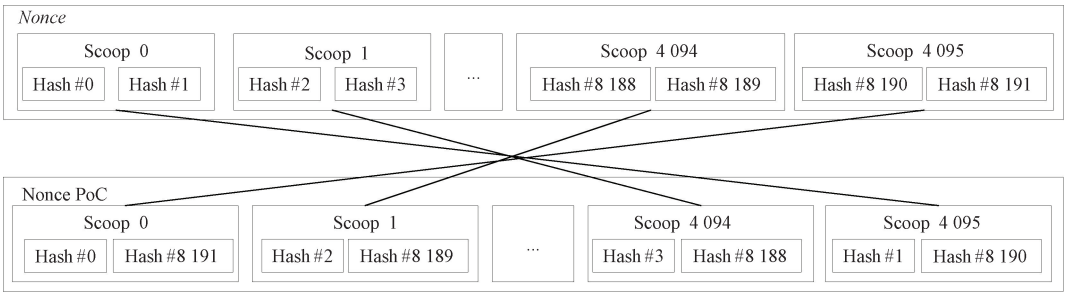

图 6　*Nonce* 值镜像互换

Figure 6　Mirror swap of *Nonce* values

在 *Nonce* 值镜像互换后,以字节为单位,字节位 7 和字节位 0 进行交换、字节位 5 和字节位 2 进行交换,以此类推按位交换操作后,得到最终的 Plot 文件。

3　系统关键部分实现

3.1　系统开发环境

系统基于 Windows 环境开发,后端利用 C#进行开发,前端基于 Java 和 Html 语言进行编写,系统编程软件采用 Visual Studio 2017。此外,利用 H2 数据库搭建了本地数据库。

3.2　MyRTP 查询链实现

3.2.1　产品信息注册与登录

系统中产品信息的注册是通过电子标签的识别来实现。读卡器读取串口后,自动读取电子标签的卡号和原始数据,识别成功后,读卡器会经过 Shabal256( )加密转换为一个 32 位字符长度的哈希值,该 Hash 值经过 SM3( )加密后作为溯源系统的脑密码登录溯源系统。脑密码是每个产品登录溯源系统的唯一登录密码,登录系统后,可自动生成唯一的账户 ID 和数字账户 ID(即产品 ID)以及公钥。

(1)UHF 卡识别。超高频(ultra high frequency,UHF)卡识别是射频识别(radio frequency identification,RFID)电子标签中的一种,具有可识别距离长、无源、防冲突性好的优点。UHF 读卡器接通电源后,通过调用 btnConn_Click( )事件,使用 GetInstance( )方法读取读卡器的串口号和波特率,完成与读卡器的串口通信。

在实现串口通信后,读卡器使用帧定义完成电子标签协议中 Inventory( )操作来获取 UHF 卡中的数据信息。获取 UHF 卡数据的关键代码如代码 1 所示。

代码 1 获取 UHF 卡数据的关键代码

```
private void btn_invt2_Click(object sender, EventArgs e)
{
LoopNum_cnt = LoopNum_cnt + 1;
txtSend. Text = Commands. BuildReadSingleFrame( );
Sp. GetInstance( ). Send(txtSend. Text);
}
```

(2)数据置换。读卡器在获取 UHF 卡信息后,将对初始数据进行置换。通过调用 UTF8. GetBytes( )函数来避免因编码方式所造成的字符数据传递失败。利用“EPC. password + MyRTP”对内容进行转换编码,以便于向 hashData 完整传递数据信息。hashData 接收完数据信息之后,利用 sha. ComputeHash( )对数据进行 50 轮次 Shabal256( )哈希运算,待运算结束后,32 字节的哈希内容将通过扩展的方式被填充至字节当中,扩展方式如代码 2 所示。

代码 2 数据置换扩展的关键代码

```
for (int i = 0; i<50; i++)
{
byte[ ] longHashData = hashData;
for (int j = 0; j<i/hashData. Length; j++){
longHashData = sha. ComputeHash(longHashData);}
int index = longHashData[i% longHashData. Length];
result. Append(hashAlphabet[index%hashAlphabet. Length]);
}
```

