Scientific Research Publishing

Communications and Network, 2022, 14, 45-67
https://www.scirp.org/journal/cn
ISSN Online: 1947-3826
ISSN Print: 1949-2421

HBRO: A Registration Oracle Scheme for Digital Rights Management Based on Heterogeneous Blockchains

Ri Ouyang

College of Information Science and Technology, Jinan University, Guangzhou, China
Email: gorocky126@126.com

How to cite this paper: Ouyang, R. (2022) HBRO: A Registration Oracle Scheme for Digital Rights Management Based on Heterogeneous Blockchains. *Communications and Network*, **14**, 45-67.
https://doi.org/10.4236/cn.2022.141005

Received: December 13, 2021
Accepted: February 13, 2022
Published: February 16, 2022

Abstract

As the Internet enters the era of big data, massive amounts of data are flooding people's mobile phones and computers. The emerging self-media industry produces many videos every day, which also exposes many security issues in digital rights management (DRM). The works of original creators can easily be infringed on public networks, and it is urgent to protect the copyright of digital content. Traditional digital rights management (TDRM) has many problems, including unqualified copyright confirmation, difficulty obtaining evidence, long time-consuming, high price, and high centralization. The combination of blockchain technology and digital rights management is one of the most popular blockchain application scenarios, the characteristics of the blockchain match DRM market demand. This paper proposes a registration oracle scheme for digital rights management based on heterogeneous blockchains, HBRO, which uses review and voting as a means to judge whether a work can be registered for copyright. HBRO is more rigorous than TDRM and DDRM in the copyright confirmation stage, rejecting many unnecessary and unqualified contents. In addition, a secure cross-chain solution is used to ensure the integrity and correctness of data transmission on heterogeneous blockchains.

Keywords

Blockchain, DRM, Cross-Chain, Blockchain Oracle, Smart Contract

1. Introduction

Digital Rights Management (DRM) [1] is an integral part of social development. With the rapid development of digital media, many short videos, audio, and

pictures are disseminated on the Internet, frequently leading to the unauthorized use of copyright content. According to statistics released by CNNIC (China Internet Network Information Center), as of December 2020, the scale of Chinese netizens is around 989 million, which makes up the world's largest digital society, short videos are popular among nearly 90% of netizens, becoming the second-largest network application after instant messaging. However, the faster the development, the more challenges there will be. Due to the ease of copying content and the lack of copyright awareness of the public, piracy cases have occurred repeatedly, and copyright disputes are often not well resolved. It is hard to distinguish who is the actual originator. At the same time, the spread of pirated digital content dampened the enthusiasm of innovators, leading to a decrease in quality.

Traditional digital rights management (TDRM) [2] faces many problems, such as unqualified copyright confirmation, difficulty obtaining evidence, long-time cost, high price. Worst of all, TDRM is centralized and public, and all TDRM details are stored in a central organization. Once the internal data is tampered or lost, copyright protection will be threatened, eventually causing ambiguity in content ownership. Blockchain technology is currently a feasible solution for decentralized digital rights management (DDRM). Blockchain is a decentralized peer-to-peer network, and its characters perfectly fit copyright protection.

Furthermore, considering the transmission of massive data, it is inefficient to register the copyright for each content, and transactions on the blockchain are challenging to manipulate. Blockchain can ensure that nodes worldwide can participate in the same network without time or space limitations. The originator does not have to pay a high fee for storing his/her content on the blockchain to guarantee the permanent storage and immutability of the content.

Monegraph [3] is a DDRM system that focuses on copyright transactions; it uses blockchain to register the copyright of the content and record every transaction that occurred on blockchain. NFT (Non-Fungible Token) [4] provides a method to mark the ownership of natively digital assets so that each content is guaranteed to be unique and cannot be copied. Mediachain [5] uses blockchain and the Interplanetary File System (IPFS) for DRM, allowing originators to attach information (such as timestamps) to their content and putting the content information on the Bitcoin blockchain. Verisart [6], Binded [7] are also current DDRM applications. In China, companies such as Alibaba and Huawei have begun to establish the DDRM platform, and they also use the underlying technology of the blockchain to register, store, and protect the copyrights of the content. The above DDRM system spent a shorter time than TDRM, and the evidence solidification is more convincing than TDRM. Unfortunately, their models are not decentralized and public enough to register the copyrights; same as the previous model, they are still entirely dependent on the central organization and do not have a fundamental revolution.

This paper proposes a registration oracle scheme for DRM based on hetero-

geneous blockchains (HBRO), which guarantees copyright registration more efficiently and improves the content quality. HBRO ensures rigorous confirmation of copyrights through the proof record and copyright registration. The whole process uses an oracle and multi-smart contracts to ensure the automation of the copyright confirmation. Besides, HBRO utilizes the notary scheme to ensure the security of data transmission between heterogeneous blockchains. Compared with TDRM and DDRM, HBRO has a more efficient copyright confirmation scheme, which reduces the duplication of copyright confirmation from the root cause, reduces the incidence of piracy, and makes the entire copyright ecology more healthy and complete.

The main contribution of HBRO can be divided into the following three points.

- Set up a registration method for the digital content of the review and voting. The review process is the main task, voting process as a supplement to improve the accuracy of the review.
- Introduce a cross-chain solution [8] to improve the interoperability of heterogeneous blockchains, expand application scenarios in DRM, and combine the oracle with DDRM to find greater possibilities for the blockchain in DRM.
- Propose HBRO, which applies blockchain to digital copyright, optimizing the process of copyright confirmation and achieving a balance between centralization and decentralization.

2. Background

2.1. Blockchain Technology

After the advent of Bitcoin [9], blockchain brought vitality to the traditional financial field and led to new changes in cryptocurrency. Since Bitcoin went online, it has been operating without interruption on a global scale for more than 12 years and has successfully processed tens of millions of transactions with no significant failures, and beyond that, the application of blockchain technology is far more than cryptocurrency. Increasing researchers have begun to explore the possibilities of blockchain technology in different fields such as the settlement of financial asset transactions, Internet of Things [10], and digital rights management. There are many reasons why blockchain is potential in these fields: First, with the rapid development of the distributed Internet, centralized entities control almost all users' private information, severely threatening privacy security. The value and characteristics embodied by blockchain perfectly fit the decentralization concept. Blockchain can be regarded as a distributed ledger, which does not require central entities to maintain the ledger data; thus, it can avoid the single point of failure (SPOF) in centralized systems. Second, transactions on the blockchain can be tamper-resistant. The block stored on the blockchain means that its transaction has reached a consensus, tampering with data on a single node will not affect the state of the entire system. Third, The blockchain does not require the existence of a centralized entity, which dramatically reduces

maintenance costs without sacrificing data security. Fourth, The blockchain can be completely transparent; users can easily trace the transaction history on the blockchain. In short, the blockchain is a distributed network that can provide more possibilities to explore the future direction of the Internet.

According to different application scenarios, the blockchain can be divided into three types: public blockchain, consortium blockchain, and private blockchain. The public blockchain is permissionless and is open to all nodes, such as Bitcoin, Ethereum [11], etc. The consortium and private blockchain are permissioned, mainly for enterprise and individual users, such as Hyperledger Fabric [12], EEA, etc.

2.2. Smart Contract

The smart contract is a program that runs on the blockchain, converting the complex logic into the smart contract that contains the code and the corresponding state data. Nick Szabo [13] proposed the concept of smart contracts in 1995, which is to write legal provisions into executable code. Due to the lagging nature of technology, smart contracts were not applied to the Internet industry in the 1990s. Since the birth of Bitcoin, people have found that smart contracts can run in the trusted execution environment of t—he underlying blockchain technology of Bitcoin. Vitalik Buterin first gained insight into the suitability between smart contracts and the blockchain and created Ethereum. Smart contracts are like a trusted party that can temporarily keep assets and automatically perform operations by pre-defined rules. Many other blockchain platforms support smart contracts. Hyperledger Fabric supports multiple languages to write smart contracts, such as Golang, Java, and JavaScript. Quorum [14] and Wanchain [15] are both branches of Ethereum, and they support solidity language to write smart contract code like Ethereum.

