DOI: 10.3969/j.issn.0255-8297.2023.02.014

一种基于区块链的电子合同共享方案

赵海鸿[1,2]， 姚中原[1,2]， 祝卫华[1,2]，
朱自强[1,2]， 潘长风[3]， 斯雪明[1,2,4]

1. 中原工学院 前沿信息技术研究院，河南 郑州 450007
2. 河南省区块链数据共享国际联合实验室，河南 郑州 450007
3. 闽江学院 新华都商学院，福建 福州 350121
4. 复旦大学 计算机科学与技术学院，上海 201203

摘 要： 为解决电子合同在存储、共享过程中出现的数据被篡改或泄露等问题，提出了一种基于区块链的电子合同共享方案。首先，将智能合约与代理重加密技术相结合，构造出一个代理智能合约来代替传统代理重加密过程中的代理商，去中心化地实现了电子合同的安全共享。其次，利用星际文件系统（inter planetary file system, IPFS）存储电子合同密文，区块链存储电子合同索引地址，有效缓解了区块链的存储压力。最后，从方案对比、安全性等方面对所提方案进行分析。

关键词： 电子合同；区块链；智能合约；代理重加密；星际文件系统
中图分类号： TP311.13 **文章编号：** 0255-8297(2023)02-0359-10

An Electronic Contract Sharing Scheme Based on Blockchain

ZHAO Haihong[1,2], YAO Zhongyuan[1,2], ZHU Weihua[1,2],
ZHU Ziqiang[1,2], PAN Changfeng[3], SI Xueming[1,2,4]

1. The Frontier Information Technology Research Institute, Zhongyuan University of Technology, Zhengzhou 450007, Henan, China
2. Henan International Joint Laboratory of Blockchain and Data Sharing, Zhongyuan University of Technology, Zhengzhou 450007, Henan, China
3. New Huadu Business School, Minjiang University, Fuzhou 350121, Fujian, China
4. School of Computer Science, Fudan University, Shanghai 201203, China

Abstract: In order to solve the problems of data tampering or leakage in the storage and sharing of electronic contracts, an electronic contract sharing scheme based on blockchain is proposed. First, a proxy smart contract is constructed by combining the contract with the proxy re-encryption to replace the traditional proxy, and the secure sharing of electronic

收稿日期： 2021-11-30
基金项目： 河南省重大公益专项（No. 201300210300）；河南省网络密码技术重点实验室开放课题（No. LNCT2019-A07）；河南省高等学校重点项目基金（No. 19A520047）资助
通信作者： 姚中原，研究方向为密码技术与应用、区块链技术等。E-mail: yaozhongyuan@zut.edu.cn

contract is decentralized. Inter planetary file system (IPFS) is then used to store the ciphertext of electronic contract, and the electronic contract index address is stored in the blockchain, which effectively alleviates the storage pressure of the blockchain. Finally, the security performance of the proposed scheme is analyzed.

Keywords: electronic contract, blockchain, smart contract, proxy re-encryption, inter planetary file system (IPFS)

随着技术的不断发展，纸质合同逐步演变成为电子合同，但是电子合同在带来极大便利的同时，也引发了一系列问题：1）电子合同多基于中心化机构存储管理，容易遭受攻击从而导致合同内容篡改、泄露；2）电子合同存储在第三方，拥有者不能掌握合同主导权，无法知晓合同能被哪些人访问查看；3）共享的合同若要验证真伪，会增加大量的时间和人力成本。

区块链这种新技术的出现为解决上述问题提供了新思路[1]，目前广泛应用于多种场景中[2-4]。在电子数据共享方面[5-6]，文献[7]提出了一个电子病历共享方案，借助区块链技术和代理重加密技术，通过搜索陷门实现电子病历数据的安全共享，解决了医生和患者之间电子数据流通难的问题。针对在不可信环境下实现数据的共享，文献[8]将Schnorr签名技术与代理重加密相结合，实现了数据的安全共享。同时借助区块链来维护可信账本，实现了数据的可追溯。文献[9]基于区块链与条件代理重加密技术提出了一个共享方案，结合分布式密钥生成技术来解决中心化的密钥托管问题，最终实现了数据的安全共享。然而，这些方案大多依赖云服务器，一旦云服务器遭到攻击或出现故障就会造成电子合同数据丢失、泄露等后果。