随后将扩展后的 32 位字节作为初始信息进而调用全节点钱包,钱包入口为 http://127.0.0.1:5125,并使用 SM3($('#login_password'). val( ))运

算进行加密置换从而得到钱包登录密钥。

作为产品登录 MyRTP 链的唯一方式,系统对产品信息进行确认后统一发放虚拟币,并将对应产品进行绑定。

(3)账户 ID 生成。MyRTP 链的产品 ID 默认是钱包登录密钥派生的 32 位标识符,而账户 ID 根据钱包登录密钥采用里所(Reed-Solomon,RS)编码方式转换生成。之所以采用 RS 编码方式,是因为此类编码方式将使地址信息很容易被识别为属于 MyRTP。MyRTP 使用 4 个“校验位”来区分不同的地址信息,并且随机地址发生冲突的机会是 $1/10^{-6}$(20 位冗余度)。同时,最多可纠正 1 个地址信息中的 2 个错字,最多可检测到 1 个地址信息中的 4 个错字。

RS 编码通过引入冗余来提高可靠性,该冗余能够在用户输入 MyRTP 账号时检测并纠正错误,其地址格式为 MYRTP-××××-××××-××××-××××××,其中,×代表数字或字母字符。为了避免出现地址混淆系统未使用字母 O 和 I、数字 1 和 0。系统采用统一格式,地址始终以“MYRTP”为前缀,所有地址都使用大写字母显示,并且使用连字符将地址分为 4、4、4、5 长度的字符组。在地址输入期间系统不强制执行,同时识别并支持数字地址,以实现向后兼容。

(4)公私钥获取。登录系统后,系统通过调用 Curve25519.clamp()方法将 Shabal256()加密后的 32 位字符输入。然后对字符串第 1 位与最后 1 位进行逻辑运算,进而得到 32 位字符的账户私钥。

通过调用 Curve25519.keygen()方法将 Shabal256()加密后的 32 位字符输入。经 Curve25519.clamp()方法运算获得私钥后,然后利用 Core()方法将私钥转换为 32 字节公钥数据。

3.2.2 溯源信息上传

上传溯源信息时,用户需要登录区块链钱包,然后才能进行溯源信息的上传。同时系统将生成的哈希值作为提取码,用于存储链上文件的下载。区块链在接收到交易信息后,会将交易信息写入交易池,等到矿工挖矿成功后将区块数据写入区块链并进行广播。

(1)登录钱包账户。用户通过轮询方式获取信息后,通过 Shabal256()和 SM3()加密置换,得到密钥登录钱包。

(2)信息上传。当用户使用账户 ID 登录区块链并上传溯源信息时,区块链钱包将利用 Web 服务器打开 dgs.html 网页,将地址信息、用户描述、文件信息、交易费用等信息录入,然后通过 API 接口调用 DGSListing.java 进行数据的传输。

DGSListing.java 利用自身的 DGSListin()将数据以("name","description","tags","quantity","priceNQT")格式传递到 APITag.CREATE_TRANSACTION 中,并将该交易信息打包发送到未确认交易池。交易池确认后,将信息打包后发送至区块链网络,从而完成数据记录的上链。溯源信息上传的关键代码如代码 3 所示。

代码 3 溯源信息上传的关键代码

```
private DGSListing()
{
super(new APITag[ ]{APITag.DGS, APITag.CREATE_
TRANSACTION},"name", "description", "tags", "quan-
tity", "priceNQT");}
String name = Convert.emptyToNull(req.getParameter("
name"));
String description = Convert.nullToEmpty(req.getParameter
("description"));
String tags = Convert.nullToEmpty(req.getParameter("
tags"));
long priceNQT=ParameterParser.getPriceNQT(req);
int quantity=ParameterParser.getGoodsQuantity(req);
}
```

