BDRM: A Blockchain-based Digital Rights Management Platform with Fine-grained Usage Control

Xie Fei

College of Information Science and Technology and the College of Cyber Security, Jinan University, Guangzhou 510632, China.

xiefei.jnu@foxmail.com

Abstract

This Digital rights management(DRM) enables data owners to protect their digital rights and maintain control over distribution. The majority of existing DRM systems store digital files in the third intermediary, which causes that the owners do not know the exact number of copies. The blockchain-based DRM schemes have been proposed to address the weakness of the untrusted intermediary. However, none of them have considered the problem of the rights transaction with fine-grained usage control which is to facilitate the transfer of rights and the flexible distribution of digital resources. In this paper, we propose BDRM, blockchain-based digital rights management with the property of fine-grained usage control. We first describe the architecture of BDRM and introduce three challenges we aim to solve. Then, we present our scheme which combines rights state machine with authorization tree to ensure content providers to fine-grained control their content. In particular, we implement a real digital rights management system and conduct large number of experiments to verify the feasibility and performance based on blockchain.

Keywords

DRM, blockchain, smart contract, fine-grained usage control.

1. Introduction

In recent years, digital resource has become of great important. The number of digital content users shows a exploding trend. According to the report released by [26], at the end of 2017, the cumulative user scale of digital publishing industry have reached 1.825 billion in China. Internet journals earned 2.01 billion yuan and E-book revenues were 5.4 billion yuan. Thus, it is a critical issue to protect the copyright of digital content nowadays.

DRM is a concept that spans the entire lifecycle of digital resources production which covers the entire digital resource value chain: storage, distribution,reception, playback and display. The traditional DRM systems employ watermarking, encryption, safe container and mobile agent to protect digital rights.However, there still exist some issues in traditional centralized DRM systems. First, the process of copyright circulation is non-transparent in the existing scheme for digital content [23]. For example, there is a music copy stored in the cloud server, but the owners have no idea about the number of the copies that really are sold. Meanwhile, DRM relies on the centralized server to store the content, which is subject to the weakness of single point of failure.

Blockchain technology has provided a secure method to realize DRM in a decentralized way. Blockchain was coined by Nakamoto in [19]. After it was published in 2008, both academic and industrial area show a lot of interests in blockchain. A series of blockchain-based solutions [9, 12, 24, 25, 27] have been proposed. However, the minority of these solutions seldom consider digital rights _ne-grained usage control problem. Fine-grained usage control is a basic and important function for DRM application. It contributes to the transfer of rights and the exible distribution of digital resources. Moreover, it makes digital resources protection more e_ective and more complete. There exists an open problem that traditional DRM systems lack a _ne-grained usage control model [28]. Therefore, it is necessary to design a secure and e_cient _ne-grained usage control mechanism for DRM system.

The purpose of this paper is using blockchain to solve problems in traditional DRM system. The contributions of this paper are as follow: (1) We design a blockchain-based DRM framework, which supports user digital rights registration, digital rights transactions and digital rights tracking. 2) We design the state machine and authorization tree to ensure the _ne-grained usage control of digital rights. 3) We implement a prototype system with consortium blockchain to verify and test the feasibility, availability and performance of our scheme.

The rest of the paper is organized as follows: section II describes the motivation and background of this paper. The related works are given in section III. Section IV introduces the design of BDRM. The concert scheme is described in section V. In section VI, we conduct security and performance analysis. We concluded this paper in section VII.

2. Background and assumption

In this section, we present the motivation for building the DRM system and the relevant background.

2.1 Digital Rights Management (DRM)

DRM is a set of usage control technologies which plays an important role in restricting the use of proprietary hardware and copyrighted content [1]. Traditional DRM contains a set of users including creator, rights holder, media distributor, trust third-party(TTP) and the content user [4]. Creator is responsible for creating the work. Rights holder is the corresponding owner. Media distributor is responsible for distributing the content. TTP is to ensure that the transactions between contents owner and consumers will be carried out legally. The content user makes use of digital content.

The current research on DRM contain contigency and passivity, such as tracking copyright infringement. By utilizing ID-based public key and group signature protocol in [28], Onino Policy Administration(OPA) and watermarking present an e_ective way to track copyright infringement.