在电子数据结合区块链的相关研究中[10-12]，考虑到集中化的数据存储方式易受到攻击，文献[13]设计了一个电子合同管理模型，引入区块链技术来实现电子合同的安全存储，结合智能合约实现了电子合同签订执行的自动化。文献[14]设计了一个基于区块链的电子合同平台以实现电子合同数据的不可篡改，通过身份认证来保证平台中电子合同数据的安全及可追溯。文献[15]基于区块链技术设计了一个电子合同系统，通过超级账本进行系统开发，用智能合约实现合同数据的集中化统一处理，保证了安全存储。但上述文献更偏向于对电子合同进行安全存储，并没有涉及到电子合同的共享。

本文基于区块链和代理重加密技术提出了一种电子合同共享方案，借助智能合约充当代理商的角色，实现了去中心化的安全共享。将电子合同密文存储于星际文件系统（inter planetary file system, IPFS）中，缓解区块链存储压力的同时降低了存储成本。

1 相关技术

1.1 区块链技术

区块链的概念是一位名为中本聪的学者在其发布的文章《比特币：一种点对点的电子现金系统》中首次提出的[1]。它结合了密码学算法、共识机制、P2P对等网络等多种技术，其本质是一种去中心化、不可篡改的分布式账本[16]。在区块链网络中，若干笔交易被打包成一个区块，区块与区块之间通过hash值相连从而形成不可篡改的区块链网络，结构如图1所示。

1.2 智能合约

智能合约的概念最早由Szabo[17]提出，定义为“一套以数字形式指定的承诺，包括合约参与方可以在上面执行这些承诺的协议”。智能合约起初应用于法律领域，但与区块链的结合则让智能合约发挥出更强大的优势。简单来说，智能合约就是部署在区块链上的一段程序代码。按照业务逻辑将约定的承诺数字化，一旦触发预设的条件，智能合约就能够准确无误地自动执行，最终完成严格的验证和计算[18]。

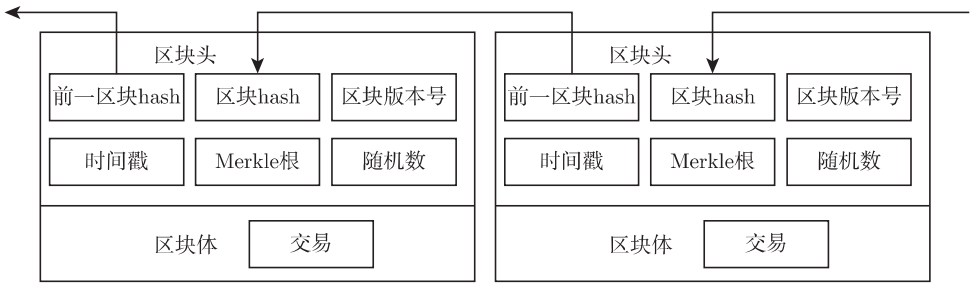

图 1　区块链结构

Figure 1　Blockchain structure

1.3　代理重加密技术

1998 年，Blaze 等 [19] 首次提出代理重加密的概念。代理重加密可以在密文状态下实现数据共享，共享过程中无需将密文解密从而保证了数据的安全性。在代理重加密过程中，数据访问者 Bob 向数据拥有者 Alice 发送访问请求并发送自己的公钥。Alice 通过自己的私钥及 Bob 的公钥为其生成代理重加密密钥，并与密文一同发给半可信代理商。半可信代理商进行重加密操作，将生成的重加密密文发给访问者。访问者使用自己的私钥即可对密文进行解密，从而在代理商不知道明文数据的情况下实现数据的安全共享。

本系统构造出代理智能合约的角色，以实现传统代理重加密过程中半可信代理商的功能。电子合同拥有者在共享电子合同时，只需将重加密密钥发送给代理智能合约。代理智能合约被触发后，按照设定的逻辑规则自动进行重加密操作，之后将重加密密文返回给申请者。