3.2.3 用户信息管理

用户进行信息传输时,通过 Web 端钱包接口进入。用户信息管理模块主要实现了查看账户 ID、用户公钥、溯源信息查询等功能。

(1)用户信息管理模块连接。系统利用 Jetty 实现一个动态的内容服务器,完成区块链与 Web 页面的端口交互。Jetty 是一个轻量级的开源 HTTP 服务器,具备可扩展、易嵌入、高效的优点。系统通过导入 Eclipse 的 Jetty 代码库来实现端口的交互。

(2)数据库调用。MyRTP 主要应用的是 H2 数据库。H2 数据库中主要涉及的信息包括账户 ID、区块 ID、交易费用、时间戳等信息。

3.2.4 交易模块实现

系统生成的溯源区块信息记录在区块链上由矿工完成。矿工在完成初始化准备后,从钱包获取挖块信息,通过生成签名并用 Shabal256()算法获取新哈希值,运算后争夺最新区块的记账权,货币作为奖励分发。

(1)P 盘实现。首先,在硬盘中确定空余的空间,之后使用 P 盘软件,按照{engraver.exe,*d*,*-n*,8 192,*-id*,18 161 244 656 848 818 323,*-sn*,10 000}的

格式开始 P 盘,写入 *Nonce* 值。其中 *n* 表示 *Nonce* 的数量,为 8 192,*id* 为矿工账号 ID,*sn* 为 *Nonce* 的开始数值。

(2)区块数据信息同步。在溯源信息打包发送至未确认交易池后,区块链将未确认交易通过 P2P 网络进行广播,并验证交易的金额加上手续费是否小于等于该账户的可用余额。若满足该条件,系统将创建 UnconfirmedTransactions( )对象,并对该对象进行校验和签名,然后添加到“未确定交易”队列,并广播“未确定交易”到 P2P 网络,如图 7 所示。

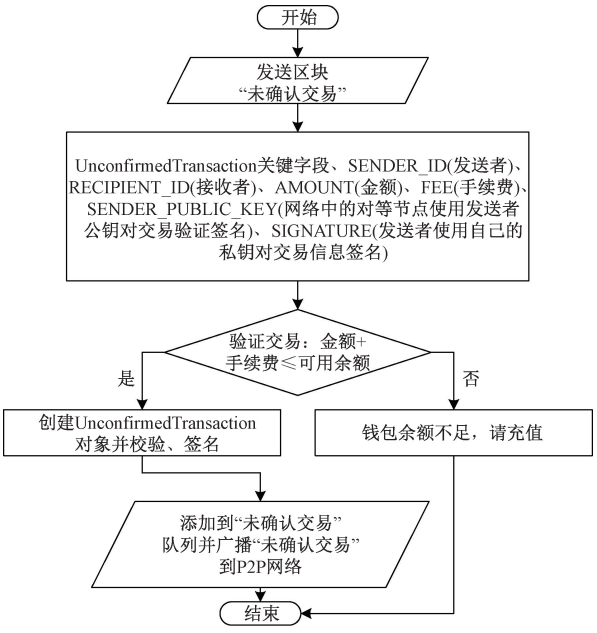

图 7　P2P 广播节点信息

Figure 7　P2P broadcast node information

(3)区块合法性检验。区块合法性校验是在矿工计算出最小的 *Deadline* 值后,将其 Nonce ID、账户 ID 参数和区块信息发送给钱包节点。钱包通过调用 Mining Plot( )方法计算出 *Nonce* 值,进而广播到全网,全网节点通过接收到的 *Nonce* 值来验证矿工的 *Deadline*。

3.3　MYSTO 存储链实现

3.3.1　IPFS 环境搭建

系统 IPFS 环境基于 go 语言实现。利用私有证书,搭建了 IPFS 私有集群,所有全节点运行 IPFS 节点,数据和区块链全节点保持同步。若有新添加到 1 个节点的数据,会同步广播到其他私有节点上。

