Design of a Blockchain-based Music Licensing Fee Payment System using Shamir's Secret Sharing

Chang Jun Choi
LSware Inc.
Republic of Korea
jacob@lsware.com

Yong Joon Joe
LSware Inc.
Republic of Korea
eugene@lsware.com

Dong Myung Shin
LSware Inc.
Republic of Korea
roland@lsware.com

ABSTRACT

Recently, due to the opaque settlement and distribution method of sound sources, reliability problems related to the distribution of licensing fees have arisen, and various blockchain-based solutions are being studied to solve this problem. If an API of an external system that cannot be controlled on-chain is called through a smart contract, the API will be executed as many times as the number of nodes, which may result in side effects on the external system. Accordingly, this paper proposes a way to safely call the open banking API using Shamir's Secret Sharing scheme.

KEYWORDS

Blockchain, Hyperledger Fabric, Oracle, Music Licensing Fee, Open Banking API, Shamir's Secret Sharing

1 INTRODUCTION

Blockchain is a distributed ledger technology that guarantees transparency and irreversibility of data in a P2P environment. In a distributed network environment, all nodes jointly verify the data, and record the verified data in their respective ledger to ensure the reliability of the data. Due to the nature of smart contracts that are automatically executed when certain conditions are met, transaction execution results must be mutually verifiable, so the transaction execution results of all blockchain nodes must be the same. In Hyperledger Fabric, a permissioned blockchain platform, each endorsing peer node simulates a transaction using chaincode to verify the execution result.

Suppose that an open banking API within a chaincode is called to payment the licensing fee of a music to a specific user. Each endorsing peer node will simulate the transaction by executing the chaincode to verify the validity of the payment transaction requested by the client. At this time, it should be considered that due to external factors such as network latency, the time to call the open banking API may be different for each endorsing peer node. In this case, a technical improvement is needed to solve the problem because the side effect of duplicate execution of the open banking API in the external banking system may occur. To solve this problem, this paper proposes a blockchain-based music licensing fee payment system that allows open banking APIs to be called only when an API execution request of an appropriate number of endorsing peer nodes is received using Shamir's Secret Sharing.

2 BACKGROUNDS

2.1 Blockchain Oracle Problem

An oracle plays a role of connecting what happened in the real world to a smart contract without the intervention of a third party so that data from off-chain can be recorded on-chain. When information that exists outside the blockchain is brought into the blockchain through a smart contract, the problem in which the data cannot be trusted is called the Oracle Problem [1]. Since the blockchain system cannot actively work with external systems and cannot collect external information, it is difficult to independently record data. In the process of calling and receiving external data that cannot be verified inside the blockchain through smart contracts, it is not possible to determine whether the data is reliable or not. Therefore, it is important to determine whether data is reliable data through an oracle that connects the blockchain ecosystem and external systems to ensure data reliability.

2.2 Shamir's Secret Sharing

It is a secret sharing scheme that splits one $Secret$ into multiple parts called $Share$ and distributes it to authorized participants and recovers the $Secret$ only when a certain number of participants gather [2]. This is referred to as '$(t, n) - threshold$', where t is the number of $Shares$ required to recover the $Secret$, and n is the number of people participating in encryption. $Share$ distribution ($Equation$ 1) and $Secret$ recovery ($Equation$ 2) in the Shamir's Secret Sharing scheme are the following polynomials.

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{t-1} x^{t-1} (mod_p) \quad (1)$$

$$f(x) = \sum_{j-0}^{t-1} \left(y_j \prod_{f=0, f \neq j}^{t-1} \frac{x - x_f}{x_j - x_f} \right) mod_p \quad (2)$$

$Share$ is equally distributed to all members, and even if some $Share$ is lost, the original $Secret$ can be safely recovered with only $Share$ above threshold. Shamir's Secret Sharing scheme is suitable for tasks that require consensus of several people rather than by one participant's unilateral decision, or when information needs to be verified by only a few of the participants. However, since all $Shares$ are equal, it is difficult to split according to the security level when dividing $Secret$, and there are limitations in that it is impossible to prevent intentionally giving incorrect values when providing $Shares$.

