第47卷 第2期 计 算 机 工 程 2021年2月
Vol.47 No.2 Computer Engineering February 2021

·网络空间安全· 文章编号：1000-3428(2021)02-0176-06 文献标志码：A 中图分类号：TP309

基于区块链的链下个人数据保护方案

纪露生[1,2]，张桂玲[1,2]，杨佳润[2]

（1.天津工业大学 计算机科学与技术学院，天津 300387；2.天津市自主智能技术与系统重点实验室，天津 300387）

摘 要：现有结合区块链保护个人数据的方案在授权第三方服务时多将用户的个人数据地址分享给第三方服务，在用户撤销对第三方服务的访问权限后，第三方服务仍然拥有个人数据地址。为避免用户数据泄露，通过采用链下存储的方式，提出一种基于区块链的匿名地址管理方案。利用资源服务处理个人数据的加密地址，并限制第三方服务只能获得用户个人数据地址的加密密文，使用户在修改对指定第三方服务的访问权限后，能够通过更改智能合约的访问策略实现细粒度访问控制。在此基础上，利用以太坊平台设计个人数据管理系统，使用Solidity语言编写智能合约，从而实现对个人数据的保护。该方案具有通用性，可由不同的区块链平台实现，合约部署后的调用结果以及对合约进行50次和500次的性能测试结果验证了其有效性和安全性。

关键词：区块链；个人数据；第三方服务；链下存储；加密地址

开放科学(资源服务)标志码(OSID)：

中文引用格式：纪露生，张桂玲，杨佳润.基于区块链的链下个人数据保护方案[J].计算机工程，2021，47(2)：176-181，187.

英文引用格式：JI Lusheng，ZHANG Guiling，YANG Jiarun. Off-chain personal data protection scheme based on blockchain[J].Computer Engineering，2021，47(2)：176-181，187.

Off-Chain Personal Data Protection Scheme Based on Blockchain

JI Lusheng[1,2]，ZHANG Guiling[1,2]，YANG Jiarun[2]

（1.School of Computer Science and Technology，Tiangong University，Tianjin 300387，China；
2.Tianjin Key Laboratory of Autonomous Intelligent Technology and System，Tianjin 300387，China）

【Abstract】Most personal data protection schemes combined with block chain share the address of user's personal data to the Third Party(TP) service when authorizing them.Even if the user revokes the access right to the TP service，they still have the address of personal data.To avoid user data leakage，this paper proposes an anonymous address management scheme based on blockchain by using off-chain storage.The Resource Service(RS) is used to process the encrypted address of personal data，and the TP service is restricted to obtain the encrypted ciphertext of the user's personal data address.After modifying the access rights of the designated TP service，users can achieve fine-grained access control by modifying the access policy of the smart contract. On this basis，the personal data management system is designed by using the Ethereum platform，and the smart contract is written in Solidity to realize the protection of personal data.The general scheme can be realized on different blockchain platforms.Its effectiveness and security are also demonstrated by the results of calling the deployed contract and the 50 and 500 times of performance tests.

【Key words】blockchain；personal data；Third Party(TP) service；off-chain storage；encrypted address

DOI：10. 19678/j. issn. 1000-3428. 0057024

0 概述

大数据、互联网和5G技术的发展，在为人类带来发展机遇的同时，也导致大量数据泄露事件的发生，造成严重后果。2018年1月，印度10亿公民身份数据库Aadhaar遭到网络攻击，该库除公民基本信息外还有指纹、虹膜等敏感信息。2018年8月，美国医疗收集局(AMCA)遭到黑客入侵，泄露了大量消费者的姓名、电话、余额、信用卡账户等重要信息。2019年1月，美国俄勒冈州公共服务部遭到了黑客的电子邮件钓鱼攻击，泄露了大量的客户社保信息和健康信息。这些重大泄露事件的报道，使人们对个人数据的安全更为担忧。