3.3.2　溯源信息存储

在完成 IPFS 的搭建后,信息的存储通过调用如表 1 所示模块进行实现。ipfs daemon 模块初始化以后,才能进行信息的添加、下载等操作。通过 ipfs add 模块将文件进行分片存储,并调用 dag 层缓存到内存,进而存储至本地。然后再利用 dag 调用交换层,进而广播宣告 provide 信息,将其存储至分布式散列表(distributed Hash table,DHT)中。通过 ipfs get 模块提取 IPFS 中的文件。

提取文件通过调用 dag 层来实现,若本地存在该内容则将文件另存到目标位置;若本地不存在该内容,则由 dag 服务调用交换、路由、网络获取对应内容。

在对 IPFS 初始化以后,可将大批量文件上传至 IPFS。同时利用区块链同步功能将 IPFS 与 MySTO 相结合,使 IPFS 成为区块链存储的一部分。

表 1　IPFS 关键模块

Table 1　IPFS key modules

模块	位置	描述
ipfs daemon	go-ipfs/cmd/ipfs/daemon. go	解析传参、初始化配置,启动网络侦听服务
ipfs add	go-ipfs/core/commands/add. go	将文件分片存储,调用 dag 层缓存到内存,进行本地存储
ipfs get	go-ipfs/core/commands/get. go	提取 IPFS 中的文件

利用 IPFS 共享密钥的方式,通过区块链直接调用 IPFS 模块,使 IPFS 成为区块链的一部分,不仅解决了大文件存储问题,而且能够使 IPFS 的每个节点与区块链全节点都完成数据的同步更新。

3.4　双区块链联通实现

双链存储的方式借鉴链式结构的优点,通过区块链交易无序的特征来构造链式交易结构,不仅避免了传统数据库中心化的问题,而且有效弥补了单链节点存储空间有限的缺点。

MyRTP 链与 MySTO 链之间通过 Web brige 进行交互调用,从而实现查询链和存储链的结合,两链均采用 PoC 共识机制。系统功能的具体实现可分为溯源数据查询和数据文件存储两大部分,分别对应系统中的 MyRTP 与 MySTO 区块链。需要进行溯源信息上传时,首先在 MyRTP 进行基本记录,MyRTP 钱包地址为 http://127.0.0.1:5125。若产品存在文件、图片、视频等大文件需要进行存储,则可以文件的形式上传至 MySTO,其流程如图 8 所示。MySTO 钱包地址为 http://127.0.0.1:9125,两条链的扫盘路径通过 plot_dirs( )设置。

从 MyRTP 上传到 MySTO 是通过调用 jQuery 库的 post( )方法实现的。然后利用 post( )方法中的 Http post 请求即可从服务器加载数据。MyRTP 上传文件到 MySTO 时,通过 Web bridge 发送 post 命

令,构造 post 的报文头。

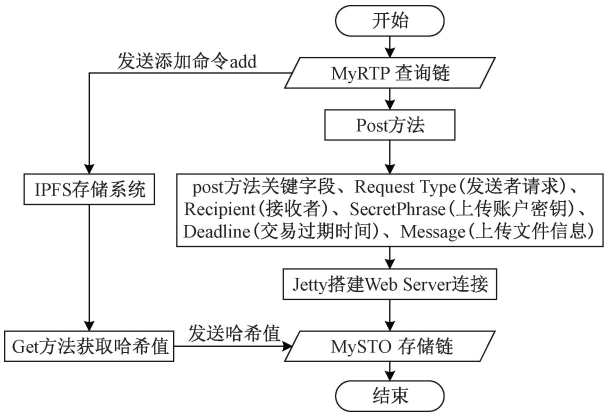

图 8 MyRTP 上传信息至 MySTO

Figure 8 MyRTP uploads information to MySTO

MySTO 收到交易请求后,通过调用 API 接口,在核实区块合法性信息后扣除交易手续费,将其广播至区块链网络中。

从 MyRTP 到 MySTO,溯源数据通过 post 方式完成两链联通,并通过 createTransaction( )将信息上链存储。已存储文件通过 add 命令添加至 IPFS 系统,并生成唯一哈希值记录在 MySTO 区块链上。用户可凭借此哈希值完成溯源文件的提取。

