Control Channel Networks: A Fast Blockchain-Based Access Control in Peer-to-Peer Networks

Siwan Noh, Hynwoo Kim, Sang-Uk Shin, and Kyung-Hyune Rhee*
Pukyong National University, Busan, Republic of Korea
{nosiwan, kper591}@pukyong.ac.kr, lacuc.pknu@gmail.com, khrhee@pknu.ac.kr

Abstract

In the past, most medical data were created and managed by hospitals. These days, however, patients can generate these data by themselves without visiting hospitals. Unlike previous data, the latter is generated by patients themselves. To efficiently manage these data without need of the third party, a blockchain-based access control was proposed. If the patient wants to delegate the right of the access to his data, he creates a token for the requester and stores it on the blockchain as a form of the transaction. Moreover, patients can make it invalid and impossible to re-use at any time. However, if the number of the transactions in the blockchain network is exceeded the maximum throughput of the system, the system cannot provide a reliable service. Moreover, due to the feature of the public blockchain, anyone can see the history of the specific data by tracing the transaction stored on the blockchain. In this paper, to solve the scalability problem on the public blockchain based access control system and its invasion of privacy, we propose a token based personal data management system based on . When the requester sending a request message, the resource owner creates a token transaction for the intermediary's address and then asks the delivery of the transaction between his primary chain and the requester's sub chain via multiple sub chain. Finally, the relayed transaction for the access control is stored on the requester's blockchain. The linkability of transactions on the different blockchain is hidden from the third party user and our system is able to process it very quickly regardless of the blockchain's throughput.

Keywords: blockchain, access control, scalability, payment channel

1 Introduction

Only a fer years ago medical records of patients were kept in the form of the paper document by hospitals. However, paper documents required space for storing it and to retrieve a record from document storages is difficult. These days, with the development of the mobile technology, patients can receive a health care service from specialists in each field (e.g. dentist, orthopedist, neurologist, etc.) without visiting a hospital by sending their Personal Health Records(PHRs) that is generated by themselves. Sharing of these PHRs between doctors who belong to the same hospital is relatively easy to achieve than the cross-institutional sharing of the record. In the data sharing protocol, to protect the privacy of patients from unauthorized accesses, a lot of researchers consider a cloud server based system as a storage of patients' PHRs. In order to share his medical record with someone, the patient uploads his PHRs to the cloud server[11, 13, 9, 3]. Cloud computing service enables patients to provide their PHRs with lower cost. However, the stored record contains a critical and sensitive information of the patient. An exposure of this information will cause damage to the finance, social status of patients, etc. Moreover, these records are stored on a semi-trusted third-party server. i.e., the cloud service provider can access these records on the server or modify the data without having a record owner's permission.

Research Briefs on Information & Communication Technology Evolution (ReBICTE), Vol. 4, Article No. 8 (August 15, 2018)
*Corresponding author: Department of IT Convergence and Application Engineering, Pukyona National University, 45Yongso-ro, Nam-gu, Busan, Republic of Korea, Tel: +82-51-629-6247

In this paper, we propose a token-based personal data management system using blockchain technology. The user grants access to his personal data stored on the cloud server for the requester by creating a token transaction on the blockchain. The user can revoke these privileges without the help of the third party. Moreover, if the user wants to update the stored data on the cloud server with issued tokens previously, he can update the data and its corresponding token without the support of the third party.

The outline of the rest of this paper is as follows. In the next section, we briefly introduce the blockchain technology and related work. In section III, we give a system design with our security goals and then describe our protocol. We analyze the proposed system in Chapter IV, and finally, conclude this paper in Chapter V.

2 Related Work

A blockchain is consist of consecutive blocks. Each block contains a hash value of the previous block and transactions. The transaction is propagated to a Peer-to-Peer(P2P) network and nodes in the P2P network propagate the received transaction after performing a transaction validation process. However, if the received transaction is invalid, it cannot be propagated to the P2P network. Also, users store a copy of the blockchain onto the local storage. If someone wants to modify the data stored on the blockchain, a nonce value in the block header must be modified together. However, it is not possible without the alteration of all subsequent block headers for all users in the network.