新兴区块链和智能合约技术[1]可为数据管理提供更好的技术支持，如在医疗数据领域，文献[2]利用区块链技术实现了个人医疗数据的存取和管理，

作者简介：纪露生(1996—)，男，硕士研究生，主研方向为数据安全、区块链技术；张桂玲(通信作者)，教授、博士；杨佳润，硕士研究生。

收稿日期：2019-12-25 修回日期：2020-03-03 E-mail：glzhang808@sohu.com

文献[3]解决了移动医疗数据的协同共享问题。此外，文献[4]基于区块链提出一种分布式个人数据和数字身份的联邦授权框架，文献[5]利用区块链实现了个人数据分布式隐私保护，文献[6-7]则从法律角度分析个人网络数据的所有权问题，为个人数据保护研究提供了新的解决方案。

在个人数据管理方面，研究者基于区块链和智能合约也提出了一些优秀方案。文献[8]提出一种将区块链作为一个不需要可信第三方机构的访问控制管理器，并由事务管理权限的机制。文献[9]为个人建立用户配置文件和数据/设备，同时为服务提供访问数据的接口，使用户可以在服务请求个人数据时设置服务可以访问的数据/设备及其访问级别，并授予或拒绝请求的权限。文献[10]通过设计一个个人元数据管理框架，提出SafeAnswers机制。该机制允许个人收集并存储自己的元数据，并允许服务询问根据元数据计算得出答案的问题，从而实现对这些元数据的细粒度访问。

本文采用链下存储的方式，提出一种基于区块链的匿名地址管理方案，使用开源区块链以太坊(Ethereum)并假设一个诚实可信的资源服务(Resource Service，RS)来共同管理用户数据。用户将个人数据的加密地址分享给区块链的智能合约，解密私钥则分享给RS。该方案通过智能合约建立用户和第三方服务(Third Party，TP)之间的授权关系，使第三方服务得到用户授权后才能获得用户个人数据的加密地址和访问令牌，从而实现细粒度访问控制，达到保护个人数据的目的。

1　相关理论与研究

1.1　区块链与智能合约

区块链源于比特币系统[11]，是一个点对点的分布式系统，其利用独特的共识算法[12]和激励机制鼓励系统成员维护系统，系统的每一个状态依靠成员共同决定，而不是依靠某个单一节点，因此，具有伪匿名性、去中心化、不可变性、透明度和安全性高等特点。比特币系统是目前全球最早也是规模最大的区块链系统，被称为区块链1.0。之后发展到带有智能合约[13]的区块链，如以太坊[14]和超级账本[15]，被称为区块链2.0。由于目前区块链的效率较低，因此研究者提出构建区块链3.0的设想，目标是实现更高性能和更高吞吐量。

以太坊是一个开源且具有智能合约功能的公共区块链平台，其通过图灵完备的以太坊虚拟机(Ethereum Virtual Machine，EVM)来处理点对点合约，运行在EVM上的程序称为智能合约。官方发布的智能合约语言是Solidity，合约部署后发布到链上，并被EVM执行其中的代码程序。智能合约与普通程序的区别在于，其程序发布到区块链上将被作为一个交易记录永久记录在块中，不可更改，这类似于纸质协议，合同双方签订协议后，一经发布则此合同将具有法律效力，不可更改。超级账本的智能合约支持多种高级语言，运行在Docker容器中。

1.2　相关研究

区块链因具有透明性、不可变性、可追溯性和防篡改等特性被应用于多个领域，如金融、供应链和数据资产管理等。在数据资产管理方面，基于区块链的个人数据保护方案大多采用链下存储[16]的方式，即链上只保存简短的访问地址或路径。这种以链下存储方式将访问数据的关键路径或关系存储在链上而将大规模数据存储在链下的方法，构建了一个完整的权限控制生态，使得区块链能够通过存储少量数据实现对大量数据的控制，极大地增强了区块链的可扩展性。