2.2 Blockchain.

Blockchain is a decentralized distributed ledger. It is the core technology of cryptocurrency and has high fault tolerance. Blockchain nodes (which is called miners in Bitcoin) compete for _nding a target nonce by making the generated hash value below specific difficulty.

In general, blockchain can be divided into three categories: public blockchain,consortium blockchain and private blockchain. Public blockchain is the blockchain network where anyone can join in maintaining the network. Users can read and post transactions, and participate in the consensus process. Public blockchain is completely decentralized. In consortium blockchain, the consensus process is controlled by some pre-selected nodes and the data in block can be only read by those authorized participants. Consortium blockchain is partially decentralized. The most famous consortium blockchain platform is Hyperledger [10]. Private blockchain is the blockchain in which writing permission are only controlled by one organization and reading permission is open to the public or restricted.

Ethereum is firstly introduced by Vitalik Buterin [6], which is known as Turing-complete public blockchain project. One of the main features in Ethereum is that it supports smart contract. Smart contract is a computer protocol which supports automatically execution in a trusted environment. The advent of blockchain has built the decentralized trusted environment for smart contract. Due to the features like non-repudiation, traceability and supporting smart contract, blockchain technology can be used to build a more secure and reliable DRM system in real world.

InterPlanetary File System(IPFS) is a peer-to-peer distributed _le system which is similar to the World Wide Web [5]. It is also an decentralized application based on blockchain technology and has no single point of failure.Meanwhile, the nodes in IPFS do not need to trust each other, which provide an eficient and secure way to save the digital content.

2.3 Security Assumption

The assumptions are highlighted for the security of system in this section. First, we consider that the system is implemented with the architecture of consortium blockchain. There exist numerous roles to

maintain the network, like publishing company and school. They organized together by blockchain technology. It can be assumed that most of blockchain nodes inside the system are honest. Second, the platform are majority honest assumption and can resist most of blockchain attacks, for example 51% attacks, Finney attack. Third, in order to ensure digital rights can be traced, we assumption that the digital watermarking algorithm used in the platform can be resist some current attacks in a certain extent.

3. Related Work

3.1 Digital Rights Management

There exist numerous overviews related with DRM. A review of current state of DRM was presented in [15], which included DRM security technologies, legal implications and the main hindrance of DRM deployment. The legal issues about DRM were also mentioned in [4], further Alapan Arnab et al. presented a great overview on the distribution of DRM and rights expression languages(REL) and XrML. The open issues and challenges in digital rights management ecosystem were introduced in [28], including lack of formalized _ne-grained usage control models and cross-domain security management, trust issues in multimedia social networks for digital communities and security risk management of digital content and rights redistribution. The challenges to the resales of digital music were discussed in [23].

Plentiful solutions were proposed for the issues in DRM system. Memon et al. proposed an interactive buyer-seller protocol for the problem that the buyer believes that the seller may disclose content with the same watermark to others in [18]. However, this scheme was inefficiencies in practice. Stefan Katzenbeisser et al. proposed a secure watermark embedding which was adequate for distribution system in practice in [11]. A security architecture was proposed to control or track digital information in [20]. To solve the DMR problem in digital home network, a role-based cross-domain usage control model was presented in [29]. Through two features of role-based Usage control and security domain constraint management, the proposed scheme can implement the transfer of digital rights license.

3.2 DRM and Blockchain

Several research have been proposed to combine DRM with blockchain technology. A new mechanism for protecting the DRM contract was introduced in [24]. In order to solve the problems such as coupling videos with rights information on the blockchain and latency of reecting the rights information, a concept for a new rights management based on the blockchain was proposed in [9]. Kishigami et al. proposed a digital block distribution system based on distributed blockchain and developed a system prototype [12]. A blockchain based license structure was designed in [27], but there was a disadvantage that this platform must have high performance to handle high concurrent acquisition keys. A network media's digital rights management scheme based on blockchain was proposed in [25], however, the design was nonholonomic because only transaction information was recorded on the blockchain. After DRM requirement and suitbility were analyzed in [17], Z. Ma et al. proposed a blockchain-based scheme for DRM, which supports for identity and privacy protection and trace with conditional identity management. The state machine is used to build a naming and storage system in [3] and crowdsourcing system based on blockchain in [14], which are helpful to construct a blockchain-based DRM platform.