2.3. Interoperability and Oracle

Interoperability [16] [17] is a combination of interaction, operation, and ability; it is defined by IEEE (Institute of Electrical & Electronic Engineers) in 1990 as the ability to exchange information between two or more systems or components and to use the exchanged information. Interoperability is widely recognized as a critical strategy of the widespread application of the blockchain; the current underlying technologies of blockchain applications in different fields are very different in terms of data structure, such as consensus mechanism, communication protocol, limiting the large-scale application of blockchain. There is no unified standard that can connect every blockchain platform. Eventually, the blockchain platforms are separated, forming "Blockchain Island" [18]. At present, the main constraints of blockchain are reflected in three aspects: 1) it is challenging to cross-chain intercommunicate between different blockchain systems; 2) it is difficult for different underlying blockchains to switch smoothly; 3) the uncertain credibility of the data interaction between heterogeneous block-

chains. The interoperability could be divided into application layer interoperability, interchain interoperability, and on-chain and off-chain interoperability.

Vitalik Buterin, the founder of Ethereum, believes that cross-chain is a manifestation of interoperability and proposed three cross-chain schemes [19]: notary scheme, sidechain (or relay chains), and hashed time-lock techniques. The notary scheme is the most straightforward cross-chain scheme, using one or a group of trusted entities to monitor and automatically execute on-chain events actively or passively send signed messages. The sidechain makes the interoperability more immediately and completes the information exchange of different blockchains automatically without relying on the assistance of an intermediary. Hash time-lock is essentially a smart contract that combines hash and time lock to support cross-chain atomic operations. The Atomic Swaps Protocol [20] proposed the exchange of assets between heterogeneous blockchains atomically.

Oracle [21] is a form of blockchain interoperability. Due to the built-in consensus mechanism, blockchain cannot actively obtain external data from off-chain data sources. For the current blockchain systems, external data is the primary prerequisite for the system's operation, e.g., the score of the just-concluded NBA Finals can be obtained on betting Dapp. Oracle is like a bridge connecting the blockchain world and the external world.

As shown in **Figure 1**, the oracle receives the request sent by the account on the blockchain and requests the specified data from external data sources, then sends the returned data to the blockchain. The current oracle platforms are in full swing, such as Oraclize [22], Chainlink [23], Town Crier [24], aeternity [25], Augur [26], etc. This paper proposes a cross-chain solution using oracle, which aims to ensure the copyright registration process's high efficiency and low cost through review on the permissioned blockchain and voting on the permissionless blockchain. Most importantly, the proposed scheme achieves a balance between centralized and decentralized. The authenticity of the data is guaranteed by cryptography, the permissioned blockchain verifies the data correctness, and the democracy of the permissionless blockchain is guaranteed.

2.4. IPFS (InterPlanetary File System)

IPFS [27] is a distributed peer-to-peer protocol to store and share files, connecting

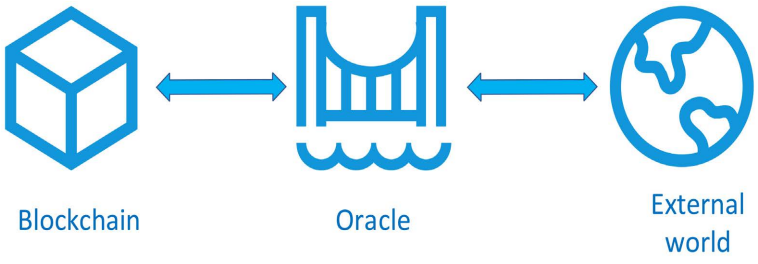

Figure 1. Oracle overview.

all computing devices, it is a content-addressed network with high throughput and does not require a centralized database to store data, so there is no SPOF (single point of failure) that occurs. IPFS aims to replace HTTP, and the central IPFS principle is to treat all data as part of the same Merkle DAG. IPFS integrates the great ideas of the previous peer-to-peer system, including DHTs, Git, BitTorrent, and SFS. Each node is unprivileged; the node stores files on the local device and different nodes can connect and transfer files.

3. Related Work

Pic-Chain is a DRM service based on blockchain, it uses a self-developed blockchain copyright registration network to generate copyright DNA for each picture, with the timestamps.

BQJ is a DRM system based on the FISCO BCOS blockchain platform [28], uses the consensus algorithm PBFT [29] to provide instant and fast proof for massive data and builds a business system centered on copyright protection services on the blockchain.

Zhigui Technology develops enterprise-level copyright solutions and provides related services such as data monitoring and protection of legal rights. Z-ledger is the underlying blockchain platform expanded by Zhigui Technology based on the Hyperledger Fabric, which has the advantages of high performance, high security, and scalability.

Zhaofeng Ma [30] *et al.* proposed a blockchain-based trust model chain DRM-Chain for digital rights management and established architecture of flexible external storage and internal block creation, providing high-level credible content protection and conditional traceability of violation content service by using a digital watermark.

Chinese Copyright Blockchain [31] is a DRM platform based on ChainMaker [32] which is a newly open-source blockchain platform, creating a consortium blockchain for DRM that is industry-wide, multi-organizational, and multi-sectoral.

Zhang [33] proposed a decentralized DRM model of data to solve the current copyright management problems, which is the system that focuses on the geographic data generated by scientific research papers, ensuring the sharing of spatial data with other geographic researchers and protecting the copyright of papers' data. In addition, Zhang proposes to store data in IPFS to avoid a single point of failure of data storage and designs three spatial digital rights management algorithms to perform registration, query, and application operations. However, this scheme ignores another problem brought about by IPFS: data security. IPFS is addressed by the content hash value, so any node with the content hash can copy the data and easily disrupt the copyright market.

ASTRAEA [34] is a voting-based blockchain oracle that returns the consensus results of voters and certifiers to the submitters through an incentive mechanism and achieves a Nash Equilibrium under certain rules. Through the reward and

punishment mechanism, honest players can be greatly encouraged to participate. Voters play a low-risk and low-reward role to resist the adversarial attack, while certifiers play a high-risk and high-reward role to ensure the accuracy of the result. The same consensus result between the voting and certifying means the final result.

Shintaku [35] is an end-to-end distributed blockchain oracle, its design idea is based on ASTRAEA, and it has been improved on the ASTRAEA scheme to solve the verifier's dilemma case.

Chainlink is one of the widest blockchain oracle network services currently, it can request multiple data sources and multiple oracles to connect the on-chain and off-chain, improving the data quality of the entire system. Besides, Chainlink requests the voting stage based on reputation, which ensures the correctness of external data.

4. System Model

4.1. Architecture

The system architecture is mainly divided into five layers: data layer, consensus layer, incentive layer, contract interface layer, and user layer. As shown in **Figure 2**, the core layers of the system are the contract interface layer and the incentive layer. In addition to smart contracts and reward and punishment mechanism, these two layers also provide a notary scheme that guarantees security and integrity of data transmission between heterogeneous blockchains.

The user gets a content hash that can be addressed and requests oracle for copyright confirmation on the permissionless blockchain by storing the preprocessed content on IPFS. The process is divided into two stages: proof record and copyright registration: the proof record ensures that the user is the actual originator, avoiding the repeated registration of similar content and reducing unnecessary copyright disputes. Copyright registration is completed through copyright review and copyright voting, copyright review is conducted on the permissioned blockchain and completed by professional reviewers, and screened voters cast copyright voting on the permissionless blockchain. Oracle smart contract combines the two results to get the final result and returns it to the requester. As for the content storage, we store the metadata and related information of the content on IPFS and store the CID returned by IPFS on the blockchain, which can greatly storage weaken the shackles of blockchain storage.

4.2. System Roles

The system roles are mainly divided into four types: user, Pre-Leader, reviewer, and voter. The user and the voter are nodes of the permissionless blockchain. The reviewer works on the permissioned blockchain, and the Pre-Leader is responsible for cross-chain data transmission.