1.4　IPFS

IPFS 是一个基于内容寻址的分布式文件系统 [20-21]。IPFS 在存储数据时使用哈希去重的方式，首先对存储的文件做 hash 处理，并将得到的文件 hash 值存储于分布式哈希表中。根据 hash 值判定两个文件是否相同，即同一份文件只保存一次，降低了系统冗余度，节省了大量存储空间，也大幅降低了数据的存储成本。文件上传至 IPFS 时，会返回该文件的唯一索引地址，通过该索引地址即可访问文件。若对上传的文件做任意修改，则会返回完全不同的索引地址。传统的中心化存储机制易遭受攻击，而上传至 IPFS 存储的数据是分布式存储的。因此 IPFS 的这些特性可与区块链有很好的结合，在降低存储成本的同时为电子合同提供安全分布式存储。

2　共享方案

2.1　方案模型

本方案模型如图 2 所示，包括电子合同拥有者、电子合同访问者、代理智能合约、区块链网络和 IPFS，电子合同拥有者及电子合同访问者均有自身的属性。

2.1.1　电子合同拥有者

电子合同拥有者对电子合同拥有主导权，并根据属性信息为电子合同设定访问控制规则。拥有者对合同明文进行哈希操作生成 hash 值，使用随机生成的临时密钥 k 加密电子合同，并将生成的密文 cph 上传至 IPFS。之后用自己的公钥将 k 和 IPFS 返回的索引地址加密生成密钥密文 Kcph，与 hash 值一起上传保存在区块链网络中，实现上链数据的不可篡改。

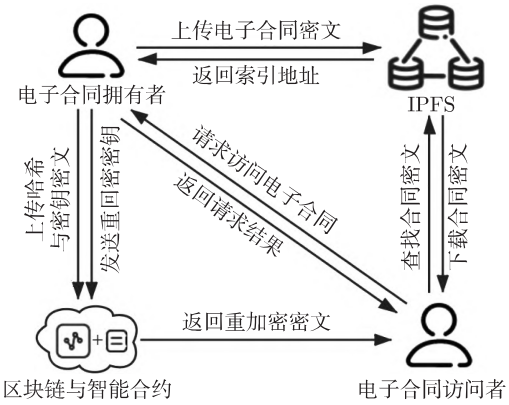

图 2 方案模型

Figure 2 Scheme model

2.1.2 电子合同访问者

在申请访问电子合同时，若属性符合设定的访问要求，则拥有者在区块链网络中调用该访问者的公钥，为其在本地生成重加密密钥，再将重加密密钥发送给代理智能合约。若不满足属性要求，则直接拒绝该申请。访问者收到重加密密文后即可解密，最终可获取电子合同的明文数据。访问者解密获取电子合同明文后对其进行哈希操作，获取该合同的 hash 值，可与区块链上存储的 hash 值进行对比从而验证电子合同的一致性。

2.1.3 代理智能合约

充当传统代理重加密中的代理商角色，在电子合同的共享过程中进行重加密操作。当接收到拥有者发送的重加密密钥后，自动被触发调用区块链上拥有者上传的密钥密文 Kcph 进行重加密操作，之后将重加密密文 Rcph 返回给访问者。

2.1.4 区块链网络

仅存储电子合同的 hash 值及对应的密钥密文 Kcph，极大地减小了区块链的存储压力。在电子合同的共享过程中，电子合同拥有者不必向代理智能合约发送密钥密文 Kcph，而是由代理智能合约从区块链网络中调用，从而可节省共享过程中的交互开销。

2.1.5 IPFS

IPFS 取代传统的云服务器，提供电子合同密文的分布式安全存储。存储于 IPFS 的电子合同密文都有唯一的索引地址，访问者可通过索引地址在 IPFS 上查找并获取电子合同密文。

2.2 方案描述

本文定义电子合同拥有者为 O，电子合同访问者为 V，代理智能合约为 R。本方案分为 4 个阶段：初始化阶段、电子合同上传阶段、电子合同请求阶段和电子合同获取阶段。

2.2.1 初始化阶段

1) Setup(1^k) 系统输入安全参数 1^k，输出全局参数 $\text{par}=(q,G,g,H_1,H_2)$。其中 G 是一个阶为素数 q 的循环群，g 是 G 的一个生成元。定义两个 hash 函数：$H_1:\{0,1\}^*\rightarrow \mathbf{Z}_q^*$，$H_2:G\rightarrow \mathbf{Z}_q^*$。