4. Design of BDRM

The design building DRM system based on blockchain BDRM is given in this section. BDRM is designed to implement a DRM system in an application layer atop on consortium blockchain.

As shown in Fig.1, BDRM contains three entities: content provider (CP), content consumer(CS) and BDRM. CP, who wants to protect or sell these rights in BDRM, creates the digital content and provides the digital content for the platform. CS pays for the digital content and uses digital content. BDRM receives the digital content from CP, provides distribution channels for CP and CS. It is also responsible for handling the _nancial transaction and tracking the rights transfer process. For example,

a user named Alice uploads a digital content in BDRM. BDRM will detect the content and then register if it passes the detection. When another user called Bob wants to purchase the rights from Alice, BDRM will detect the state and the attributes of Alice's rights. If Bob passes the authentication and detection, the rights will be sold successfully. Finally, BDRM will distribute digital content to Bob. More details will be described in the following sections.

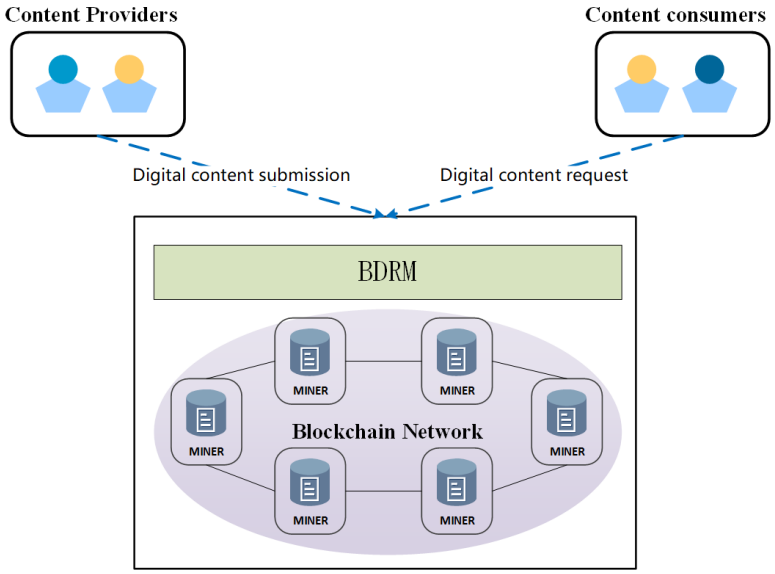

Fig. 1 The system model of BDRM

4.1 Technical Challenges

The challenges of building digital rights management systems based on blockchain are as follows:

Design of Fine-grained Usage Control: Usage control technology has been applied in traditional DRM system [22]. But usage control in traditional DRM system is not completely suitable in blockchain architecture. Moreover, traditional DRM system lacks a _ne-grained usage control protocol. Therefore, it is difficult to design a fine-grained usage control model in DRM system based on blockchain.

Limitation on Rights Authorization: TTP in DRM system is responsible for authenticating transfer rights. However, blockchain is decentralized, so there does not exist fully trusted party. Therefore, how to transfer rights successfully and smooth execution of the transaction will be a challenge in our model.

4.2 Overview of BDRM Architecture

BDRM is a decentralized application that runs on the application layer atop on blockchain. It utilizes smart contract to store the key information of the digital rights and records copyright transactions for users. Specifically, a novel authorization tree is designed in the blockchain. Each time a user conducts a rights transaction, the information of the copyright authorization tree will be updated automatically. By employing smart contract, BDRM can achieve copyright management related operations, such as copyright registration and copyright transactions. As shown in Fig.2, BDRM mainly contains three layers: application layer, blockchain layer and storage layer.

4.2.1 Layer1: Application Layer

The application layer occupies the highest tier. It serves three purposes: 1) receiving digital rights content data from outside and registering content on the blockchain once the digital content passes similarity detection. 2) recording digital rights transactions between CP and CS in this layer. 3) content owner can change the digital rights status in this layer. We design that digital rights operations can only be performed at this level.