2018年，欧盟通过了《一般数据保护条例》(简称GDPR)，指出了个人数据处理原则，即合法性、公平性、透明性、保密性、完整性和有限控制等。

文献[17]将数据分为静态用户数据和动态使用服务生成的数据两类，同时从技术角度将区块链分为3层，即智能合约层(用于存储用户和第三方服务之间的合约)、接入层(利用区块链不可变性和完整性特点保护隐私)和哈希存储层(用于存储数据哈希)，旨在构建一个以人为中心且符合GDPR的个人数据和身份管理系统。其中，链下数据库存储的个人数据使用对称密钥加密，对称密钥分别被拥有数据的各方持有。该方法存在的主要问题在于：只在第一次授权给第三方服务时是安全的，因为第一次授权之后，第三方服务将拥有个人数据地址和对称密钥，通过可解密密文得到完整的个人数据，之后即使撤销第三方服务对个人数据的访问，依靠个人数据地址和对称密钥其依然可以得到数据；此外，第三方服务也很有可能将地址和对称密钥泄露出去，或者将明文泄露出去，造成严重的数据泄漏风险。

文献[18]设计了一个遵循GDPR的个人数据管理系统，其允许数据所有者强制执行数据使用许可，确保只有指定方可以处理个人数据，并使用智能合约和加密技术将所有数据活动记录在一个不可变的分布式账本中。该系统采用链下存储方式，将个人元数据的访问地址以加密的形式存储在链上，第三方服务得到用户授权之后将被授予访问令牌并获得用户元数据的加密地址，同时得到解密此地址的私钥。第三方服务使用用户的个人数据地址和访问令牌请求资源服务。由于个人数据地址一般情况下是不变的，因此即使用户撤销了对第三方服务的访问

权限,但第三方服务仍拥有用户个人数据地址,利用此地址其仍有可能获得个人数据,同时该地址也存在泄露风险。

2 基于区块链的个人数据保护架构

2.1 当前架构及存在问题

本文给出当前个人数据保护体系的综合性框架,如图1所示,用以分析当前基于区块链的个人数据保护方案存在的问题。该架构为一个通用架构,不同的方案采用不同的算法。架构中包含4个实体,分别为用户(User)、管理用户数据访问权限的数据控制者(Data Controler,DC)、想要访问用户数据的第三方服务(TP)和管理链下数据存储的资源服务(RS)。

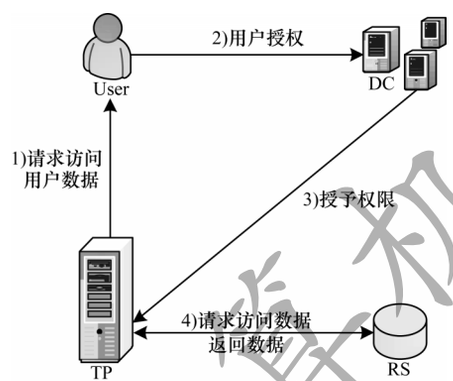

图1 个人数据保护体系的综合性框架

Fig.1 Comprehensive framework of personal data protection system

如图1所示,在现有系统架构中,一个TP想要访问User的数据,需要经过以下4步:

1)TP向User提出访问其个人数据的请求。

2)User确认授权,将请求发送给DC。

3)DC验证User和TP身份后,授权TP访问User的数据,并将授权记录在链上,发送User个人数据的地址或访问令牌给TP。

4)TP利用访问令牌或地址访问资源服务,资源服务验证后,处理TP请求并返回数据。

在第1步中,TP直接向User发送访问请求,User确认授权后提交自己的签名和TP的签名发送给DC验证,由DC处理授权,由此可减少DC的负载。在第2步中,若DC验证TP的身份不合法,则不会通过这一请求,导致授权失败。因此,TP需要经过DC验证和授权,在系统中拥有自己的身份。如果TP直接向User发送请求,那么这一步TP身份的合法性就无法进行验证,只能等到User确认授权后发送给SP时将TP和User的签名一同发送给DC,再由DC验证。因此,应先验证TP身份的合法性。