2）KeyGen(par) 电子合同拥有者 O 随机选择 $o \in \mathbf{Z}_q^*$ 作为私钥 sk_o，计算公钥 $pk_o = g^o$。电子合同访问者 V 随机选择 $v \in \mathbf{Z}_q^*$ 作为其私钥 sk_v，计算公钥 $pk_v = g^v$。

2.2.2　电子合同上传阶段

电子合同拥有者使用函数 H_2 对电子合同进行哈希操作，生成合同的 hash 值。之后随机产生一个对称密钥 k，使用 k 加密电子合同 m，得到电子合同密文 $\text{cph} = \text{Enc}(k, m)$。拥有者上传密文 cph 至 IPFS，获得 IPFS 的索引地址 url，并使用自己的公钥 pk_o 对 url 及对称密钥 k 加密成密钥密文 $\text{Kcph} = \text{Enc}(pk_o, k\|\text{url})$。最后拥有者将 hash 值与密钥密文 Kcph 上传至区块链网络。

2.2.3　电子合同请求阶段

访问者请求访问电子合同。智能合约可被自动触发，检验访问者的属性信息，判断是否符合设定规则。若访问者的属性满足设定要求，拥有者利用自己的公私钥对 (pk_o, sk_o) 与访问者的公钥 pk_v，为访问者生成重加密密钥 $rk_{o\to v} = \text{ReKeyGen}(pk_o, sk_o, pk_v)$。之后拥有者只需将重加密密钥 $rk_{o\to v}$ 发送给代理智能合约 R。代理智能合约收到重加密密钥 $rk_{o\to v}$ 后被自动触发，调用区块链上的密钥密文 Kcph 进行重加密操作，将生成的重加密密文 $\text{Rcph} = \text{ReEnc}(rk_{o\to v}, \text{Kcph})$ 返回给访问者。

2.2.4　电子合同获取阶段

访问者收到重加密密文 Rcph 后，使用私钥 sk_v 对其解密 $\text{Dec}(sk_v, \text{Rcph}) \to k, \text{url}$，得到对称密钥 k 及 IPFS 索引地址 url。通过 url 可从 IPFS 下载获取电子合同密文 cph，通过密钥 k 可得到明文 $m = \text{Dec}(k, \text{cph})$。访问者若对解密的电子合同存疑，可对电子合同明文 m 进行哈希运算得到 hash 值，通过核验智能合约与区块链上存储的 hash 值进行对比，从而验证电子合同的真伪。

2.3　智能合约设计

本文采用 HyperLedger Fabric 作为开发环境。Fabric 通过智能合约可以读取和修改账本数据，两者进行交互时需要通过 Fabric 提供的系统包——shim 包。Shim 包提供了交互的接口，可以方便地操作 Fabric 中的账本数据。本节主要介绍五个核心智能合约的设计：上链智能合约、查询智能合约、授权智能合约、代理智能合约和核验智能合约。

2.3.1　上链智能合约

上链智能合约用于将电子合同的 hash 值以及密钥密文 Kcph 上传至区块链。首先通过 shim 包的接口获取上传的参数信息，经判断符合设定要求后，即可调用 Fabric 中的 PutState 方法将其上链，最后返回上链信息。上链算法流程如算法 1 所示。

算法 1　上链

Input: ID, hash, TimeStamp

Output: contract details

while the arguments input are correct **do**

```
ContractInfoJsonBytes, err: = dataProcessing(args)
if err != nil then
    return shim.Error(err, Error())
else
    err = stub.PutState(ContractInfoJsonBytes)
```

return contract details on-chain
end
end

2.3.2 查询智能合约

电子合同查询智能合约通过合同的唯一ID值进行搜索，经判断输入的查询信息符合要求后，通过调用Fabric中的GetState方法从账本中获取数据，判断是否存在错误，最后返回查询结果，查询流程如算法2所示。

算法 2 查询

Input: ID
Output: query results
if the arguments input are correct **then**
Avalbytes⇓stub.GetState(ID)
return query results
else
return an error message
end