4.2.2 Layer2: Blockchain Layer

The main logic on transaction lies in the blockchain layer. Part of the function execution in the application layer will correspond to a transaction and the transaction will be recorded on the blockchain perpetually. BDRM adopts Parity to build the underlying blockchain. Parity supports Turing-complete script language, which has a rich a set of data types such as map, array and composite structures [8]. It has been verified that Parity has the lowest latency [7]. More importantly, we can deploy consortium blockchain by utilizing Parity which implements Proof-of-Authority(POA) [21].

4.2.3 Layer3: Storage Layer

The storage layer occupies the lowest tier, which stores the original digital rights content. All the stored contents are encrypted with the secret key of content owner. By storing digital content outside of the blockchain, BDRM allows digital content of arbitrary size. The owners of digital content do not need to worry about their digital content being leaked. Even though other users have the downloading address, the content has been already encrypted and they can not obtain the plaintext of the content.

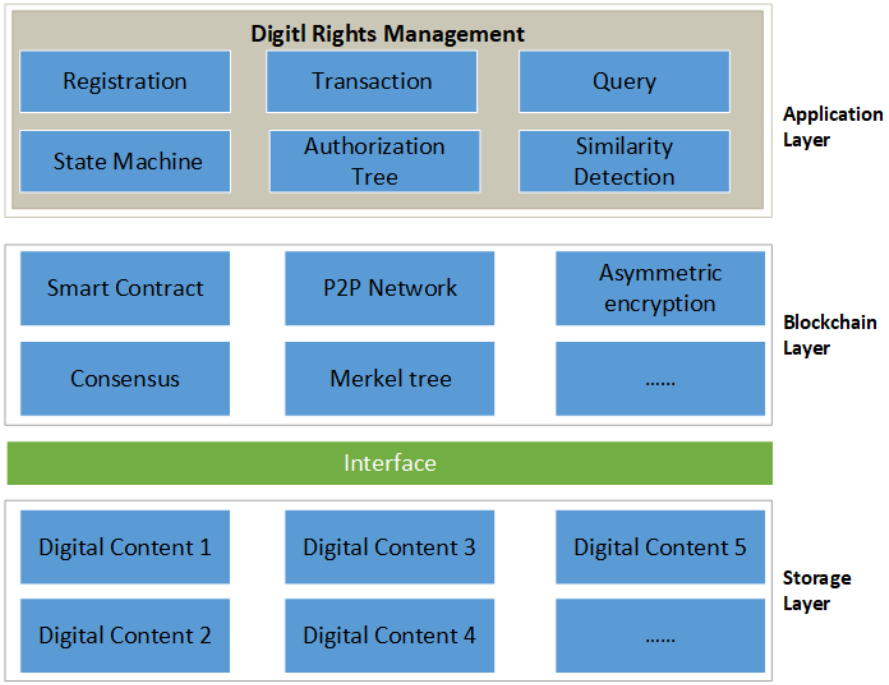

Fig. 2 Overview of BDRM architecture.

5. The Proposed Scheme

BDRM utilizes the three tiers to implement a complete DRM system. In BDRM, a user _rst registers the digital rights, then the key information is stored in the blockchain and the encrypted digital content is stored in the distributed file system IPFS. The first user who registers digital content successfully is granted the ownership of the digital content in BDRM. Meanwhile, an authorization tree will be generated once a new digital content being registered in BDRM. Further, any subsequent digital content registration will be deemed as invalid because of similarity detection fuzzy hash [13]. When a digital rights transaction is performed, a usable digital content watermark is embedded and digital content distribution is performed under the encryption domain. After that, updating the authorization tree and the transaction is recorded on the blockchain.

5.1 State Machine Management

BDRM is implemented by de_ning a state machine and rules for digital rights state transitions in the application layer. As shown in Fig.3, we assume that the number of the states is not less than five, which are "Publish", "Lock", "Cancel", "SellPermission" and "SellOwnership". "Publish" is the initial state after copyright successful registration. "Lock" is an invisible state, others can not pick up copyright information except the owner. If the owner does not want to continue to register copyright

on the platform, he can set the copyright state to "Cancel". The two state of "SellPermission" and "SellOwnership" are set to transfer rights expediently for the owner. "Transfer" only appears as an intermediate transition state during transaction.