链下数据的存储大多使用IPFS系统[19],其将数据以文件的形式分散地存储在系统中,更新文件时重新分配地址。由于是分散存储,文件被散列在各处,因此只有通过创建文件时分配的地址才能找到分散在各处的文件。若使用传统数据库存储,则一般不能更改存储路径和关系。

在图1的第3步中,TP获得访问User数据的权限后,将会获得User的个人数据地址。在这种情况下,即使User撤销TP对其数据的访问权限,数据地址仍然被TP持有,TP仍然有可能继续访问到User的数据。

针对上述问题,本文提出一个解决方案,即利用RS处理TP得到授权后分配的加密地址,而不是直接把用户个人数据地址分配给TP。TP提交数据的加密地址和访问令牌给RS,由RS处理TP的请求并返回数据。

2.2 改进的系统架构设计

本文方案中改进的系统架构如图2所示。

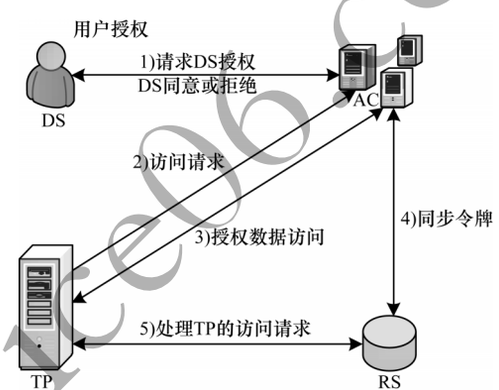

图2 改进的系统架构

Fig.2 Improved system architecture

从C/S模式的角度来分析,服务S帮助User管理其数据,用来处理TP的访问请求和User的授权,User即数据主体(Data Subject,DS)。此时,服务S可以看作是一个服务于DS和TP的权限控制者(Authority Controller,AC)。如果TP违反协议,它将会被永久地记录在账本中。本文假设RS是“honest but curious”,这意味着RS诚实地执行所需的协议,即使它可能对操作后收到的结果感到好奇。利用区块链的伪匿名性,RS被假设成一个只存储DS身份(以太坊地址)和解密其加密地址私钥的资源服务,DS匿名提交自己的数据给RS,得到真正的地址,其利用第三方加密软件对地址进行加密得到其数据地址的加密地址和解密私钥,原始地址由DS保存,加密地址被分享给AC,解密私钥被分享给RS。DS的身份和私钥以一个键值或表的形式存储而不关联DS的身份和地址。由此,RS没有配套的加密地址,

无法单独查询库中指定用户的数据，只有TP请求RS访问DS的数据时，RS根据TP提交的加密地址和访问令牌（访问令牌中包含DS的身份）得到DS的身份和加密地址，使用与DS身份关联的私钥解密或加密地址才能返回给TP数据。

如图2所示，在改进的系统架构中，一个TP想要访问DS的数据，需要经过以下5步：

1）TP向AC提出对DS数据访问的申请。

2）AC验证TP身份后向DS发送TP的访问请求，DS选择同意或拒绝。

3）AC验证DS的身份并处理授权，若DS同意授权，则AC发送给TP访问令牌和加密地址。

4）AC和RS同步访问令牌。

5）TP将访问令牌和加密地址作为参数提交给RS，RS验证令牌并使用解密算法解密地址，根据地址查找数据并返回给TP。

在改进的系统架构中，每个参与的实体都有一个以太坊地址，使用AC、DS、TP在以太坊中的地址id_AC、id_DS、id_TP来代表各个参与实体在系统中的身份。访问控制列表以细粒度控制方式记录哪些TP被授权以及被授予的操作和授权期限，其包含已经被授权的TP集合和每个TP的授权状态（当前被授权的操作和访问令牌），DS可随时查看当前已被授权的TP。访问控制列表的权限被限制为只有DS才能修改，通过更改访问控制列表中对指定TP的访问策略可修改或移除TP的权限，DS可以随时授权、修改和撤销TP的访问权限。