The first distributed blockchain was conceptualized by Satoshi Nakamoto[6] and it was implemented the following year as a Bitcoin cryptocurrency's basic component. In Bitcoin, a user creates a transfer transaction with his digital signature that uses Elliptic Curve Digital Signature Algorithm (ECDSA) and then it is broadcasted to the bitcoin network. These transactions are verified by network members, usually based on the digital signature verifying the ownership of the previous transaction output. Only if the transaction has a valid digital signature and the previous transaction output is unconsumed, it can be disseminated by network members.

The key idea of the blockchain-based access control system[4, 7, 12] is similar to the transfer of funds in Bitcoin payment system. In [4] Maesa et al. proposed a blockchain based access control system. In the system, each transaction expresses an access right and policies, all transactions are publicly visible on the blockchain. In [4], all users can efficiently manage their data without the help of centralized authorities. However, they do not consider the privacy of users and the limited throughput of the blockchain network.

A public blockchain system has a maximum throughput due to the constant interval and the limited block size. In the case of the bitcoin payment system, it is only 7 transaction/sec. The number of confirmed transactions per day in Bitcoin has been increased by 150,000 in the past 5 years. To enhance throughputs of the blockchain network, many cryptocurrency developers consider a reparametrization(i.e., modify the system parameter). However, Croman et al. show that such scaling by reparametrization can achieve only limited benefits[1]. For instance, the increase of the block size leads to the delay of the propagation. Eventually, the blockchain fork rate is growing as well.

In the Bitcoin payment system, users should wait until the transaction is included in the current block by miners. However, only a few transactions can be included in the block due to the blockchain consensus bottleneck. To solve the low-latency problem in the blockchain system, a payment channel is proposed[]. Two parties establish point-to-point channels between them and exchange their funds on the off-chain(i.e., outside of the blockchain network). In the payment channel, users do not need to wait for the transaction confirmation. However, payment channels only create a relationship between two parties. To extend the payment channel without creating channels individually, a network of payment channels is proposed in [8] with hashed timelock contracts(HTLCs).

3 Proposed System

In this section, we describe our new blockchain based access control system. We assume that the data owner stored an encrypted data that want to share with someone in the cloud server and only if the requester has appropriate access, the data owner provides the decryption right.

Our proposed system consists roughly of two-phase: the setup phase and the access management phase. In the setup phase, the user who wants to share his data with the others(called grantor) or wants to access other users' data(called grantee) establishes control channel between them via intermediaries. In the access management phase, the grantor creates off-chain transaction as a token for access to his data and sends it via intermediaries. After receiving it, the grantee submits the token transaction as a ticket for access to the grantor's data to cloud server administrator with proofs of the token's ownership. The administrator verifies the validity of the token and then only if the token is valid, he provides corresponding data.

3.1 System Model

In this paper, we consider the following scenario as an example: Alice has a chronic disease such as diabetes. Hence, she is taking a health check from the doctor regularly. To increase the accuracy of checks, she uses a wearable smart device. This device consistently checks her diet, glucose levels, and other information. Collected information is stored on the cloud server after encrypting process. To share this information with doctors, she creates the token for her doctor and stores it in the blockchain. The doctor sends the access request message to the cloud server administrator with the . The cloud server administrator provides Alice's data for the doctor only if the doctor has a valid token with proofs of token's ownership. Moreover, it can be revoked by Alice at any time without the help of any others.