Digital rights state machine runs through the entire process of BDRM. Every time a user makes a digital transaction, the corresponding state will be checked. Only those digital rights which are in the correct state can be traded. Moreover, one state can be changed to another pursuant to the rules. In this way, not only the user clearly knows his copyright situation, but also can he achieve the _ne-grained usage control of digital rights.

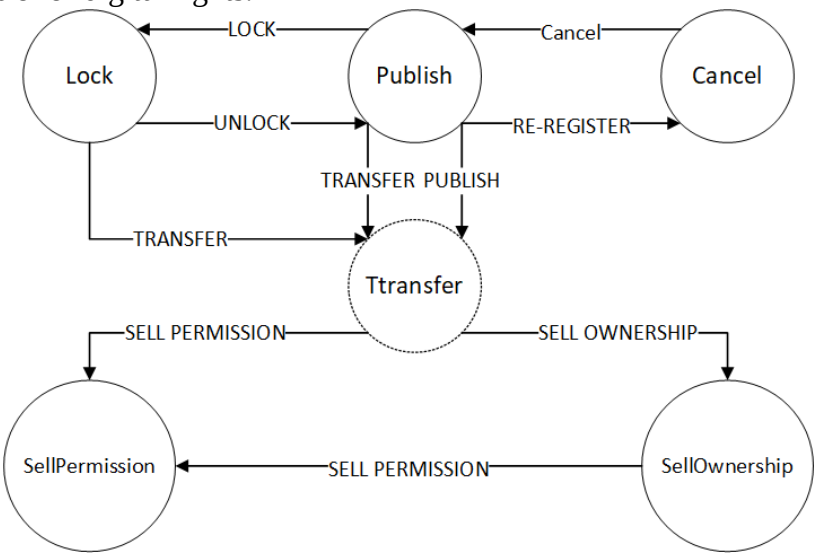

Fig. 3 Digital rights state machine

5.2 Authorization Tree

An authorization tree which is responsible for usage control model is defined for BDRM. Each node in the authorization tree corresponds to a user.

In the authorization tree, we define the security authorization policy with authorization attributes. An authorization attribute refers to unique user and a unique privilege. Authorization attribute is defined as follows: it is a pair *<subject; level>* where subject identifies a user and level takes its value from the following set: *{1, 2, ..., n}*. The data structure of the authorization tree node is as follows. The owner represents the subject and the "permissionList" represents a binary array of level.

```
struct Node{
  address owner;
  uint parent;
  uint[] childs;
  bool extendable;
  uint8[] permissionList;
  string startTime;
  string durations;
  uint8 status;
}
```

TreeNode Structure

Authorization attributes are associated with nodes. An authorization attribute can be set as different level, i.e., *{1, 2, ..., n}*. Attribution can be a combination of multiple privilege. For example, if the level of authorization attributes is equal to 1, the user of the node can use the digital content, but he can not copy the digital. The level of authorization attributes is equal to 2, the node of user can copy and use the digital content. In addition, if the user wants to authorize down, the level of authorization

attributes must be more than 3. The _ne-grained division of user's permission can achieve the purpose of fine-grained management permission. Only transactions that satisfy the permission conditions will be executed.

Fig.4 shows an example of the authorization tree. When a user registers in BDRM system, a root node will be generated for him. The user of root node has all the authorization attributes. Each user can purchase rights from the original owner if the state of the original owner is correct. For example, Alice want to purchase the usufruct and right of reproduction while Bob only purchases the usufruct. When Carol purchases the usufruct from Alice, the transaction will be executed successfully if the state and attribute of Alice's rights is correct. The ownership will be transferred after the new owner purchases the ownership from the original owner.