在初始阶段，DS同意AC管理其个人数据，并将链下地址pointer的加密密文enc_pointer分享给AC，将解密密钥sk_dec分享给RS。AC和DS建立一张可证明授权关系的访问控制列表A_list，其为address地址类型到A_list的映射，即每个地址都对应一个A_list列表，此表标识DS、AC和TP的关系。TP是一个address地址类型到address=>TP_policy的映射，标识唯一当前DS对应TP的访问策略。当有TP被授权时，TP加入A_list关系表，由于DS可能授权多个TP访问权限，因此使用一个集合tps存放已被授权的TP。根据A_list[DS].tps可查询当前授权的TP，通过A_list[DS].TP_policy[TP]可查询当前TP的授权状态。A_list列表和TP列表结构设计代码如下：

```
enum actions;/*授权操作(CRUD),为枚举类型*/
actions action;
address AC;
mapping(address=>list) public A_list;/*DS地址到其访问控制列表的映射,保证唯一性*/
struct list {/*DS列表结构*/
uint8 mflag;
string enc_pointer;/*加密指针*/
address AC;
address DS;
mapping (address=>mapping (address => TP_policy)) Tp;/*DS对指定TP的授权列表*/
mapping(address=> address[]) tps;/*DS已授权的TP集合*/
uint tpsflage;
}
struct TP_policy{/*TP被授予权限的列表*/
uint tpflag;
address tp;
address ds;
actions act;
bytes32 token;
}
```

在struct list中，mflag标识当前AC是否已经请求AC管理其个人数据，tpsflag标识当前已被授权的TP数量，tpsflage标识被授权TP的数量。在struct TP中，tpflag标识当前TP已被授权。

2.3　算法设计

图3为DS同意授权TP的组件交互时序图，具体过程如下：1）TP向AC发送访问DS数据的请求；2）AC询问DS是否同意授权；3）DS同意并向AC发送授权确认；4）AC收到DS的确认，更新A_list；5）AC发enc_pointer和访问令牌，授权成功。

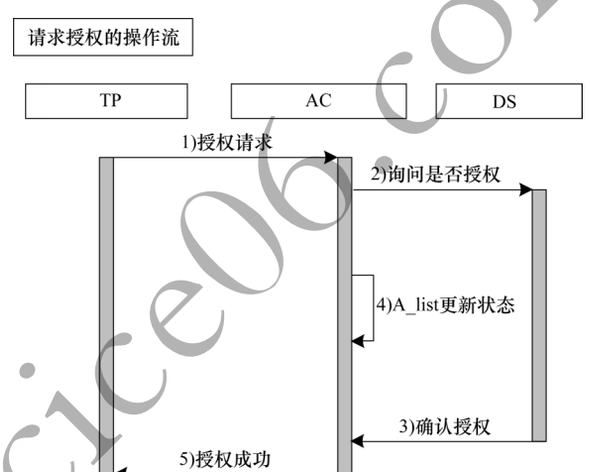

图3　DS授予TP权限的组件交互时序图

Fig.3　Component interaction sequence diagram for DS granting TP permission

算法1为授予TP权限的算法。DS同意授权TP一组权限，需要提交希望授予TP的身份和访问状态（CURD）。AC收到授权确认后，首先验证DS是否已经授权AC管理其数据（第1行）；然后确认DS和A_list中的DS相同并查看TP是否已经被授权（第2行和第3行），如果条件成立，则将TP加入tps列表（第4行），更新A_list中TP的授权状态（第5行~第10行）；最后输出success（第11行）。