Users: Users are divided into three groups: content owner, miner, and content consumer. The content owner stores their content on IPFS and requests oracle

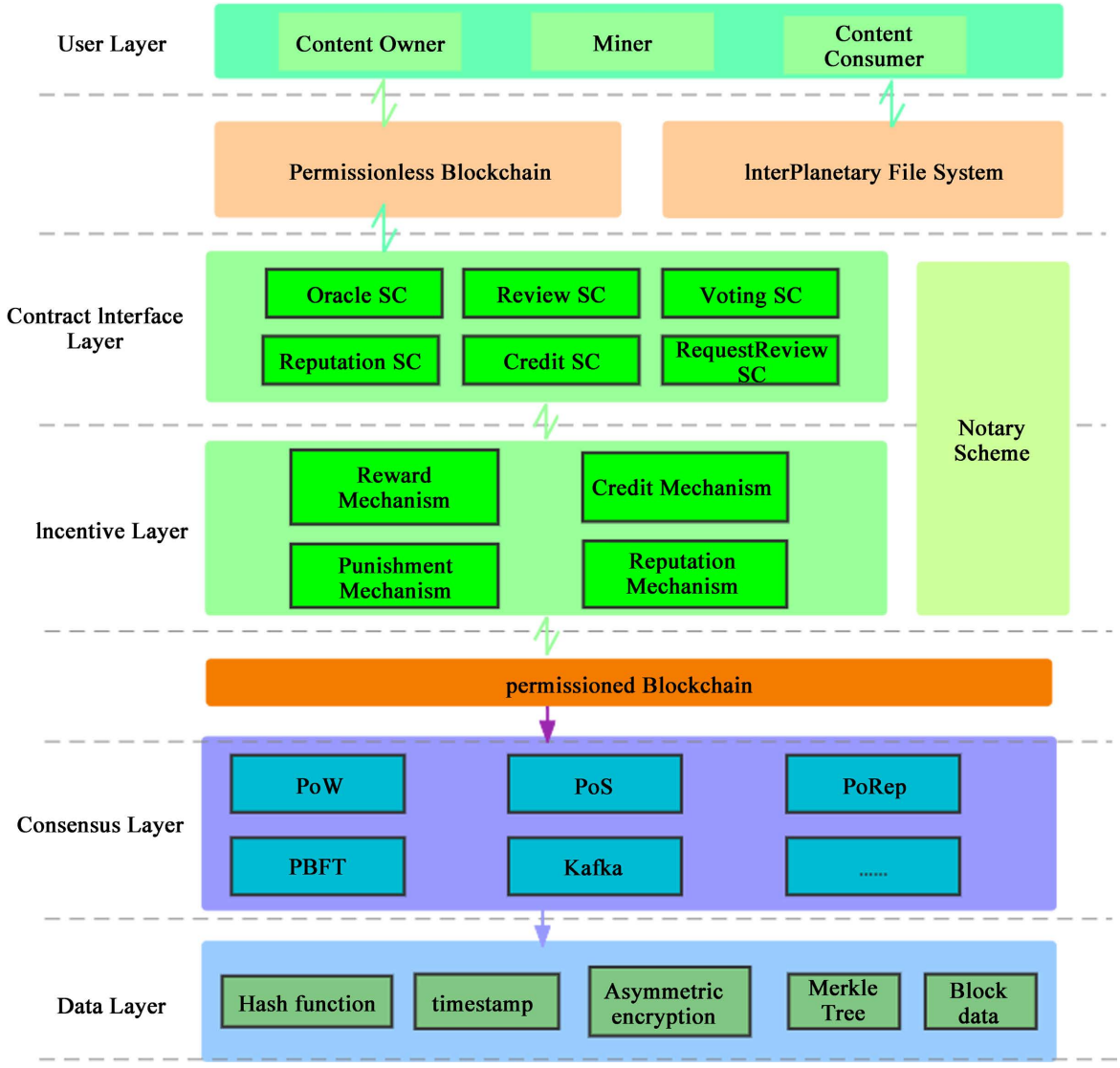

Figure 2. The architecture of the proposed system.

to register the copyright. When copyright registration is successful, the related proof is sent to the miner on the blockchain. The Miner is responsible for putting the content owner's content information into a new legal block and permanently storing it on the copyright blockchain. Content consumers can purchase any authorized content or product on the copyright blockchain.

Pre-Leader: The pre-leader is in charge of monitoring the review request from the permissionless blockchain. When a user requests a copyright review, the system will select the Leader from the Pre-Leader group. The Leader acts as a notary and is responsible for the data interaction between the permissionless blockchain and the permissioned blockchain, including the copyright review process and reward and punishment distribution.

Reviewer: The reviewer is to the proposed system like the judge is to the court, they are professional in the copyright field and passively review the content information sent by the Leader. The reviewer should pledge credits (credit is

equivalent to the cryptocurrency of the permissioned blockchain) as a deposit to participate in the copyright review.

Voter: nodes on the permissionless also need to send a deposit to obtain the vote right, then higher-reputation nodes will be prioritized to be voters in that round. After voting, voters will get the corresponding reward and their deposit back if their results are deemed to be correct, otherwise, they will lose their deposit and obtain nothing. At the same time, the system will reevaluate the reputation of every voter according to their performance in this round.

4.3. System Assumption

We assume that the honest nodes in the proposed system account for the majority. The blockchain is a decentralized ledger. If the malicious nodes are more than the honest nodes, the ledger may be maliciously manipulated, and the system faces security vulnerabilities. We assume that the results have only one truth value (T or F, where T stands for Pass and F stands for Fail) for each content c. Whether it is a review result or a vote result, there is only a T or F result. As shown in Formula (1), we assume that the vote results are independent of each other and that every voter's probability T or probability F is the same, which is subject to the Bernoulli distribution.

$$\text{Result}(c)=\begin{cases}\text{T}(\text{PASS}) & \text{with the probability of } q\\ \text{F}(\text{FAIL}) & \text{with the probability of } (1-q)\end{cases} \quad q\in[0,1] \qquad (1)$$

For different voters, their results can only be one of T or F. For different reviewers, their results do not affect each other; unlike voters, their results are an integer from 1 to 4 points.

5. HBRO: A Registration Oracle Scheme for Digital Rights Management Based on Heterogeneous Blockchains

This section combines oracle with copyright protections to propose HBRO, a registration oracle for DRM based on heterogeneous blockchains. HBRO applies the review and voting mechanism to ensure the rigour of copyright confirmation. The copyright confirmation process consists of two phases: the proof record and the copyright registration. The proof record phase prevents the pretender from impersonating the content owner, and the copyright registration phase ensures the mapping relationship between the owner and the content. HBRO has two heterogeneous blockchains, one is the permissionless copyright blockchain, where anyone can access the data on the blockchain, and the copyright blockchain is mainly responsible for interaction with users and copyright transactions. The other is the copyright review blockchain, a permissioned blockchain specially used for the copyright review process, in which the nodes are all internal members of the copyright entity.

5.1. Content Pre-Processing

Content pre-processing could prevent voters or reviewers from maliciously dis-

closing the full content during the copyright confirmation process. Content owners pre-process the content to protect the confidentiality of the content from being public before he/she gets the copyright.

Digital watermarking technology can track and monitor productions, reducing the occurrence of infringements and accelerating the efficiency of judicial proceedings. Digital watermarking technology uses signal processing methods to embed hidden watermarks in digital multimedia data, which can verify the copyright owner of digital productions, identify sellers, purchasers, or provide other additional information about the content of digital productions. Information is embedded in digital images or video sequences in a form invisible and used to confirm the ownership of digital productions and track infringements. Content owners can add digital watermarks to the content or use an abridged version of the content, pc represents the processed content, and fc represents the full content without any modification.