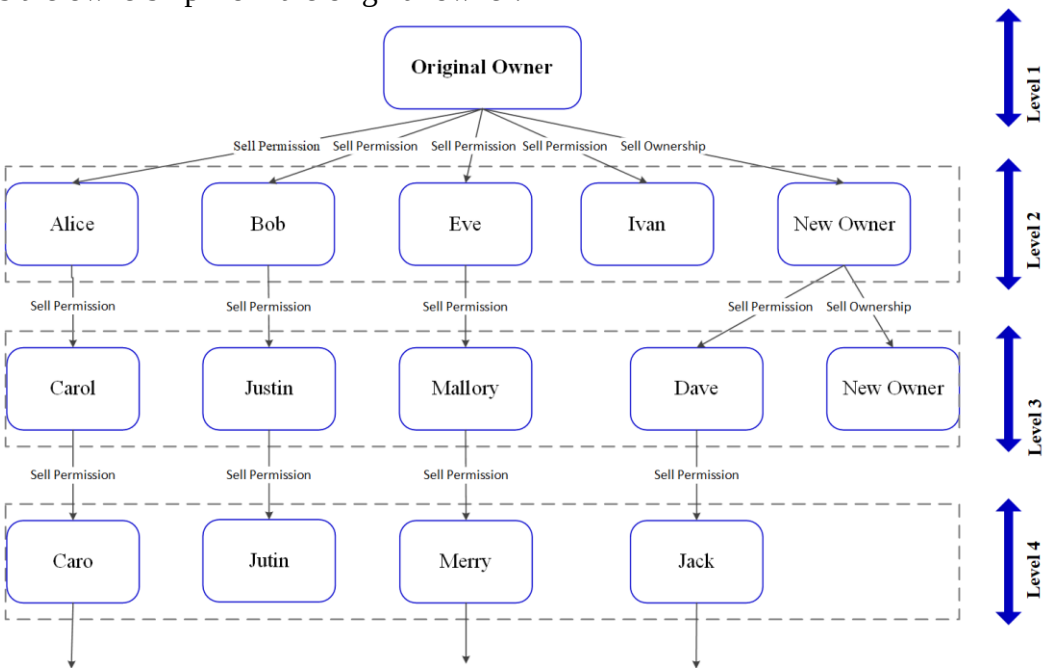

Fig. 4 Authorization tree

In general, we implement a _ne-grained usage control through combining state machine with authorization tree. Each step of the user's operation on the platform requires the state machine and authorization tree. Smart contract will automatically execute code when the conditions are met. Through smart contract, we guarantee the execution of transactions and the transfer of rights without TTP. Finally, the personal information of user will be converted to the address on the platform. Meanwhile, the nodes on the authorization tree can only know the number of their own subtree nodes. In this way, not only is it transparent for users to understand the process of copyright circulation, but also it will not lead to the leakage of user privacy.

5.3 Security Analysis

In this section, we analyze the security of our system.

Blockchain-based Platform Security: Parity is used to deploy consortium blockchain in the prototype system. This not only ensures that the system is decentralized, but also enables low-latency execution of transactions. Furthermore, the platform will not have a single point. Because we adopt the model of consortium blockchain, the limitation on "Gas Limit" can be removed. In the case of no communication delay, the calculation formula of the TPS is:

$$TPS = \frac{\text{NetWorkSpeed}}{(\text{NodeNum} - 1) \times \text{TransactionSize}} \quad (1)$$

Authoritative Tree-based Platform Digital Resource Security: The authorization tree implemented through smart contract guarantees the transparency of the authorization process. On the one hand, the illegal user can not appear in the authorization tree. On the other hand, expired users and the common

users who do not meet the authorization conditions are not allowed to trade and authorize. Usage control will be strictly limited in this way.

Dynamic Security of Rights State Machine: The state of the rights is tied to the user in the prototype system. The state can only be updated by the owner. Abnormal operation will be ignored. Users adjust the status of digital rights dynamically, which ensures the real-time and security of copyright and enhances the flexibility and security of the system application.

6. Performance Evaluation

The performance of the system is evaluated in this section. We implement the application layer with Java and Solidity. Five serves which installed Ubutun OS with single core processor (2GB RAM, 5M peak bandwidth, 40G disk) are used to deploy the consortium blockchain and Java service. We adopt PC which installed Deepin OS and Intel(R) i5 processor with 64 bit operating system and 4GB RAM to test the interfaces.

Fig.5 (a) shows the response time required for single user to register 1000 copyrights. The average time is 45.965 ms. It shows the response time required for a single user to register 50 copyright in multi-thread environment in Fig.5 (b). The average time is 47.895 ms. Similarly, we test the response time of registration in multi-users using multi-threads, as shown in Fig.5 (c) and the average time is 46.252 ms. Those results show that the copyright registration function can be quickly completed on the system. However, the speed of transaction in blockchain affects the response time of copyright registration.