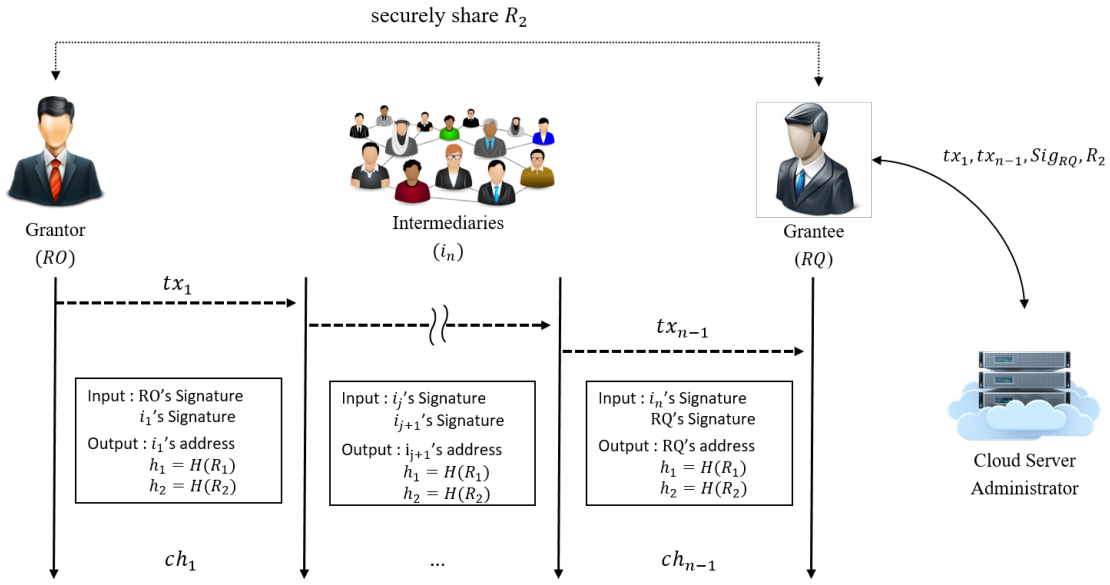

Figure 1: A Structure of control channel networks in the off-chain

We consider a system model shown in Figure.1 which consists of a resource owner(grantor), the requester(grantee), cloud server administrator and intermediaries. A detailed role of entities is as follows:

1. *Resource Owner(RO)*: A resource owner *RO* stores his encrypted data that want to share into the

cloud server. The RO issues the token for access to data when the appropriate requester asked it. At the same time, the RO generates a proxy re-encryption key[10] to delegate decryption right to the requester and sends it to the cloud server administrator.

2. *Requester(RQ)*: A requester wants to access the RO's data stored in the cloud server. After gaining the token for access to data from the RO, The RQ asks the cloud server to provide the RO's data.

3. *Cloud Server Administrator(Ad)*: The Ad is responsible for storing data of RO and transform the encrypted data under the RO's public key into a new ciphertext under the RQ's public key on the same record. CSPs store re-encryption key for re-encryption to its local storage. When the RQ submits the token for access to the stored data, he verifies a validity of the submitted token and the RQ's proof. If the requester has the proper permission, he provides the requested data after performing the re-encryption process.

4. *Intermediaries(i_n)*: A n th intermediary i_n transfers the RO's token transaction on the RO's blockchain to the RQ's blockchain via various blockchain(e.g., Litecoin, Ethereum, etc.) instead of sending it directly. Each hop along the route from the RO to the RQ will create HTLC outputs, which is only released once the control channel is closed.

To design the proposed system, we consider the following security requirements as our design goal.

1. *Rapidity of the token management*: The token for the access to the grantor's data should be managed by the RO without waiting for the confirmation time.

2. *Privacy*: All tokens are stored in the public blockchain, in which the permissions granted by the RO are publicly linked with RO's identifier. It means that a third party can trace the token transaction and know that who is granted by the RO for the access to his data. To avoid this situation, the link between the grantor's identity and the grantees' identity should be hidden on the blockchain.