The content owner must prepare the symmetric key K to encrypt fc to get the ciphertext Y and ensure that the content stored in IPFS is not fully disclosed. Even if other users get your content, they cannot view fc without the key K. After the copyright is confirmed, the symmetric key K of the full content can be released to copyright consumers. Encryption and decryption of content is as in Formula (2):

$$\begin{aligned} Y &= E(K, fc) \\ fc &= D(K, Y) \end{aligned} \tag{2}$$

5.2. Proof Record

As shown in Figure 3, the content owner stores the encrypted full content on IPFS and records the content hash value ($IPFS_{fc}$), then he packs the preprocessed

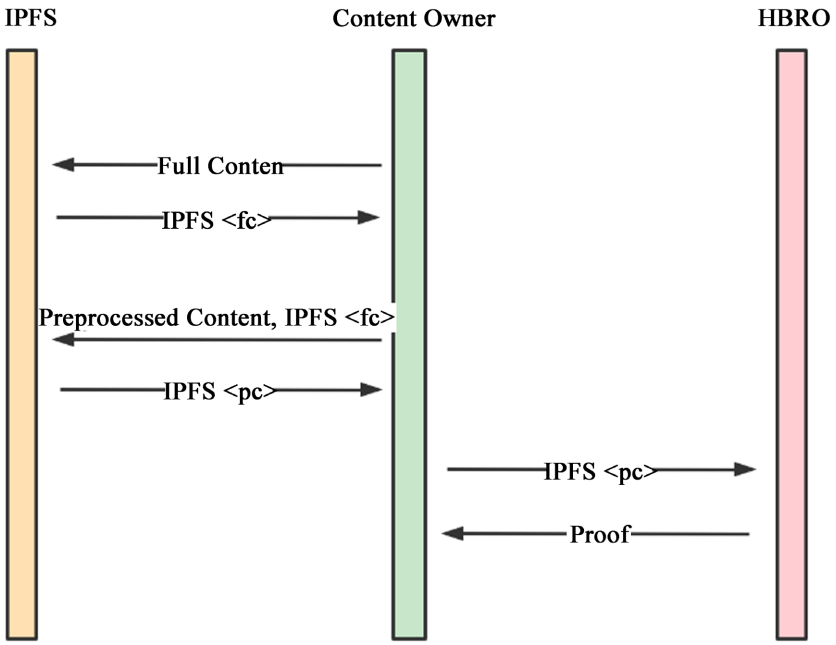

Figure 3. The flow of the proof record.

content and $IPFS_{fc}$ together and stores it in the IPFS, and saves the content hash value $IPFS_{pc}$ locally. The owner requests the HBRO for the proof record to tell other nodes that this content belongs to the owner, HBRO will store a global mapping table to save the mapping relationship from $IPFS_{pc}$ to the owner address. This process is permanently stored on the blockchain and is open and transparent. For some contents that do not meet copyright registration requirements, such as some secondary creations, users can protect the right of authorship through the proof-record procedure.

5.3. Copyright Registration

After finishing the proof record, the content owner can start the copyright registration. This phase includes the request to HBRO, the review process, the voting process, reward and punishment distribution. The notary scheme is used on the copyright blockchain and the copyright review blockchain. HBRO utilizes the notary scheme as a cross-chain solution for data transmission to ensure the integrity and correctness of the data. The combination of review and voting can significantly reduce the rate of copyright confirmation of similar content and optimize the copyright ecology. In addition, the reward and punishment mechanism can improve the performance and honesty of reviewers and voters about the content. After HBRO receives the owner's copyright registration request, it asks the copyright review blockchain reviewer to review and asks the voters on the copyright blockchain to vote. Figure 4 shows the general framework of HBRO.

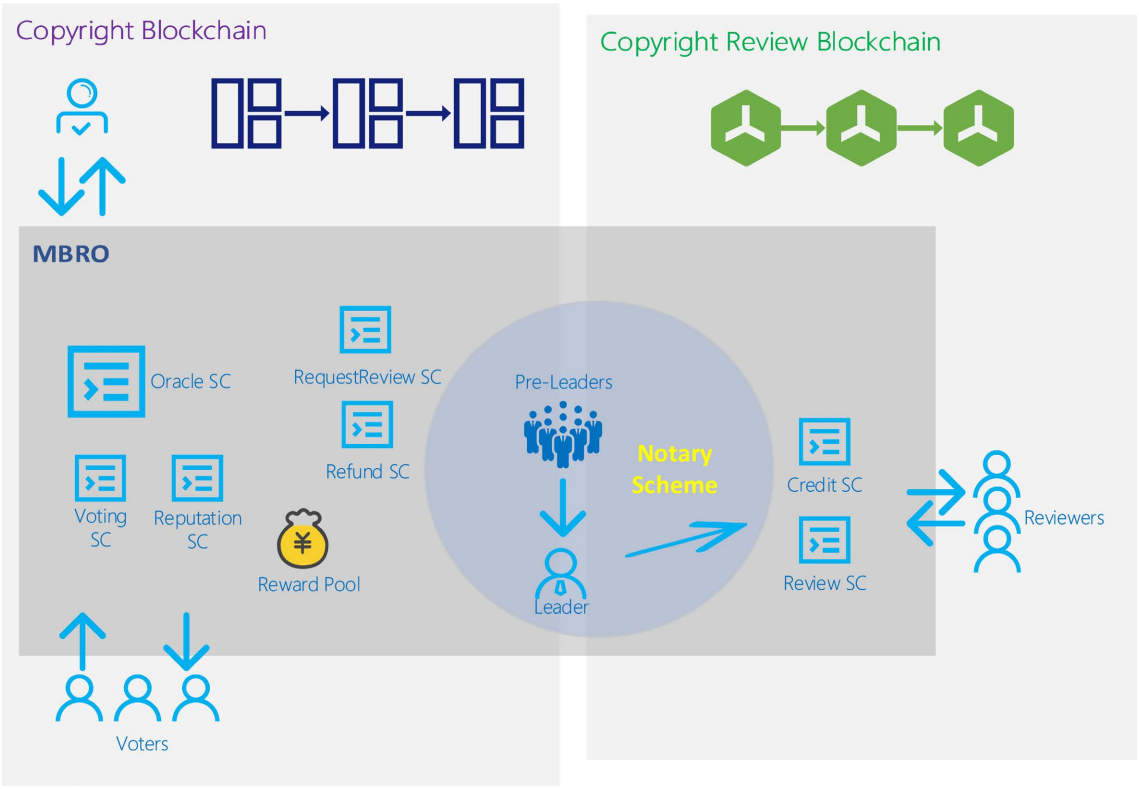

Figure 4. The general framework of HBRO.

5.3.1. Request HBRO

The content owner requests HBRO to register the copyright of the content and send <*OwnerAddr, Name, IPFS*$_{pc}$, *Token, Proof*> to *Oracle SC*. *OwnerAddr* denotes the copyright blockchain address of the content owner, *Name* denotes the name of the content, *IPFS*$_{pc}$ is the content hash of processed content stored on IPFS in advance, and the *pc* can be queried according to the hash value. *Token* is the cost required by the owner to register a copyright. Part of *Token* is used as a service fee for copyright registration, and the other part is used as a bonus for review and voting. (Figure 5)

5.3.2. Review Process

Figure 6 describes the review process flow. The notary scheme completes the data interaction of heterogeneous blockchains in the review process. Pre-Leaders act as cross-chain notaries; they listen to the review request on the copyright blockchain. After the *RequestReview SC* is triggered, the copyright review blockchain will select a Leader responsible for the content review. The Leader collects content information <*OwnerAddr, Name, IPFS*$_{pc}$, *Proof*> on the copyright blockchain and initiates a round of review by calling the *Review SC* on the copyright review blockchain. The reviewers can find the content owner's processing on IPFS through *IPFS*$_{pc}$ and pledge their deposits (credits on the copyright review blockchain) if they want to participate in this review. After the review is completed, the individual results of the reviewers will be returned to *Review SC*; then, the Leader calculates the Review Result and returns it to the *Oracle SC* on the copyright blockchain. The Leader will call the *Credit SC* to generate equivalent credits and distribute the credits to the correct reviewers as rewards.

Leader Election: Assuming *N* pre-leaders in the current organization, each pre-leader has its id. After listening to the RequestReview SC being invoked, the Leader will be selected according to his/her id.

The Leader calls the *Review SC* to request the reviewers to review the content on the copyright review blockchain. Reviewers review the content through *IPFS*$_{pc}$ search in IPFS and then send a deposit (credits on the copyright review blockchain) to *Review SC* for the right of review. $w_{i,c}$ denotes the weight of the *i-th* reviewer in the process of reviewing the content *c*. $d_{i,c}$ represents the deposit

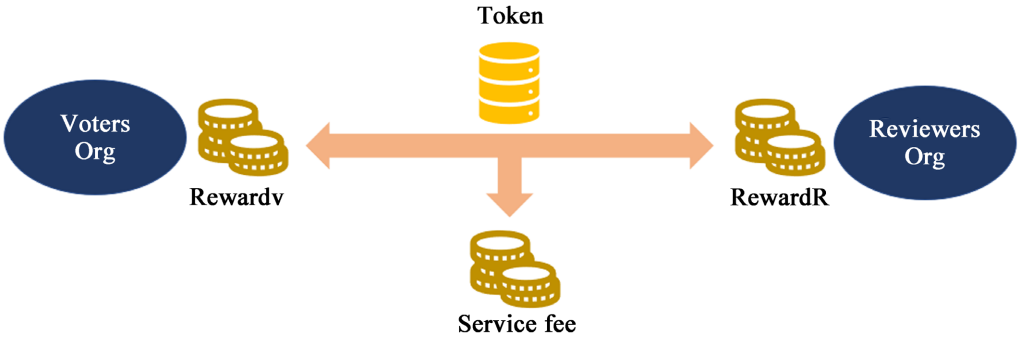

Figure 5. The distribution of Token.