Fig. 5 The response time of registration function

The response time required for the function of purchasing copyright is shown in Fig.6. In short, the average time of purchasing copyright is 50.012 ms. The average time for single user to purchase copyright, single user to purchase copyright in multi-threaded environment and multiple users to purchase copyright in multi-threaded environment are 51.157 ms, 48.538 ms and 50.825 ms.

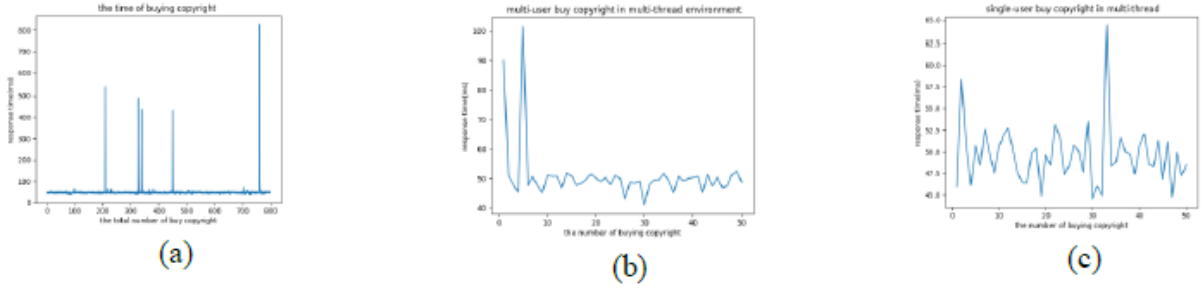

Fig. 6 The response time of purchasing copyright function

When single user operates in a single thread, the response time is relatively smooth, as shown in Fig.5 (a) and Fig.6 (a). Sometimes the response time is even more than 300 ms in this case. The cause of the peak may be the delay of the network. But the most fundamental reason is that single thread can't process the request in time. In the case of multi-thread, the server needs to process multiple requests at the same time, so the response time will have a large uctuation during the processing, as shown in Fig.5 (b), Fig.5(c), Fig.6 (b) and Fig.6 (c). The overall situation of response time for multi-user and single user in multi-thread is similar, because there are always threads active.

In order to prevent mishandled exceptions raised in [16], the non-recursive algorithm is chosen to implement the traversal function of the authorization tree. Before adding a node to the authorization tree, it is necessary _nd the parent of the node. Thus, the time complexity of adding a node to the authorization tree is O(n).

7. Conclusion

In this paper, we describe the challenges of the current digital rights management system in this paper. To solve above challenges, we design the digital rights state machine and authorization tree in smart contract which the owners control their own copyright more _nely and track the circulation of digital rights through them. Finally, we analyze system security and performance. The implemented system will be commercialized. One defect of our work may be only applicable to copyright registration of the single owner, but the problem of how to negotiate the registration of multiple owners still exists. We will work to solve or weaken this problem in our future work.