4 系统测试与分析

4.1 系统功能测试与分析

4.1.1 信息登录功能测试

系统登录模块主要调用了 UHF 电子标签读取、数据解析、串口识别、数据转换、区块链钱包接入等方法。启动射频识别读卡器,刷卡,使用射频识别电子标签,登录查询链 MyRTP,如图 9 所示。

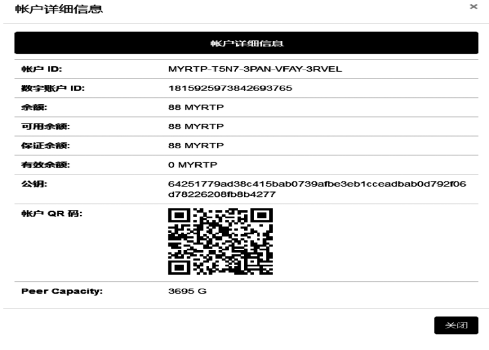

图 9 账户详细信息

Figure 9 Account details

溯源信息打包上链后,矿工为竞争记账权进行爆块,打包区块信息到区块链并进行广播。区块链其他节点确认交易后,用户即可在区块链上查询到自己上传的区块信息。

4.1.2 溯源信息查询功能测试

系统用户通过注册后的 ID 信息作为登录溯源系统的脑密码,通过 Web 接口登录到系统的查询界面,可看到区块链上的所有交易信息,如图 10 所示。

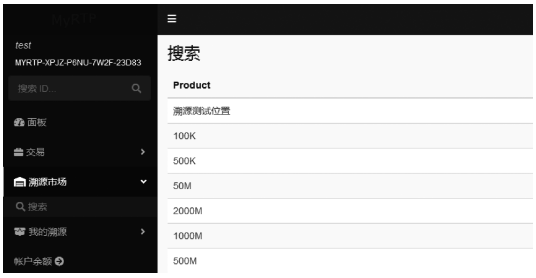

图 10 溯源信息查询

Figure 10 Traceability information query

若用户对指定产品进行查询,则可通过输入产品对应的唯一账户 ID,即可查到该产品在生产和运输过程中的溯源信息;对于存放在 MySTO 链的具体溯源信息,可通过哈希值从 MySTO 链提取。

4.1.3 溯源信息存储功能测试

对于大批量的溯源信息,在登录系统并获得公私钥的基础上,可选择本地文件进行上传,上传成功后,返回文件的特征哈希值作为文件的存储凭证。当需要下载时,客户可凭此凭证进行文件的提取,如图 11 所示。

图 11 文件提取

Figure 11 File extraction

4.2 系统性能测试与分析

4.2.1 溯源查询性能测试

对系统溯源查询性能进行测试。在区块链高度为 8 335 情况下,根据溯源标签进行产品溯源查询,http 响应时间为 7 ms,如图 12 所示。

图 12 查询性能测试

Figure 12 Query performance test

4.2.2　溯源存储性能测试

(1)系统上传与下载测试。系统经过上传测试,上传平均速度 80.66 M/s;系统经过下载测试,下载平均速度 90.75 M/s,具有良好的上传和下载性能,具体测试数据如表 2 所示。

表 2　系统上传、下载性能测试表

Table 2　System upload and download performance test table

文件大小/Byte	上传时间/ms	下载时间/ms	上传速度/($M \cdot s^{-1}$)	下载速度/($M \cdot s^{-1}$)
658 954 334	8 704	7 421	75.7	88.8
330 998 852	4 195	3 689	78.9	89.7
213 156 752	2 588	2 319	82.4	91.9
145 489 234	1 698	1 572	85.7	92.6