算法1　授予TP权限的算法

输入　id_DS，id_TP，act/*输入DS地址和权限*/

输出　success

```
1.require(A_list[id_DS].mflag==1)
2.require(msg.sender==A_list[id_DS].DS)
3.require(A_list[id_DS].Tp[id_DS][id_TP].tpflag! =1)
4.A_list[id_DS[id_DS].push(TP)
5.A_list[id_DS].Tp[id_DS][id_TP].tp=TP
6.A_list[id_DS].Tp[id_DS][id_TP].act=at
7.A_list[id_DS].Tp[id_DS][id_TP].ds=msg.sender
```

8.A_list[id_DS].Tp[id_DS][id_TP].token = sha256(abi.encodePackedid_TP,act,block.timestamp))

9.A_list[id_DS].Tp[id_DS][id_TP].tpflag=1

10.A_list[id_DS].tpsflage++

11.return true

算法2为撤销TP访问的算法。在DS输入想要撤销的TP后,首先判断DS是否已授权AC管理其数据(第1行);然后确认DS和A_list中的DS相同(第2行),并遍历列表,找到TP在tps中的索引,若没有则返回(第3行),删除tps列表中的TP和TP的授权列表(第4行~第6行);最后输出success(第7行)。

算法2 撤销已授权TP的算法

输入 id_DS,id_TP/*输入DS和TP的地址*/

输出 success

1.require(A_list[id_DS].flag==1)

2.require(msg.sender==A_list[id_DS].DS)

3.遍历tps,找出TP在列表中的所以索引i

4.delete A_list[id_DS].tps[id_DS][i]

5.delete A_list[id_DS].Tp[id_DS][TP]

6.A_list[id_DS].tpsflage--

7.return true

3 实验结果与分析

本文实验基于以太坊平台,使用remix编译器的JavaScriptVM环境部署合约,消耗2 145 356 Gas。合约控制台界面如图4所示,其中,reqMange为请求合约管理其个人数据的函数,grant和revoc分别为授予和撤销TP访问权限的函数。DS请求AC后和授权指定TP访问权限后的响应分别如图5和图6所示。

图4 合约控制台界面

Fig.4 Contract console interface

图5 DS请求AC后得到的响应

Fig.5 The response after DS requesting AC

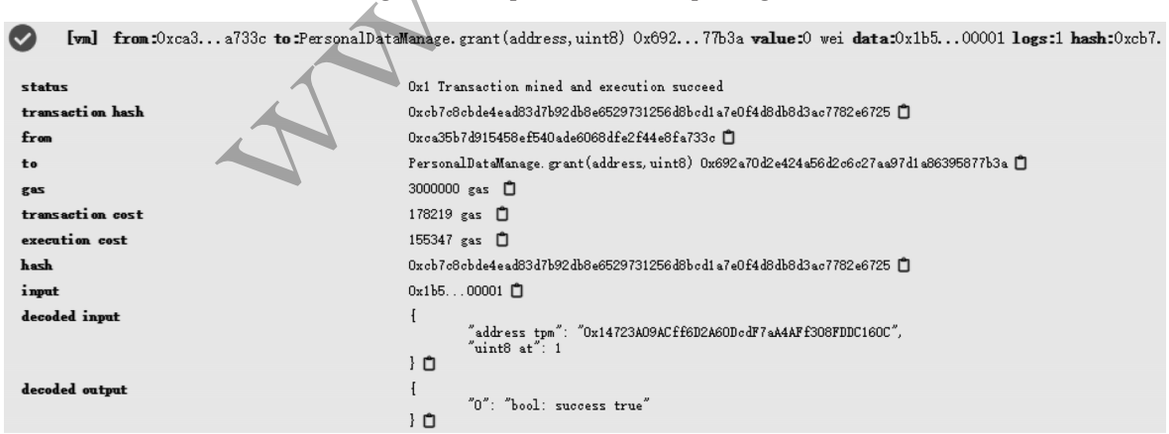

图6 授权指定TP访问权限后的响应

Fig.6 The response after authorizing specified TP access rights

本文使用remix搭建的测试网络，每次合约的调用时间约为2 s。由于测试网络节点较少，因此挖矿难度低、出块时间快。在实际应用中，随着节点数量增加，如果出块挖矿难度低，将会导致同时出块的交易过多，造成阻塞。考虑此因素，当前以太坊主网出块的平均时间已上调至15.5 s。此外，实际应用中还应考虑挖矿难度、Gas价格和网络阻塞程度。