3.2 Setup

In the setup phase, RO and RQ generates a key pair (sk_x, pk_x) for the ECDSA signature generation and creates his blockchain address using his public key. When the RO stored his data in the cloud server, he computes a hash value of the stored data and creates a *data transaction* that contains this value(e.g., he can use the `OP_RETURN` script code in bitcoin system.). The RO randomly chooses two integer R_1, R_2 and shares the random number R_2 with the RQ through a secure channel[5]. After that, the RO establishes a channel with the first intermediary i_1. The t-th intermediary i_t establishes a new channel with the next intermediary i_{t+1} until they reach the RQ. A channel creating process is same as [2] and a structure of the channel is shown in Figure 1. Each channel on control networks can be created on the different blockchain. For instance, if the RO establishes his channel with the 1st intermediary i_1 on the bitcoin blockchain, the 2nd intermediary i_2 can establish his channel with the next intermediary i_3 on the litecoin blockchain as shown in Figure 2.

3.3 Access Management

If the RQ wants to access the RO's data stored in the cloud server, he sends a request message with the required data's data transaction ID(i.e., a hash value of the transaction). In this point, we do not consider the authentication of the requester's identity, since our research goal is the access control of the user's personal data. If the RO want to grant the access to requested data for the requester, he creates a *token*

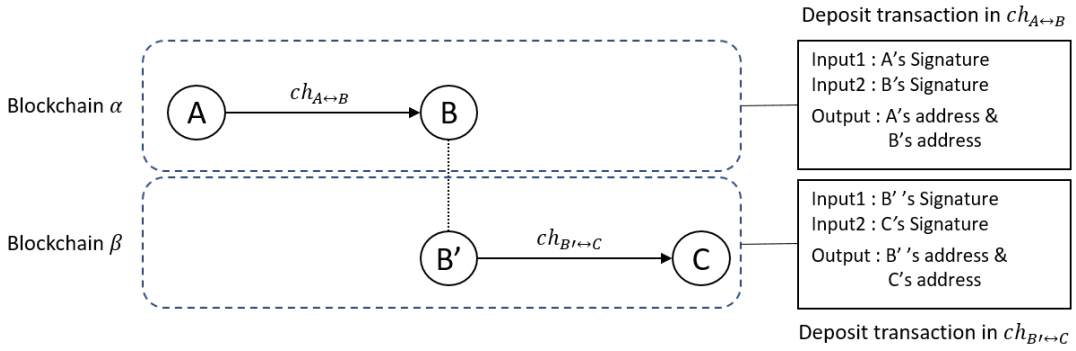

Figure 2: The user B has two accounts on the different blockchain. After creating the channel $ch_{A\leftrightarrow B}$ on the blockchain α, he establishes a new channel $ch_{B\prime\leftrightarrow C}$ on the blockchain β using his account on the β .

transaction on the channel $ch_{RO\leftrightarrow i_1}$. This transaction creates HTLC output as shown in Figure 1. To link the token transaction and the requested data transaction, the RO creates a token data message as follows:

$$msg_{token} =< txID_{data}, address_{RQ}, policy_{data,RQ}, H(R_1), H(R_2), \sigma > \quad (1)$$

$$\sigma = Sig_{sk_{RO}}(txID_{data}, address_{RQ}, policy_{data,RQ}, H(R_1), H(R_2)) > \quad (2)$$

The RO transfers it to the RQ after creating the token transaction tx_1on the channel $ch_{RO\leftrightarrow i_1}$. After the last intermediary i_n creates the token transaction tx_{n-1}on the channel $ch_{i_n\leftrightarrow RQ}$ as shown in Figure 1, the RQ submits the proof of the ownership of the token to the Ad for access. To prove the ownership of the valid token, the RQ submits his signature that can be able to validated by the public key that is written in the output of the token transaction tx_{n-1}, the first token transaction tx_1 and a pre-image of the hash value $H(R_2)$. The Ad checks the state of control networks(i.e., whether channels are closed or not) and verifies the signature and the submitted value R_2'.