(2)系统交易测试。随着区块链的不断发展,部分主流区块链项目因网络阻塞问题导致无法在高并发业务领域实施。利用 Hyperledger Caliper 工具对 MyRTP 链和部分主流区块链项目进行测试分析,表 3 为交易测试详情。

表 3　MyRTP 与其他虚拟货币的交易测试

Table 3　Transaction test between MyRTP and other virtual currencies

区块链项目	单个区块大小/B	每笔交易大小/B	区块总交易数	平均出块时间/s	*TPS*/(笔·s^{-1})
BTC	1 048 576	250	4 194.304	600	6.99
BCH	8 388 608	586	14 315.031	600	23.86
ETH	21 345	93	229.516	13	17.66
LTC	1 048 576	1 434	731.225	13	56.25
MyRTP	1 329 328	176	7 553	60	125.88

TPS 指系统每秒能够处理的事务数,是衡量区块链系统性能的重要指标之一。由表 3 可以看出 BTC 每秒仅处理 6.99 笔交易,BCH 每秒处理 23.86 笔交易,ETH 每秒处理 17.66 笔交易,LTC 每秒处理 56.25 笔交易。可以看出,BCH、ETH 作为继 BTC 之后发展起来的区块链项目,其处理交易的能力有所提升。而 LTC 作为最近两年兴起的新项目,在处理交易的效率上更是有了非常大的进步。本文设计实现的 MyRTP 链每秒可处理 125.88 笔交易,每秒所完成的交易量远胜于其他区块链主流项目,因此 MyRTP 链的工作效率也要远高于其他区块链项目。

4.2.3　恶意节点识别测试

当系统中存在未授权的恶意节点登录时,MyRTP 链通过账户余额权限设置,使其无法上传溯源信息。为保护 MySTO 中存储的溯源文件,系统以扣除余额的方式保证下载文件的统一管理。余额不足的用户,即使拿到提取哈希值也无法进行文件的下载,如图 13 所示。

图 13　MySTO 无权限下载测试

Figure 13　MySTO download tests without permission

4.3　系统方案对比分析

本节从共识机制、查询性能、存储容量、去中心化以及成本 5 个维度将本方案与其他溯源方案进行对比分析,如表 4 所示。

表 4　溯源方案对比

Table 4　Comparison of traceability schemes

方案来源	共识机制	查询效率	存储容量	去中心化	成本
传统溯源	无	慢	大	否	高
文献[5]	PoET	快	小	是	高
文献[9]	PoS	慢	小	半中心化	低
文献[10]	PoW	慢	小	是	高
文献[17]	PoW+PBFT	快	大	半中心化	高
本文	PoC	快	大	是	低

从表 4 可以看出,文献[5]中的 PoET 共识依靠专用硬件来保障安全性;文献[9]和文献[17]并没有做到完全的去中心化,在溯源应用中无法确保数据可信;文献[10]依赖 PoW 共识在生成区块信息的同时也造成了大量的无用计算。本方案通过双链模式在实现去中心化溯源的基础上,利用 IPFS 提升系统可靠存储和分享能力,同时改进 PoC 共识算法将 PoW 高能源消耗替代为低能耗的硬盘查找。不仅保证底层的安全性,而且在减少能源消耗的同时也能满足区块链溯源系统的应用需求。

5　结论

利用双区块链和 IPFS 技术解决了传统溯源方案和现有单区块链溯源方案所存在的数据造假、数据泄露、大批量数据存储等问题。利用改进后的 PoC 共识算法解决了现有单区块链和双区块链溯源方案工作效率不高的问题。经测试表明,系统在较为理想的时延范围内实现了溯源信息的上传、存储和下载等功能,能够根据区块链记录进行溯源信息过程查询,可有效防止数据篡改和恶意攻击,保证溯源信息结果完整可信。

参考文献:

[1] 叶云. 农产品质量追溯系统优化技术研究[D]. 广州:华南农业大学, 2017.
YE Y. Research on technology for optimizing agricultural product quality traceability system [D]. Guangzhou: South China Agricultural University, 2017.