通过使用Caliper[20]对合约进行性能测试，共进行2次测试，第1次测试分2轮，每轮500次，第2次测试分3轮，每轮50次，测试结果如表1和表2所示。

表1　合约运行500次的性能测试结果

Table 1　Performance test result after contract running 500 times

操作	成功次数	失败次数	发送速率/TPS	最大延时/s	最小延时/s	平均延时/s	吞吐率/TPS
reqMagn	1	0	1.0	0.21	0.21	0.21	4.8
getTP	500	0	297.1	46.47	18.50	42.22	10.4
getTP	500	0	307.3	44.02	18.10	40.03	11.0

表2　合约运行50次的性能测试结果

Table 2　Performance test result after contract running 50 times

操作	成功次数	失败次数	发送速率/TPS	最大延时/s	最小延时/s	平均延时/s	吞吐率/TPS
reqMagn	1	0	1.0	0.21	0.21	0.21	4.8
getTP	50	0	102.0	4.57	2.53	4.27	9.9
getTP	50	0	205.8	4.82	2.34	4.39	9.9
getTP	50	0	314.5	4.43	2.61	4.05	10.9

实验结果表明，两次测试都执行成功，在运行500次的情况下，延时性能均优于运行50次的情况，且吞吐量比较稳定，保持在10 TPS左右。

在本文系统中，DS和TP是多对多的关系，然而一个TP地址无法对应多个DS的TP_policy。针对该问题，本文通过Solidity语言的mapping制造二级映射：mapping(address=>mapping(address=>TP))。通过使用DS地址所对应TP的TP_policy来唯一标识DS对指定TP的授权。对于DS的身份认证，利用Solidity语言自身的特性msg.sender来保证授权等操作必定由DS发出，而如果使用超级账本或其他编程语言，则需要编写身份认证的合约代码。随着系统中节点数目的增多，分布式账本的数量也相应增加，只篡改少量节点的数据无法发起51%攻击，由此可逐步提高篡改区块链数据的难度。

4　结束语

在现有基于区块链的个人数据保护方案中，存在个人数据地址无法收回的问题。本文提出一种匿名地址管理方案，将个人数据的控制分权，由AC管理加密地址，由RS管理解密私钥，但双方都没有获得真正的地址，只有在授权TP访问后，TP提交加密地址给RS并由其解密地址，才能得到真正的地址并执行TP的数据请求，以此实现对个人数据的有效保护。后续将结合区块链3.0的发展趋势对本文方案进行优化，进一步提高出块速度和吞吐率。

参考文献

[1] ZHANG Liang，LIU Baixiang，ZHANG Ruyi，et al.Overview of blockchain technology[J].Computer Engineering，2019，45(5)：1-12.(in Chinese)

张亮，刘百祥，张如意，等.区块链技术综述[J].计算机工程，2019，45(5)：1-12.

[2] AZARIA A，EKBLAW A，VIEIRA T，et al.MedRec：using blockchain for medical data access and permission management[C]//Proceedings of the 2nd International Conference on Open and Big Data.Washington D.C.，USA：IEEE Press，2016：25-30.

[3] LIANG X，ZHAO J，SHETTY S，et al.Integrating blockchain for data sharing and collaboration in mobile healthcare applications[C]//Proceedings of the 28th Annual International Symposium on Personal，Indoor，and Mobile Radio Communications.Washington D.C.，USA：IEEE Press，2017：1-5.