2.3.3 授权智能合约

Fabric可以使用基于属性的方式来对电子合同访问者的访问权限进行细粒度控制。智能合约中进行权限控制需要通过客户端身份库，该库提供了获取访问者的身份属性的相关方法。电子合同授权智能合约通过调用cid.AssertAttributeValue方法来确定访问者是否满足设定的属性要求。若属性不符合用户设定要求，则拒绝对该访问者授权访问。相关流程如算法3所示。

算法 3 授权

Input: attribute information
Output: authorization results
while
err⇓cid.AssertAttributeValue(stub.”Att.init”, ”ture”)
do
if attributes meet the requirement **then**
return the authorization results
else
deny the authorization
end
end

2.3.4 代理智能合约

若访问者的属性满足电子合同的设定要求，则拥有者为其生成重加密密钥$rk_{o\to v}$，代理智能合约在接收到用户发送的代理重加密密钥后，被自动触发执行。调用区块链上存储的密钥密文Kcph进行代理重加密操作，生成重加密密文返回给访问者。代理重加密流程如算法4所示。

算法 4　代理重加密

Input: $rk_{o\to v}$, Kcph

Output: Rcph

```
while the arguments input are correct do
        Getparams: = args[0]
        Recph, err: = ReEncrypt(stub, Getparams)
        if err != nil then
            return shim.Error(err, Error())
        else
            return shim.success(Rcph)
        end
    end
```

2.3.5　核验智能合约

用户若对于获取到的电子合同存在疑问，可通过核验功能来鉴定电子合同的真伪。用户上传存疑的电子合同，核验智能合约通过 GetState 方法来获取电子合同存于区块链上 hash 值，与访问者上传的电子合同 hash 值进行对比，最后返回给访问者核验结果。核验流程如算法 5 所示。

算法 5 核验

Input: ID, uploadHash

Output: False or Varified

```
while the arguments input are correct do
        uploadHash: = args[0]
        queryHash, err: = GetState(ID)
        if err != nil then
            return shim.Error(err, Error())
        else if uploadHash != queryhash then
            return False
        else
            return Varified
        end
    end
```

3　测试与分析

3.1　性能测试

本方案基于 HyperLedger Fabric 作为测试环境，智能合约使用 Go 语言进行开发。本实验的硬件配置为：CPU 为 Intel® Core™ i5-8400M CPU@2.80 GHz，RAM 为 16 GB，操作环境为 VMware Workstation 15.0.2 上安装的 Ubuntu 16.04.6 LTS。

本方案使用 tape 进行性能测试。它是一个极简的轻量级测试工具，能够快速准确地获取性能测试结果。在测试中，将数据大小设置为 100 KB 逐次增加到 500 KB，每次测试的增量为 100 KB，测试出上传时间、查询时间以及核验时间的结果如图 3 所示。可观察到，智能合约的执行时间随着数据量的增加也在逐步增加，但是时间增幅较缓，所需时间也均在合理范围内。

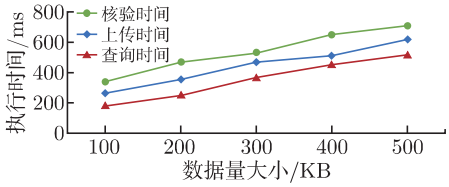

图 3 智能合约执行时间

Figure 3 Smart contract execution time

在测试中，将电子合同交易数量由 100 笔增加到 500 笔，每次测试的增量为 100 笔，交易速度的测试结果如图 4 所示。可观察到随着交易数量的增加，执行时间也在相应增加，但所需时间均在合理范围内。

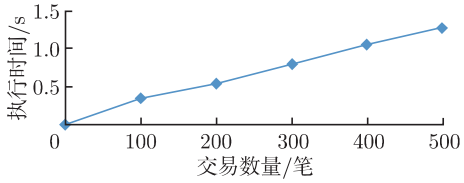

图 4 交易速度

Figure 4 Transaction speed

依旧将电子合同交易数量值设定为由 100 笔递增至 500 笔，测得吞吐量结果如图 5 所示。可观察到交易数量从 200 笔逐步增加到 500 笔时，吞吐量并没有很大的改变，保持相对稳定。测试结果显示该方案可满足一般系统的需求。