[2] 李明佳, 汪登, 曾小珊, 等. 基于区块链的食品安全溯源体系设计[J]. 食品科学, 2019, 40(3):279-285.
LI M J, WANG D, ZENG X S, et al. Food safety tracing technology based on block chain [J]. Food Science, 2019, 40(3): 279-285.

[3] 李永强, 刘兆伟. 基于区块链的车联网安全信息共享机制设计[J]. 郑州大学学报(工学版), 2022, 43(1): 103-110.
LI Y Q, LIU Z W. Blockchain-based secure data sharing mechanism design in the vehicular networks[J]. Journal of Zhengzhou University (Engineering Science), 2022, 43(1): 103-110.

[4] LI J, WANG X Y. Research on the application of blockchain in the traceability system of agricultural products [C]//2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). Piscataway: IEEE, 2018: 2637-2640.

[5] BARALLA G, PINNA A, CORRIAS G. Ensure traceability in European food supply chain by using a blockchain system [C]//2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB). Piscataway: IEEE, 2019: 40-47.

[6] 张国英, 毛燕琴. 一种基于区块链的去中心化数据溯源方法[J]. 南京邮电大学学报(自然科学版), 2019, 39(2): 91-98.
ZHANG G Y, MAO Y Q. Blockchain-based decentralized data provenance method[J]. Journal of Nanjing University of Posts and Telecommunications (Natural Science), 2019, 39(2): 91-98.

[7] 刘雅东. 基于区块链的溯源信息存储平台的研究与实现[D]. 北京:北京邮电大学, 2019.
LIU Y D. Research and implementation of trace source information storage platform based on blockchain [D]. Beijing: Beijing University of Posts and Telecommunications, 2019.

[8] 高琰晨. 基于区块链技术的物流信息追溯机制研究[D]. 杭州:浙江工业大学, 2019.
GAO Y C. Research on traceability mechanism of logistics information based on blockchain technology [D]. Hangzhou: Zhejiang University of Technology, 2019.

[9] LENG K J, BI Y, JING L B, et al. Research on agricultural supply chain system with double chain architecture based on blockchain technology [J]. Future Generation Computer Systems, 2018, 86: 641-649.

[10] 刘家稷, 杨挺, 汪文勇. 使用双区块链的防伪溯源系统[J]. 信息安全学报, 2018, 3(3): 17-29.
LIU J J, YANG T, WANG W Y. Traceability system using public and private blockchain[J]. Journal of Cyber Security, 2018, 3(3): 17-29.

[11] 高文涛, 张桂芸. 基于联盟区块链和IPFS的音乐共享模型[J]. 天津师范大学学报(自然科学版), 2020, 40(2): 68-74.
GAO W T, ZHANG G Y. Music sharing model based on consortium blockchain and IPFS[J]. Journal of Tianjin Normal University (Natural Science Edition), 2020, 40(2):68-74.

[12] 冯国富, 胡俊辉, 陈明. 基于区块链的水产品交易溯源系统研究与实现[J]. 渔业现代化, 2022, 49(1): 44-51.
FENG G F, HU J H, CHEN M. Research and implementation of aquatic product transaction traceability system based on blockchain[J]. Fishery Modernization, 2022, 49(1): 44-51.

[13] 曾卫民. 基于区块链的可溯源IPFS系统研究与实现[D]. 扬州:扬州大学, 2022.
ZENG W M. Research and implementation of traceable IPFS system based on blockchain[D]. Yangzhou: Yangzhou University, 2022.

[14] 代小龙. 基于区块链的分布式数据存取应用方案研究[D]. 重庆:重庆邮电大学, 2020.
DAI X L. Research on distributed data access application scheme based on blockchain [D]. Chongqing: Chongqing University of Posts and Telecommunications, 2020.