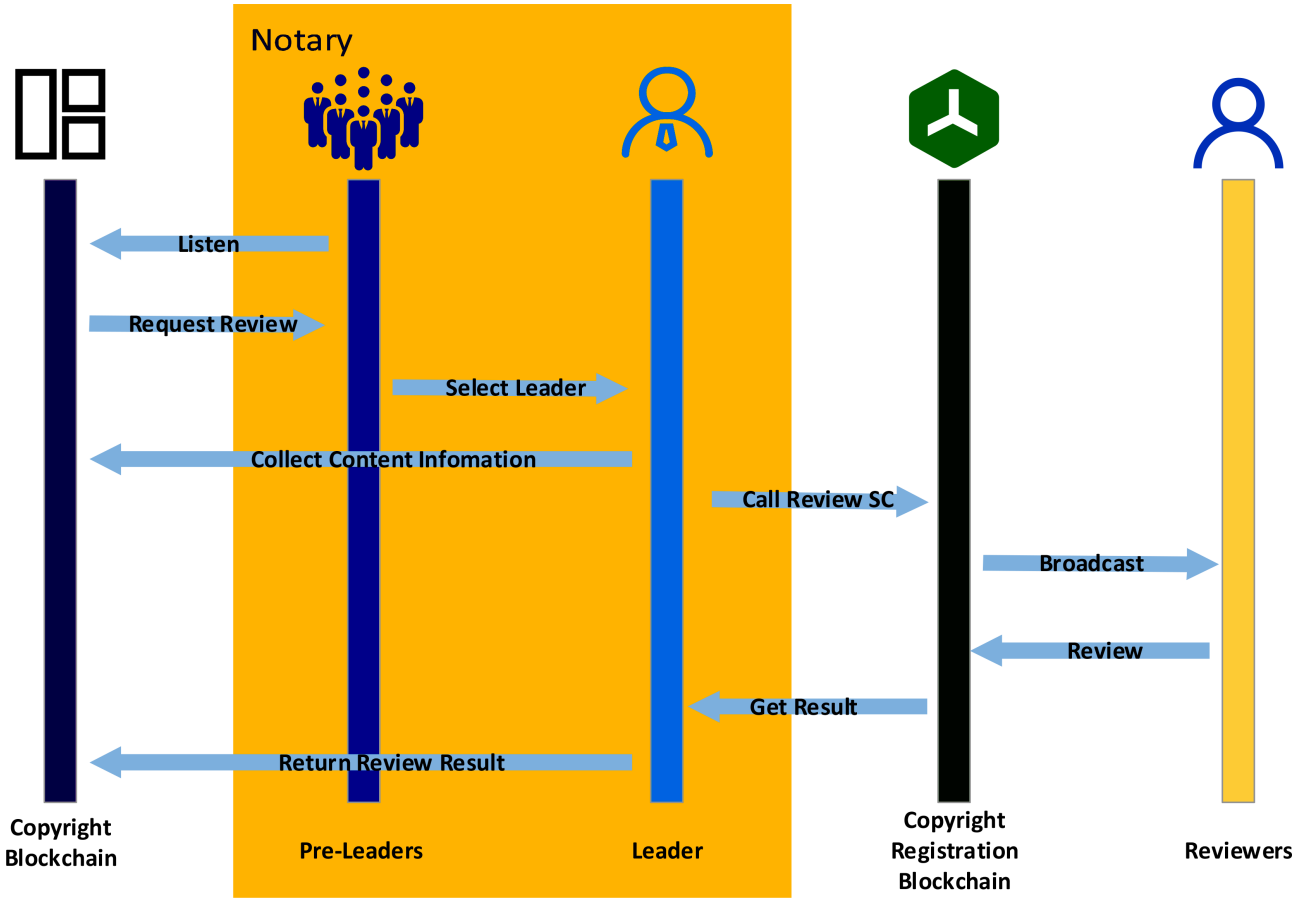

Figure 6. The flow of the review process.

(credits on the copyright review blockchain) pledged by the *i-th* reviewer on content *c*, different-valued credits correspond to different-valued weights in this review round. To ensure the honesty of the review, $d_{i,c}$ *must* be above the minimum to make dishonest reviewers pay more, and $d_{i,c}$ *must* be below maximum to ensure that a single reviewer with high weight will not control the entire review result. The specific formula is as follows:

$$w_{i,c} = \frac{d_{i,c}}{\sum_{i=1}^{N} d_{i,c}} \times N, \qquad \min \le d_{i,c} \le \max \tag{3}$$

Reviewers are more professionals than voters; the review results should be more rigorous to ensure infringement's difficulty and confirmation accuracy. The review process refers to the Likert scale. Reviewers review content by scoring it; HBRO uses integers from 1 to 4 points to evaluate the content score. 1-point means fail entirely, 2-point means may fail, 3-point means may pass, and 4-point means pass entirely. There are 25 reviewers for each content, the upper limit score is 100. $S_{i,c}$ denotes the score of the *i-th* reviewer for content *c*, and $Score_c$ represents the total score of content *c*. During the review process, the leader will aggregate and evaluate each review result of the proposal production.

$$Score_c = \sum_{i=1}^{N} \left(s_{i,c} \times w_{i,c} \right) \tag{4}$$

In the case that every reviewer $i \in N$ gets the same w_{ic}, when the score of 2 points and 3 points is the same, the critical value is 63 (12 for 2 points, 13 for 3 points), which means that when $Score_c \geq 63$, the review result is Pass, otherwise, it is Fail.

$$ReviewResult = \begin{cases} \text{Pass} & Score_c \geq 63 \\ \text{Fail} & Score_c < 63 \end{cases} \tag{5}$$

After the Leader signs the $ReviewResult_c$ by using his private key, the leader sends the $sig_{sk(leader)}(ReviewResult_c)$ to the *Oracle SC* through the designated account of the copyright blockchain.

5.3.3. Voting Process

Figure 7 describes the voting process on the copyright blockchain. When *Voting SC* is invoked, the $Reward_V$ for the voting process will be stored in the *Reward Pool*, and the *Voting SC* requests voters to vote for the content. Voters can first find the processed content in IPFS according to the $IPFS_{po}$, and they could submit a deposit to earn the voting rights. If the voter's reputation value *rep* reaches the threshold, the corresponding voting rights can be obtained, e.g., the initial rep is 100, and the requirement to become a voter is 75, which means that when rep < 75, the vote right cannot be obtained temporarily.

Suppose that there are N voters and each voter has the same voting weight. $T_{i,c}$ means that the content c is passed for the *i-th* voter, $F_{j,c}$ means that the content c is failed for the *j-th* voter, the following formulas $C_{vote}(T)$ and $C_{vote}(F)$ denote the total number of votes for Pass and Fail. As shown in Formula (6)-(8):

$$C_{vote}(T) = \sum_{i=1}^{N} T_{i,c} \tag{6}$$

$$C_{vote}(F) = \sum_{i=1}^{N} F_{i,c} \tag{7}$$

$$VoteResult = Max\left(C_{vote}(T), C_{vote}(F)\right) \tag{8}$$

When the voting process is over, *Voting SC* calculates the amount of $C_{vote}(T)$ and $C_{vote}(F)$. If one party is more than 1/2 of the total number of votes, it is the final $VotingResult_c$. *Voting SC* will return the result to *Oracle SC*. The $VotingResult_c$ is considered correct when it is the same as the $ReviewResult_c$; The system will reward the correct voter whose vote answer is the same as the $VotingResult_c$ with

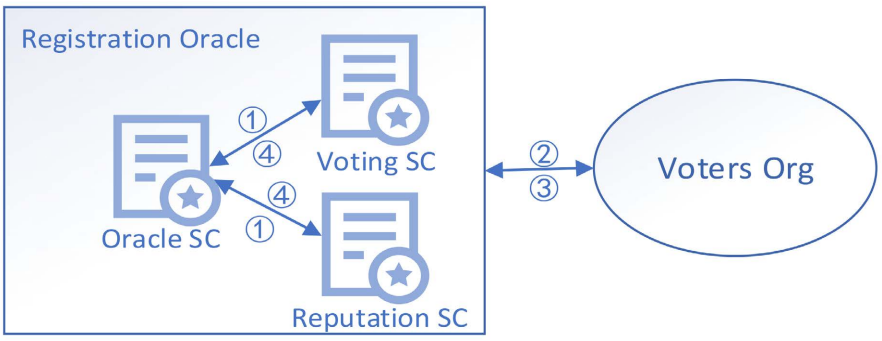

Figure 7. The flow of the voting process.

token and reputation by calling *Reputation SC*.