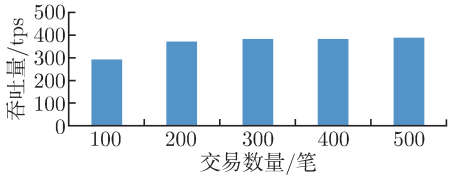

图 5 每秒处理的交易数

Figure 5 Transactions per second

3.2 安全性分析

3.2.1 机密性

在本方案中，电子合同文件加密保存于 IPFS 中。结合智能合约与代理重加密算法，实现对称密钥的安全传输。电子合同密文的 IPFS 索引地址存储在区块链上，从而避免了中心化的存储方式易遭受攻击的问题。攻击者无法获取密文，也无法将密文转化为明文，电子合同的机密性得到了保障。

3.2.2 完整性

电子合同密文保存在 IPFS 上，与返回的唯一索引地址形成对应。若密文遭到篡改，则其对应的索引地址也会变化。同时，电子合同明文的 hash 值存储于区块链网络中，访问者在获

取电子合同数据后，也可通过对比 hash 值来验证合同明文是否一致，从而保证了电子合同的完整性。

3.3　方案比较

针对本文提出的电子合同共享方案，通过与其他同类型的电子数据共享方案进行对比，结果如表 1 所示。

表 1　方案对比

Table 1　Schemes comparison

方案	去中心化存储	引入智能合约	数据拥有者具有主导权	去中心化代理重加密
文献 [7]	否	否	是	否
文献 [8]	否	是	是	否
文献 [9]	是	否	是	否
本文	是	是	是	是

首先，从数据的存储角度来看，文献 [7-8] 都是采用第三方云服务器存储数据密文，容易遭受攻击导致数据泄露。本文与文献 [9] 均将密文数据存储于 IPFS 中，实现分布式安全存储。其次，从引入智能合约角度来看，文献 [7-9] 均未设计智能合约。文献 [8] 引入智能合约来管理元数据及访问控制数据，避免人工失误造成错误。本文借助智能合约取代代理商，实现去中心化的代理重加密。再次，从数据拥有者是否具有主导权的角度来看，本文与文献 [7] 都要求访问者在请求访问数据时需要获得拥有者同意才能进行后续操作。文献 [8] 中数据拥有者将访问控制权限上传至区块链，只有满足访问权限的访问者才可下载访问。文献 [9] 用条件代理重加密算法可以让拥有者有指派地共享数据，实现了拥有者对数据的主导权。最后，从代理重加密的去中心化角度来看，上述文献在代理重加密过程中均需要第三方代理商的参与，无法保证代理商的完全可信。通过与上述文献中的方案对比，本文提出的共享方案结合智能合约与代理重加密技术可以实现去中心化，并且能够安全存储电子合同密文数据，与访问者进行可信安全共享。

4　结　语

本文针对电子合同在存储和共享过程中出现的数据被篡改或泄露等问题，提出了一种基于区块链的电子合同共享方案。在传统的代理重加密过程中需要一个半可信的代理商角色，而这一需求与区块链的去中心化特性相矛盾。因此本文构造出一个代理智能合约，能够去中心化地实现数据的安全共享。采用 IPFS 与区块链相结合的存储方式可以实现电子合同的安全存储，缓解区块链存储压力的同时降低了存储成本。最后对本方案进行性能测试，并与其他方案进行对比，结果表明该方案可以较好地实现数据的安全共享且满足应用需求。

参考文献:

[1] Nakamoto S. Bitcoin: a peer-to-peer electronic cash system [EB/OL]. [2021-06-15]. https://bitcoin.org/en/bitcoin-paper.

[2] Kang J W, Yu R, Huang X M, et al. Blockchain for secure and efficient data sharing in vehicular edge computing and networks [J]. IEEE Internet of Things Journal, 2019, 6(3): 4660-4670.

[3] Ekblaw A, Azaria A, Halamka J D, et al. A case study for blockchain in healthcare: "MedRec" prototype for electronic health records and medical research data [C]//IEEE Open & Big Data Conference, 2016: 1-13.