[15] 刘懿中, 刘建伟, 张宗洋, 等. 区块链共识机制研究综述[J]. 密码学报, 2019, 6(4): 395-432.
LIU Y Z, LIU J W, ZHANG Z Y, et al. Overview on blockchain consensus mechanisms[J]. Journal of Cryptologic Research, 2019, 6(4): 395-432.

[16] KUMAR R, TRIPATHI R. Implementation of distributed file storage and access framework using IPFS and blockchain[C]//2019 Fifth International Conference on Image Information Processing (ICIIP). Piscataway: IEEE, 2020: 246-251.

[17] SUN W R, ZHU X H, ZHOU T, et al. Application of blockchain and RFID in anti-counterfeiting traceability of liquor[C]//2019 IEEE 5th International Conference on Computer and Communications (ICCC). Piscataway: IEEE, 2020: 1248-1251.

Research and Implementation of Product Traceability System Based on Dual Blockchain

HAN Yanyan[1,2], WEI Wanqi[1], DOU Kaili[3], ZHANG Qi[1]

(1. Department of Electronics and Communication Engineering, Beijing Electronic Science and Technology Institute, Beijing 100070, China; 2. College of Information Engineering, Xidian University, Xi'an 710071, China; 3. Department of Cyberspace Security, Beijing Electronic Science and Technology Institute, Beijing 100070, China)

Abstract: Due to the limited block information storage capacity, the existing blockchain traceability system mostly adopted the combination of blockchain and cloud storage, which could not fundamentally solve the problem of blockchain traceability information storage and data leakage. The dual blockchain mode was constructed, the query chain was used to complete the uploading of traceability information and to realize the basic functions of the traceability system. The storage chain was combined with the interplanetary file system to ensure data integrity and security. On this basis, the improved capacity proof consensus algorithm was used to ensure the underlying security and meet the application requirements of blockchain traceability system while reducing energy consumption. The test showed that the average upload speed of the system could reach 80.66 M/s, and the average download speed could reach 90.75 M/s, with good upload and download performance. Blockchain transactions per second reached 125.88, and the carrying capacity of a single transaction could meet the demand for block transactions of the traceability system.

Keywords: dual blockchain; capacity proof; interplanetary file system; traceability system; reliable data storage

(上接第 33 页)

Sub-frame Crack Rig Testing and Simulation Analysis Based on Full-vehicle Rough Road Spectrum

PAN Gongyu[1], XU Rui[1,2], YANG Xiaofeng[1]

(1. School of Automobile and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China; 2. Evergrande New Energy Automotive R&D Institute Global Headquarters, Shanghai 201620, China)

Abstract: The study was conducted to examine the durability issue occurred in front stabilizer bar bracket connected to sub-frame in full vehicle testing. Firstly, stabilizer bar system was plastered with strain gauges and calibrated, and drop link force and stabilizer bar twist displacement acquired on proving ground, sub-frame with stabilizer bar system physical test rig was designed and built, and rig tests in accordance with durability specifications were conducted. The test results showed that built physical test rig could greatly reappear crack location in full vehicle testing, the fatigue life of the physical bench had a deviation of 2.5% compared to full vehicle testing. Based on this, a stabilizer bar and sub-frame multi-body virtual model was built with the same constraint boundary and the same loading method of the physical test rig. Then CAE fatigue simulation was used through quasi-static finite element fatigue life analysis method to reappear related area risk. The simulation results showed that phase and amplitude had a good coincidence in time domain, the PSD spectrum also had a good accuracy in frequency domain, the relative damage was almost closed to 1 with the comparison between the simulation and test in droplink force and stabilizer bar relative displacement. The deviation between the simulated fatigue life at the relevant risk position and the test life of the full vehicle was 6.25%. A higher accuracy risk position load was obtained, and the reappearance of durability risk position was achieved. Finally, based on simulation fatigue load, the optimization risk structure was evaluated. Optimized proposal eventually passed the test rig and full vehicle testing successfully.

Keywords: rig test; durability; virtual iteration; fatigue simulation; correlation