[4] HARDJONO T.A federated authorization framework for distributed personal data and digital identity[EB/OL].(2019-06-09)[2019-12-10].https://arxiv.org/pdf/1906.03552.pdf.

[5] ZYSKIND G，NATHAN O.Decentralizing privacy：using blockchain to protect personal data[C]//Proceedings of IEEE Security and Privacy Workshops.Washington D.C.，USA：IEEE Press，2015：180-184.

[6] JANEČEK V.Ownership of personal data in the Internet of things[J].Computer Law & Security Review，2018，34(5)：1039-1052.

[7] MALGIERI G.Property and (intellectual) ownership of consumers' information：a new taxonomy for personal data[J].Privacy in Germany-PinG，2016(4)：1-17.

[8] XIAO Yang，ZHANG Ning，LOU Wenjing，et al.PrivacyGuard：enforcing private data usage control with blockchain and attested off-chain contract execution[EB/OL].(2019-04-15)[2019-12-10].https://arxiv.org/pdf/1904.07275.pdf.

（下转第187页）

（上接第181页）

[9] ALESSI M, CAMILLO A, GIANGRECO E, et al. Make users own their data: a decentralized personal data store prototype based on Ethereum and IPFS[C]//Proceedings of the 3rd International Conference on Smart and Sustainable Technologies. Washington D.C., USA: IEEE Press, 2018: 1-97.

[10] MONTJOYE Y A, SHMUELI E, WANG S S, et al. openPDS: protecting the privacy of metadata through safeanswers[J]. PloS One, 2014, 9(7): 1-9.

[11] NAKAMOTO S. Bitcoin: a peer-to-peer electronic cash system [EB/OL]. [2019-12-10]. https://bitcoin.org/en/bitcoin-paper.

[12] YUAN Yong, NI Xiaochun, ZENG Shuai, et al. Blockchain consensus algorithms: the state of the art and future trends[J]. Acta Automatica Sinica, 2018, 44(11): 93-104. (in Chinese)
袁勇, 倪晓春, 曾帅, 等. 区块链共识算法的发展现状与展望[J]. 自动化学报, 2018, 44(11): 93-104.

[13] CLACK C D, BAKSHI V A, BRAINE L. Smart contract templates: foundations, design landscape and research directions[EB/OL]. (2017-03-15)[2019-12-10]. https://arxiv.org/pdf/1608.00771.pdf.

[14] WOOD G. Ethereum: a secure decentralised generalised transaction ledger[EB/OL]. [2019-12-10]. http://gavwood.com/Paper.pdf.

[15] CACHIN C. Architecture of the hyperledger blockchain fabric [C]//Proceedings of Workshop on Distributed Cryptocurrencies and Consensus Ledgers. Chicago, USA: [s.n.], 2016: 1-5.

[16] EBERHARDT J, HEISS J. Off-chaining models and approaches to off-chain computations[C]//Proceedings of the 2nd Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers. Washington D.C., USA: IEEE Press, 2018: 7-12.

[17] TRUONG N B, SUN K, GUO Y. Blockchain-based personal data management: from fiction to solution[C]// Proceedings of the 18th International Symposium on Network Computing and Applications. Washington D.C., USA: IEEE Press, 2019: 1-8.

[18] TRUONG N B, SUN K, LEE G M, et al. GDPR-compliant personal data management: a blockchain-based solution[J]. IEEE Transactions on Information Forensics and Security, 2019, 15: 1746-1761.

[19] BENET J. IPFS-content addressed, versioned, P2P file system [EB/OL]. (2014-07-14) [2019-12-10]. https://arxiv.org/pdf/1407.3561.pdf.

[20] Hyperledger Performance and Scale Working Group. Hyperledger blockchain performance metrics[EB/OL]. [2019-12-10]. https://www.hyperledger.org/wp-content/uploads/2018/10/HL_Whitepaper_Metrics_PDF_V1.01.pdf.

编辑　金胡考