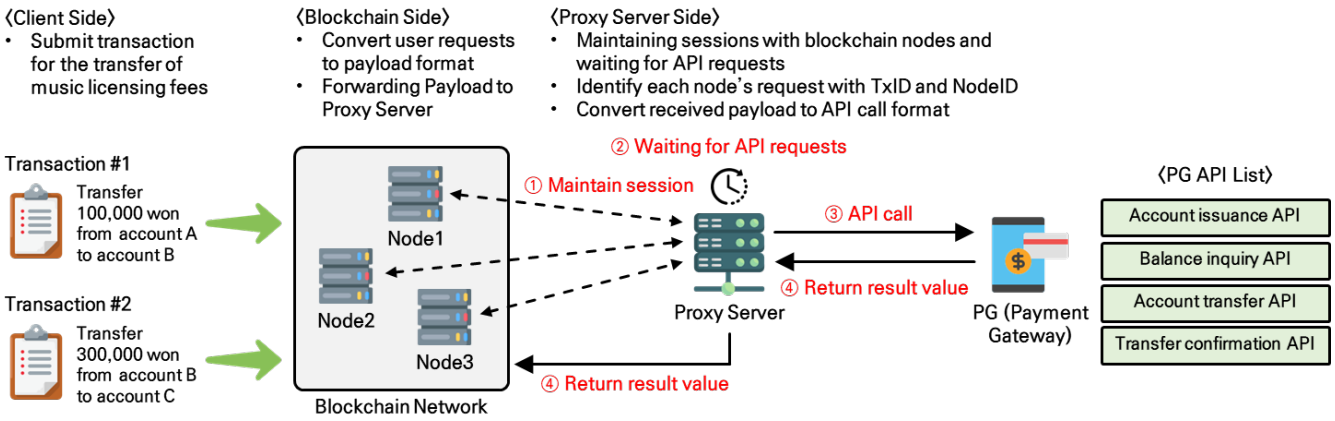

Figure 1: Blockchain-based PG API request and call process with Proxy Server

3 External Open Banking API Call Process

The external banking system targeted in this paper is PG (Payment Gateway) that supports banking services so that online stores can provide customers with various payment methods to safely pay between users. In order to call the open banking API of an external banking system through the block chain, an entity acting as an oracle is required. The method proposed in this paper uses a proxy server that can deliver PG API calls and result values to blockchain nodes. Figure 1 shows the process of calling the PG API using the Proxy Server to transfer the music copyright fee based on the blockchain.

The Proxy Server uses the Shamir's Secret Sharing scheme to generate $Secret$ in advance and distribute it to authorized nodes by dividing it by the number of blockchain nodes. Proxy Server maintains a continuous connection with nodes for simultaneously return PG API result values to each blockchain node.

When a licensing fee transfer transaction occurs, each blockchain node generates information necessary for payment (e.g., transfer amount, deposit account, sender, etc.) and identification information in the form of a payload. After that, the PG API call is requested by passing the payment payload and the $shares$ owned by each to the proxy server. The Proxy Server distinguishes transactions and nodes through the identification information received from each blockchain node, and calls the corresponding PG API when the number of payment payloads exceeds a threshold.

The Proxy Server simultaneously returns the result value received from the PG to the blockchain nodes connected to it. Each blockchain node updates the distributed ledger by recording the result received from the proxy server in the blockchain.

3.1 Comparison with Traditional Method

Compared with the existing method, the method proposed in this paper has the following advantages. When calling the open banking API of an external banking system through a smart contract, it is possible to prevent duplicate execution of commands that cause side effects in the banking system. Since the open banking API of the external banking system is only called when there is a share above the threshold, it is possible to prevent a malicious attacker's crafted API call. That is, in the process of delivering a specific command to an external server, the reliability of the command can be secured. The execution result on the smart contract always matches the execution result of the external banking system.

4 CONCLUSIONS

In this paper, we propose a system that can call the open banking API of an external banking system using Shamir's Secret Sharing scheme to transfer music licensing fees based on blockchain. The proposed system prevents duplicate execution of commands that cause side effects of external banking systems when calling open banking APIs with smart contracts, and through this, reliability can be secured in the process of sending commands to external systems.

In the proposed system, the proxy server acting as an oracle is a single point of failure, so it can be subjected to various network attacks from malicious attackers. Accordingly, as a future study, we intend to conduct research to solve the problem of centralized oracle nodes by analyzing attack scenarios that may occur in the blockchain system. In addition, in the process of delivering the execution result value of the open banking API to the inside of the blockchain, there may be a possibility that the API execution result may be forged or falsified due to various factors. In order to solve this problem, we intend to study a way to cross-verify the result data by multiplexing the proxy server.

ACKNOWLEDGMENTS

This work was supported by Korea Copyright Commission(KCC) grant funded by the Korea government(MCST) (No. 2020-MC-9500, Open API Blockchain Platform for Fair and Transparent Settlement and Distribution of Theme, Background and Signal Music Licensing Fees)

REFERENCES

[1] Caldarelli, Giulio. "Understanding the blockchain oracle problem: A call for action." Information 11.11 (2020): 509.
[2] Son, Ae-Seon, et al. "A Study on Smart Contract Platform Using Secret Sharing Scheme." The Journal of Korean Institute of Information Technology, vol. 18, no. 11, Korean Institute of Information Technology, 30 Nov. 2020, pp. 131–138. Crossref, doi:10.14801/jkiit.2020.18.11.131.