5.3.4. Register and Refund

After Oracle SC receives $ReviewResult_c$ and $VoteResult_c$, it compares the two results and generates the final $RegistrationResult_c$. This process follows the following principles:

ReviewResult / VoteResult	TRUE	FALSE
TRUE	Pass	Fail
FALSE	Fail	Fail

If the *RegistrationResult* is PASS, *Oracle SC* will return the evidence that denotes by $<sig_{sk(leader)}\ (ReviewResult_c), VoteResult_c, RegistrationResult_c, OwnerAddr, Name, IPFS_{pc}, Proof>$ to the owner. The Leader in charge signs the $ReviewResult_c$, and the public key and the Pre-Leaders' ids have been stored in the mapping table of the copyright blockchain, so the content owner and other nodes can verify the validity of the signature. The *Oracle SC* storage mapping relationship can be mapped to the content owner's address through $IPFS_{pc}$. After the miner packs the evidence of the content on the copyright blockchain, the content owner will obtain a certified certificate, which contains information about content and the block's location, and all nodes can query the block transaction. The content consumer who needs to purchase the content's copyright can trade with the content owner to obtain symmetric key and $IPFS_{fc}$, and download encrypted *fc* on IPFS, eventually get the origin *fc*.

Sometimes, if the $ReviewResult_c$ or $VoteResult_c$ is FALL, the $RegistrationResult_c$ of the content is FALL, then the *Refund SC* has been triggered automatically. The HBRO returns the *service fee* to the content owner, but the $Reward_V$ and $Reward_R$ used for voting and review are deducted.

5.3.5. Reward and Punishment

HBRO restricts the behaviour of nodes through a reward and punishment mechanism. Correct nodes will receive corresponding rewards and improve their reputation. In contrast, malicious nodes will pay for their wrong behaviour. Not only can they not get rewards, but they will also lose their pledged deposits.

The incentive for reviewers is to distribute $Reward_R$ through Leader on the copyright review blockchain. Each reviewer submits additional credits as a stake during the review process to obtain different review weights. The reward for the correct reviewer is allocated according to the reviewer's weight $w_{i,c}$. Assuming that N reviewers participated in the review process, the reward distribution is calculated according to the following formula. $Credit_{i,p}$ represents that the credits submitted by the reviewer i for content c during the review process, $i \in N$. $reward_{r,i,c}$ denotes the reward for reviewer i in the review process of the content c:

$$reward_{r,i,c} = \left(\sum_{i=1}^{N} credit_{i,c}\right) \times \frac{w_{i,c}}{\sum_{i=1}^{N} w_{i,c}} \quad (9)$$

$Reward_V$ provides the reward for voters in the Reward Pool and *Reputation SC*, and the voter's reputation is modified through *Reputation SC* to ensure the voter's high honesty. The voter's reward is fairly distributed; as long as the result is deemed correct, the voter can obtain the reward. Assuming that there are N voters in the system, $reward_{v,i,c}$ denotes the reward for each voter i who votes to content c, N denotes the correct voter numbers:

$$reward_{v,i,c} = \frac{1}{N} \times Reward_V \quad (10)$$

6. Experimental Evaluation and Security Analysis

6.1. Experimental Evaluation

We simulate the whole system process and implement the architecture service logic. We utilize the Ethereum and Hyperledger Fabric to act as copyright blockchain and copyright review blockchain. For external storage, we store the content in IPFS to solve the blockchain big data storage problem. Solidity, golang, Node.js, Truffle, ganache-cli, webpack are used for test environment setup and smart contracts development.

The following table is the hardware and software parameters of our experiment:

Key	Value
CPU	Intel i5-4210M
Operating System	Ubuntu@18.04
RAM	2 GB
ROM	40 GB
External Storage	IPFS@2.13.0
Develop tools	Truffle@5.3.9 Golang@1.16.5 Solidity@0.4.26 webpack Remix
Blockchain	Hyperledger Fabric@2.3 Ethereum
Test Network	Goerli
Wallet	MetaMask@10.6.4

a. hardware and software parameters of our experiment.

We preprocess the content, store the encrypted full content and processed content on IPFS respectively, and request the Ethereum smart contract for proof record and copyright registration. The algorithm logic of proof record is as follows, we upload specific audio to ipfs and request Oracle SC for the proof record,

$IPFS_{fc}$ is *QmYDfbGaPMC5jWxzM7hPMGm1ZfPg3SEQkB5AnN9VMd5RsD*

$IPFS_{pc}$ is QmU1A4bzne7v6AsiR3VQN4DdiBwrdpWERuxq5rXxeZGfvq

Algorithm 1: ProofRecord function of the content

Input: $IPFS_{pc}$
Output: *OwnerAddr,IsExist*

```
function ProofRecord (IPFS_pc)
    Query the global mapping IpfsToAddr to find if IPFS_pc already exists
    if IpfsToAddr [IPFS_pc] ≠ 0 then
        The IPFS_pc already exists
        return IpfsToAddr[IPFS_pc], false
    else
        Let IpfsToAddr[IPFS_pc] ← OwnerAddr
        return OwnerAddr, true
    end if
end function
```

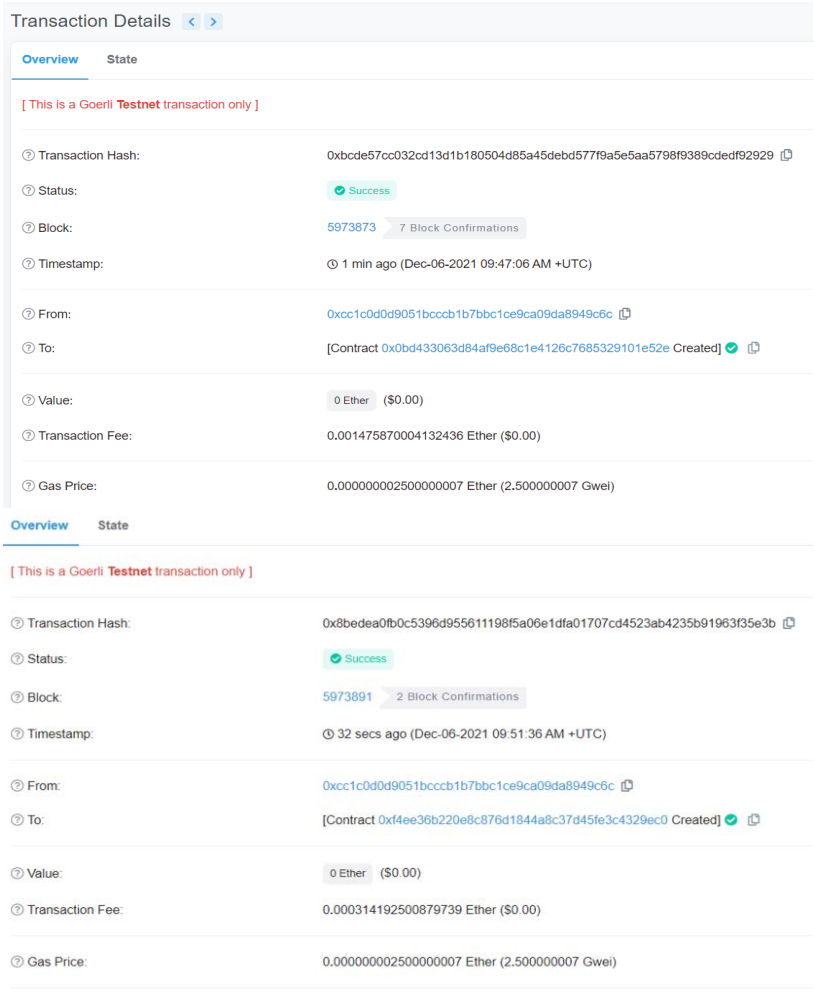

Figure 8. the information of Oracle SC and Voting SC.