[4] 周正强, 陈玉玲, 李涛, 等. 基于联盟链的医疗数据安全共享方案 [J]. 应用科学学报, 2021, 39(1): 123-134.
Zhou Z Q, Chen Y L, Li T, et al. Medical data security sharing scheme based on consortium blockchain [J]. Journal of Applied Sciences, 2021, 39(1): 123-134. (in Chinese)

[5] Xia Q, Sifah E B, Asamoah K O, et al. MeDShare: trust-less medical data sharing among cloud service providers via blockchain [J]. IEEE Access, 2017, 5: 14757-14767.

[6] Hawlitschek F, Notheisen B, Teubner T. The limits of trust-free systems: a literature review on blockchain technology and trust in the sharing economy [J]. Electronic Commerce Research and Applications, 2018, 29: 50-63.

[7] 牛淑芬, 刘文科, 陈俐霞, 等. 基于代理重加密的电子病历数据共享方案 [J]. 计算机工程, 2021, 47(6): 164-171.
Niu S F, Liu W K, Chen L X, et al. Data sharing scheme of electronic medical record based on proxy re-encryption [J]. Computer Engineering, 2021, 47(6): 164-171. (in Chinese)

[8] 李莉, 曾庆贤, 文义红, 等. 基于区块链与代理重加密的数据共享方案 [J]. 信息网络安全, 2020, 20(8): 16-24.
Li L, Zeng Q X, Wen Y H, et al. Data sharing scheme based on the blockchain and the proxy re-encryption [J]. Netinfo Security, 2020, 20(8): 16-24. (in Chinese)

[9] 唐飞，陈云龙, 冯卓. 基于区块链和代理重加密的电子处方共享方案 [J]. 计算机科学, 2021, 48(S1): 498-503.
Tang F, Chen Y L, Feng Z. Electronic prescription sharing scheme based on blockchain and proxy re-encryption [J]. Computer Science, 2021, 48(S1): 498-503. (in Chinese)

[10] Franks P C. Implications of blockchain distributed ledger technology for records management and information governance programs [J]. Records Management Journal, 30(3): 287-299.

[11] Wang X, Feng L B, Zhang H, et al. Human resource information management model based on blockchain technology [C]//2017 IEEE Symposium on Service-Oriented System Engineering (SOSE). IEEE, 2017: 168-173.

[12] Yue X, Wang H J, Jin D W, et al. Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control [J]. Journal of Medical Systems, 2016, 40(10): 218.

[13] Guo L L, Liu Q F, Shi K, et al. A blockchain-driven electronic contract management system for commodity procurement in electronic power industry [J]. IEEE Access, 2021, 9: 9473-9480.

[14] Ma F, Tang N, Xu R, et al. Electronic contract ledger system based on blockchain technology [J]. Journal of Physics: Conference Series, 2021, 1828(1): 012112.

[15] 尹稚淳. 基于区块链技术的电子合同系统设计与实现 [D]. 沈阳: 沈阳师范大学, 2018.

[16] 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 Y, Wang F Y. Blockchain: the state of the art and future trends [J]. Acta Automatica Sinica, 2016, 42(4): 481-494. (in Chinese)

[17] Szabo N. Smart contracts in essays on smart contracts [J]. Commercial Controls and Security, 1994, 2(9): 1-22.

[18] 向伟静, 蔡维德. 法律智能合约平台模型的研究与设计 [J]. 应用科学学报, 2021, 39(1): 109-122.
Xiang W J, Cai W D. Research and design of legal smart contract platform model [J]. Journal of Applied Sciences, 2021, 39(1): 109-122. (in Chinese)

[19] Blaze M, Bleumer G, Strauss M. Divertible protocols and atomic proxy cryptography [C]//International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 1998: 127-144.

[20] Benet J. IPFS-content addressed, versioned, P2P file system [DB/OL]. 2014 [2021-11-30]. https://arxiv.org/abs/1407.3561.

[21] Ali M S, Dolui K, Antonelli F. IoT data privacy via blockchains and IPFS [C]//Seventh International Conference on the Internet of Things, 2017: 1-7.

(编辑：王 雪)