References

[1] Digital rights management. https://en.wikipedia.org/wiki/Digital_rights_management.
[2] Mustafa Al-Bassam. Scpki: A smart contract-based pki and identity system. In Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts, pages 35-40. ACM, 2017.
[3] Muneeb Ali, Jude C Nelson, Ryan Shea, and Michael J Freedman. Blockstack: A global naming and storage system secured by blockchains. In USENIX Annual Technical Conference, pages 181-194, 2016
[4] Alapan Arnab and Andrew Hutchison. Digital rights management-an overview of current challenges and solutions. In Proceedings of Information Security South Africa (ISSA) Conference, volume 2004. Citeseer, 2004.
[5] Juan Benet. Ipfs-content addressed, versioned, p2p file system. arXiv preprint arXiv:1407.3561, 2014.
[6] Vitalik Buterin. Ethereum: A next-generation smart contract and decentralized application platform, 2013. http://ethereum. org/ethereum.html, 2017.
[7] Tien Tuan Anh Dinh, Ji Wang, Gang Chen, Rui Liu, Beng Chin Ooi, and Kian-Lee Tan. Blockbench: A framework for analyzing private blockchains. In Proceedings of the 2017 ACM International Conference on Management of Data, pages 1085-1100. ACM, 2017.
[8] Ethcore. Parity Next Generation Ethereum Browser. https://www.parity.io.
[9] Shigeru Fujimura, Hiroki Watanabe, Atsushi Nakadaira, Tomokazu Yamada, Akihito Akutsu, and Jay Junichi Kishigami. Bright: A concept for a decentralized rights management system based on blockchain. In Consumer Electronics-Berlin (ICCE-Berlin), 2015 IEEE 5th International Conference on, pages 345-346. IEEE, 2015.
[10] Hyperledger. Blockchain technologies for business. https://www.hyperledger.org/.
[11] Stefan Katzenbeisser, Aweke Lemma, Mehmet Utku Celik, Michiel van derVeen, and Martijn Maas. A buyer{seller watermarking protocol based on secure embedding. IEEE Transactions on Information Forensics and Security, 3(4):783{786, 2008.
[12] Junichi Kishigami, Shigeru Fujimura, HirokiWatanabe, Atsushi Nakadaira, and Akihiko Akutsu. The blockchain-based digital content distribution system. In Big Data and Cloud Computing (BDCloud), 2015 IEEE Fifth International Conference on, pages 187-190. IEEE, 2015.
[13] Jesse Kornblum. Identifying almost identical les using context triggered piecewise hashing. Digital investigation, 3:91-97, 2006.
[14] Ming Li, Jian Weng, Anjia Yang, Wei Lu, Yue Zhang, Lin Hou, Liu JiaNan, Yang Xiang, and Robert Deng. Crowdbc: A blockchain-based decentralized framework for crowdsourcing. IEEE Transactions on Parallel and Distributed Systems, 2018.
[15] Qiong Liu, Reihaneh Safavi-Naini, and Nicholas Paul Sheppard. Digital rights management for content distribution. In Proceedings of the Australasian information security workshop

conference on ACSW frontiers 2003-Volume 21, pages 49{58. Australian Computer Society, Inc., 2003.
[16] Loi Luu, Duc-Hiep Chu, Hrishi Olickel, Prateek Saxena, and Aquinas Hobor. Making smart contracts smarter. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pages 254-269. ACM, 2016.
[17] Zhaofeng Ma, Ming Jiang, Hongmin Gao, and Zhen Wang. Blockchain for digital rights management. Future Generation Computer Systems, 89:746-764, 2018.
[18] Nasir Memon and Ping Wah Wong. A buyer-seller watermarking protocol.IEEE Transactions on image processing, 10(4):643-649, 2001.
[19] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. Consulted, 2008.
[20] Jaehong Park, Ravi Sandhu, and James Schifalacqua. Security architectures for controlled digital information dissemination. In Computer Security Applications, 2000. ACSAC'00. 16th Annual Conference, pages 224-233.IEEE, 2000.
[21] POA. Proof-of-authority. https://poa.network.
[22] Alexander Pretschner, Manuel Hilty, and David Basin. Distributed usage control. Communications of the ACM, 49(9):39-44, 2006.
[23] Pamela Samuelson. A copyright challenge to resales of digital music. Communications of the ACM, 56:25, 2013.
[24] Hiroki Watanabe, Shigeru Fujimura, Atsushi Nakadaira, Yasuhiko Miyazaki, Akihito Akutsu, and Jay Kishigami. Blockchain contract: Securing a blockchain applied to smart contracts. In Consumer Electronics (ICCE), 2016 IEEE International Conference on, pages 467{468. IEEE, 2016.
[25] Ruzhi Xu, Lu Zhang, Huawei Zhao, and Yun Peng. Design of network medias digital rights management scheme based on blockchain technology. In Autonomous Decentralized System (ISADS), 2017 IEEE 13th International Symposium on, pages 128-133. IEEE, 2017.
[26] Li Zhang. Annual report on digital publishing industry in china: 2017-2018. 2018.
[27] Zehao Zhang and Li Zhao. A design of digital rights management mechanism based on blockchain technology. In International Conference on Blockchain, pages 32{46. Springer, 2018.
[28] Zhiyong Zhang. Digital rights management ecosystem: Open issues and challenges. International Journal of Digital Content Technology and its Applications, 5(11), 2011.
[29] Zhiyong Zhang, Tao Huang, Danmei Niu, and Lili Zhang. Usage control model for digital rights management in digital home networks. Journal of Multimedia, 6(4), 2011.