Goerli is a test network of Ethereum; We develop smart contracts on the Remix compiler (Its environment is Injected Web3) and deploy *Oracle SC* and *Voting SC* through the Metamask wallet. Specific smart contract information is shown in **Figure 8**. *Oracle SC* is the core of the entire system. Its specific algo-

rithm logic is as follows. The content owner can request the copyright registration contract.

We deployed *Review SC* on the Fabric and set up different organizations for different tasks, such as reviewer org for review and *Review SC* for execution. Regarding the copyright registration of unqualified works, we conducted a comparative experiment. Assuming that the attacker used a large number of pirated contents that were difficult to identify, the number of contents rejected under the same degree of review, the performance of TDRM and HBRO is shown in Figure 9.

Algorithm 2: Copyright Registration function of the content

Input: *OwnerAddr, Name,* $IPFS_{pc}$*, Token, Proof*
Output: $ReviewResult_c$, $VoteResult_c$, $RegistrationResult_c$

```
function CopyrightRes(OwnerAddr, Name, IPFS_pc, Token, Proof)
    Check the Proof to ensure the proof record of the content has been
finished
    if the Proof is incorrect then
        return null, false
    end if
    Continue with the Review Process and Voting Process, assign the Token to Reward_R, Reward_v, and service fee
    Let Reward_R ← Token for review
    Let Reward_V ← Token for vote
    Let Service fee ← Token for service fee
    Call RequestReview SC for review, awaiting review by reviewers on the copyright review blockchain.
  Call the Voting SC's Vote function to request voters to vote on the current blockchain
  After receiving the ReviewResult_c and VoteResult_c, aggregate them to generate the final RegistrationResult_c
  if ReviewResult_c ≠ VoteResult_c
      return null, false
  end
  else
      Let RegistrationResult_c ← aggregate(ReviewResult_c, VoteResult_c)
  end if
  return RegistrationResult_c, true
end function
```

With different amounts of pirated content, HBRO will have more failed registrations. HBRO will have more rigorous registration than existing traditional review methods, which promotes the optimization of copyright ecology.

For the overall performance comparison of HBRO, we have also analyzed it through experiments. As shown in Figure 10, when the point is more outside, it represents the advantage of the DRM in this field.

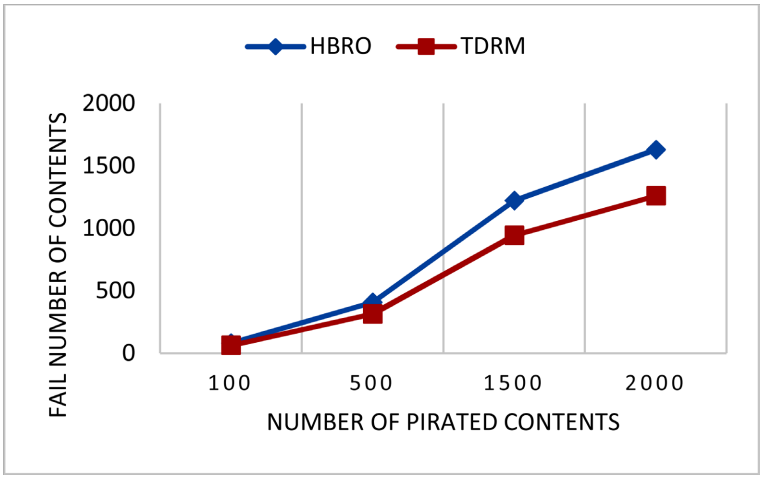

Figure 9. The Performance of TDRM and HBRO.

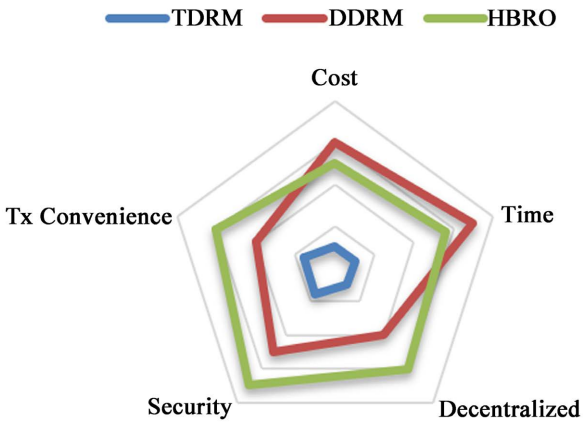

Figure 10. Comparison of 3 types of DRM.

6.2. Security Analysis

In our design, HBRO implements several security attributes to ensure the security of the entire system, which we will discuss here:

Accountability: HBRO is an oracle scheme, the accountability [36] of external data sources is essential. Only authentic data can guarantee the security and accuracy of the blockchain system. Malicious attackers often try to submit wrong results to make the voting or review process go wrong, causing the system to malfunction.

Every vote or review will be stored on the blockchain. To realize accountability, we use digital signatures to ensure the authenticity of external data. If an attacker submits an incorrect result, not only is it easier to lose his/her stake, his/her reputation will also be deducted so that digital signatures can trace the external data source in the system, and malicious attackers' damage to HBRO will make them lose even more, the proposed system significantly reducing the motivation for the attack. For illegal operations, all nodes on the blockchain can find the corresponding evil node address based on the history of the blockchain and pull the address into the blacklist. The Leader plays a very high authority

role in the review process. To prevent the Leader from doing evil behaviour, we set a dynamic map structure on the copyright blockchain to store every Pre-Leader's id and public key address. After the Leader signs the review result, the node can verify the result on the copyright blockchain, if the Leader violates the rules, the Leader would be punished.

No single point of failure (SPOF): A single point of failure is common in many centralized systems. As long as one single point in the system fails, it will cause the entire system to fail. HBRO is a DRM system based on heterogeneous blockchains, transactions are stored on the blockchain, and if a single node tries to tamper with the data on HBRO, it is impossible to succeed without controlling most of the computing power or nodes. The peer-to-peer distributed feature of the blockchain can ensure that the system is decentralized and will not cause a single point of failure.

Anti-Sybil attack: Sybil attack [37] is a systematic attack method in decentralized networks. The attacker weakens the redundancy of the entire network by generating multiple identities. HBRO reduces the occurrence of Sybil attacks by adding a reward and punishment mechanism and reduces the attacker's motivation to attack the system by making the attack cost greater than the benefits obtained from the attack. In addition, reputation can also affect the frequency of Sybil attacks. The system will prioritize the voters' reputation to determine whether they are eligible to be voters, raising the threshold of attack.

7. Future Work

In this section, we propose improvements as future work.

Multiple signatures can be used in the proposed system to protect data security in a permissioned blockchain. Leader generates *credits* through *Credit SC*, relying on the public key to verify the signature may be less efficient. The reward allocated by behavior can be distributed using *m*-*n* multi-signature and m Pre-Leader private key signatures, similar to the multi-signature on Bitcoin.

Regarding the expansion of copyright transactions, setting up agents can ensure the convenience of transactions. Combining key distribution and zero-knowledge proof [38], the system can allow users who have purchased the copyright to act as the agent to trade copyright with the current consumer without the need for content owner access.

8. Conclusions

This paper proposes HBRO, a registration oracle scheme for digital rights management based on heterogeneous blockchains, improved on the voting-based blockchain oracle, combining voting on a permissionless blockchain and review on a permissioned blockchain, and enabling data transmission on heterogeneous blockchains through the notary mechanism. The content owner submits the relevant information of the production to HBRO, voters and reviewers will vote and review the content respectively. After aggregation by *Oracle SC*, the final

registration results will be returned to the content owner. The whole process guarantees the honesty of participants through the reward and punishment mechanism, and Leader acts as a notary in the system.

Our scheme considers the security and integrity of the data and guarantees the qualification and honesty requirements for participants, which enhances the degree of robustness of the system. Through the interaction of review and voting, the quality of copyright registered works is improved, and specification and rigor of DRM are guaranteed. HBRO is more decentralized than TDRM and DDRM, preventing power from being concentrated in the hands of a few entities and increasing the participation of copyright confirmation through the reward and punishment mechanism, making the copyright ecology more healthy.

Funding

National Natural Science Foundation of China (Grant No. 61932011), Guangdong Basic and Applied Basic Research Foundation (Grant No. 2019B1515120010), Guangdong KeyR&D Plan2020 (No. 2020B0101090002), National KeyR&D Plan2020 (No. 2020YFB1005600).

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Rosenblatt, B., Trippe, B. and Mooney, S. (2002) Digital Rights Management. M&T Books, New York.

[2] Copyright Protection Center of China. https://www.ccopyright.com.cn

[3] MONEGRAPH. https://monegraph.com

[4] Chohan, U.W. (2021) Non-Fungible Tokens: Blockchains, Scarcity, and Value. Critical Blockchain Research Initiative (CBRI) Working Papers. https://doi.org/10.2139/ssrn.3822743

[5] Mediachain Labs (2018) Mediachain Attribution Engine: Find Great Images and Reward Creators. http://www.mediachain.io/

[6] Verisart. https://www.verisart.com

[7] Binded. https://binded.com

[8] Garoffolo, A., Kaidalov, D. and Oliynykov, R. (2020) Zendoo: A zk-SNARK Verifiable Cross-Chain Transfer Protocol Enabling Decoupled and Decentralized Sidechains. 2020 *IEEE* 40*th International Conference on Distributed Computing Systems* (*ICDCS*), Singapore, 29 November-1 December 2020, 1257-1262. https://doi.org/10.1109/ICDCS47774.2020.00161

[9] Nakamoto, S. (2008) Bitcoin: A Peer-to-Peer Electronic Cash System. Decentralized Business Review.

[10] Khan, M.A. and Salah, K. (2018) IoT Security: Review, Blockchain Solutions, and Open Challenges. *Future Generation Computer Systems*, **82**, 395-411. https://doi.org/10.1016/j.future.2017.11.022

[11] Wood, G. (2014) Ethereum: A Secure Decentralisedgeneralised Transaction Ledger. *Ethereum Project Yellow Paper*, **151**, 1-32.

[12] Androulaki, E., Barger, A., Bortnikov, V., *et al.* (2018) Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. *Proceedings of the Thirteenth EuroSys Conference*, Porto, 23-26 April 2018, 1-15. https://doi.org/10.1145/3190508.3190538

[13] Szabo, N. (1997) Formalizing and Securing Relationships on Public Networks. *First Monday*, **2**, No. 9. https://doi.org/10.5210/fm.v2i9.548

[14] Morgan, J.P. (2018) A Permissioned Implementation of Ethereum. https://github.com/jpmorganchase/quorum

[15] Wanchain: Building Super Financial Markets for the New Digital Economy. https://www.wanchain.org/files/Wanchain-Whitepaper-EN-version.pdf

[16] Tolk, A. and Muguira, J.A. (2003) The Levels of Conceptual Interoperability Model. *Proceedings of the* 2003 *Fall Simulation Interoperability Workshop*, Orlando, 14-19 September 2003, 1-11.

[17] Belchior, R., Vasconcelos, A., Guerreiro, S., *et al.* (2021) A Survey on Blockchain Interoperability: Past, Present, and Future Trends. *ACM Computing Surveys* (*CSUR*), **54**, 1-41. https://doi.org/10.1145/3471140

[18] German, C. (2018) The Blockchain Island. *ITNOW*, **60**, 20-21. https://doi.org/10.1093/itnow/bwy092

[19] Buterin, V. (2016) Chain Interoperability. R3 Research Paper.

[20] Herlihy, M. (2018) Atomic Cross-Chain Swaps. *Proceedings of the* 2018 *ACM Symposium on Principles of Distributed Computing*, Egham, 23-27 July 2018, 245-254. https://doi.org/10.1145/3212734.3212736

[21] Lo, S.K., Xu, X., Staples, M., *et al.* (2020) Reliability Analysis for Blockchain Oracles. *Computers & Electrical Engineering*, **83**, Article ID: 106582. https://doi.org/10.1016/j.compeleceng.2020.106582

[22] Oraclize API Documentation. http://docs.oraclize.it/

[23] Ellis, S., Juels, A. and Nazarov, S. (2017) Chainlink: A Decentralized Oracle Network. White Paper. https://research.chain.link/whitepaper-v1.pdf

[24] Zhang, F., Cecchetti, E., Croman, K., *et al.* (2016) Town Crier: An Authenticated Data Feed for Smart Contracts. *Proceedings of the* 2016 *ACM SIGSAC Conference on Computer and Communications Security*, Vienna, 24-28 October 2016, 270-282. https://doi.org/10.1145/2976749.2978326

[25] Hess, Z., Malahov, Y. and Pettersson, J. (2017) Æternity Blockchain. https://raw.githubusercontent.com/keypair/white-paper/master/aeternity-whitepaper.pdf

[26] Peterson, J. and Krug, J. (2015) Augur: A Decentralized, Open-Source Platform for Prediction Markets. arXiv:1501.01042.

[27] Benet, J. (2014) IPFS-Content Addressed, Versioned, p2p File System. arXiv:1407.3561.

[28] FISCO BCOS. The Building Block of Open Consortium Chain. http://www.fisco-bcos.org/

[29] Sukhwani, H., Martínez, J.M., Chang, X., *et al.* (2017) Performance Modeling of PBFT Consensus Process for Permissioned Blockchain Network (Hyperledgerfabric). 2017 *IEEE* 36th *Symposium on Reliable Distributed Systems* (*SRDS*), Hong Kong, 26-29 September 2017, 253-255. https://doi.org/10.1109/SRDS.2017.36

[30] Ma, Z.F., Jiang, M., Gao, H.M., *et al.* (2018) Blockchain for Digital Rights Manage-

ment. *Future Generation Computer Systems*, **89**, 746-764. https://doi.org/10.1016/j.future.2018.07.029

[31] Chinese Copyright Blockchain. http://copyright.weibo.com

[32] ChainMaker. https://chainmaker.org.cn/

[33] Zhang, Y., Tang, Z., Huang, J., *et al.* (2020) A Decentralized Model for Spatial Data Digital Rights Management. *ISPRS International Journal of Geo-Information*, **9**, Article No. 84. https://doi.org/10.3390/ijgi9020084

[34] Adler, J., Berryhill, R., Veneris, A., *et al.* (2018) Astraea: A Decentralized Blockchain Oracle. 2018 *IEEE International Conference on Internet of Things* (*IThings*) *and IEEE Green Computing and Communications* (*GreenCom*) *and IEEE Cyber, Physical and Social Computing* (*CPSCom*) *and IEEE Smart Data* (*SmartData*), Halifax, 30 July-3 August 2018, 1145-1152. https://doi.org/10.1109/Cybermatics_2018.2018.00207

[35] Kamiya, R. (2018) Shintaku: An End-to-End-Decentralized General-Purpose Blockchain Oracle System. Gitlab.com. https://gitlab.com/shintakugroup/paper/blob/master/shintaku.pdf

[36] Neisse, R., Steri, G. and Nai-Fovino, I. (2017) A Blockchain-Based Approach for Data Accountability and Provenance Tracking. *Proceedings of the* 12th *International Conference on Availability, Reliability and Security*, Reggio Calabri, 29 August-1 September 2017, 1-10. https://doi.org/10.1145/3098954.3098958

[37] Cai, Y.X., Fragkos, G., Tsiropoulou, E.E., *et al.* (2020) A Truth-Inducing Sybil Resistant Decentralized Blockchain Oracle. 2020 2*nd Conference on Blockchain Research & Applications for Innovative Networks and Services* (*BRAINS*), Paris, 28-30 September 2020, 128-135. https://doi.org/10.1109/BRAINS49436.2020.9223272

[38] Cao, Z.H. and Zhao, L. (2021) A Design of Key Distribution Mechanism in Decentralized Digital Rights Management Based on Blockchain and Zero-Knowledge Proof. 2021 *the* 3*rd International Conference on Blockchain Technology*, Shanghai, 26-28 March 2021, 53-59. https://doi.org/10.1145/3460537.3460556