4 Analysis

In this section we discuss the security and privacy properties of our system.

4.1 Rapidity of the token management

Token transactions are stored on the off-chain without broadcasting it to the blockchain network. A creating of the token transaction on the channel is able to process it faster than an on-chain transaction. In networks of channels, the confirmation of the transaction is occurred by participants of the channel. That means we do not need to wait for the transaction confirmation time. Although the token transaction is stored in the off-chain, the security of the system still relies on the blockchain network. In the token verifying process, the RQ submits the token transaction $tx_!$ and tx_{n-1}. If the attacker wants to modify the token tx_{n-1} for the unauthorized access to the RO's data, he should be able to generate the last intermediary i_n's signature with the knowledge of the pre-image of the hash value $H(R_2)$.

4.2 Privacy

To avoid the trace of the transaction stored on the public blockchain from the third party, we use control channel networks. In each channel, intermediaries receive the permission from the RO and transfer

it to the RQ using the channel. In this phase, intermediaries can create the token transaction on the different blockchain using his another account(i.e., address). The RQ know that a link information between the first token tx_1 and the last token tx_{n-1}. However, the third party user cannot trace the transaction link information between them. Although the attacker finds the intermediary in the control channel by comparing HTLC outputs, without knowing the number of the intermediary and the kind of the used blockchain, he can not find the link between the first token tx_1 and the last token tx_{n-1}. Moreover, if he finds the link information among them, without knowledge of the hash value $H(R_2)$'s pre-image, he cannot access to the RO'data.

4.3 Integrity and confidentiality of the content of records

In our system, users store their data to the cloud storage. Data in the cloud server are encrypted by their public key to protect their privacy from the malicious cloud server administrator. When the RO uploads his data onto the cloud server, he creates a record transaction that contains a hash value of data. It is like a timestamp of the data. The RQ can verify the validity of the data when he gets it. If a malicious cloud server administrator wants to change some data in the cloud server for the wrong purpose, he must change a corresponding record transaction in the blockchain and the block header that containing the record transaction. However, it will be hard, because of an immutability of the blockchain. Moreover, a confidentiality of the content of records is can be guaranteed by a proof of the scheme in [10].

5 Conclusion

A typical public blockchain has a maximum throughput and does not provide privacy of users. In this paper, to overcome this problem, we propose a payment channel based personal data management system. To reduce the burden of the blockchain network, we use a payment channel scheme proposed in the Bitcoin payment system. Moreover, to protect the privacy of user we use a hashed timelock contract as a proof of knowledge of the relationship between the RO's token and the RQ's token. Due to the features of a blockchain technology, we can make the data owner controls who can access their data in the cloud server without participant of the fully trusted third party.

Acknowledgment

This research was supported by the MSIT(Ministry of Science, ICT),Korea, under the ITRC(Information Technology Research Center) support program (IITP-2018-2015-0-00403)supervised by the IITP(Institute for Information &communications Technology Promotion) and partially supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. NRF-2018R1D1A1 B07048944)

References

[1] K. Croman, C. Decker, I. Eyal, A. E. Gencer, A. Juels, A. Kosba, A. Miller, P. Saxena, E. Shi, E. G. Sirer, et al. On scaling decentralized blockchains. In *Proc. of the 2016 International Conference on Financial Cryptography and Data Security (FC'16), Christ Church, Barbados*, volume 9604 of *Lecture Notes in Computer Science*, pages 106–125. Springer, Berlin, Heidelberg, February 2016.

[2] C. Decker and R. Wattenhofer. A fast and scalable payment network with bitcoin duplex micropayment channels. In *Proc. of the 2015 Symposium on Self-Stabilizing Systems (SSS'15), Edmonton, Alberta, Canada*, volume 9212 of *Lecture Notes in Computer Science*, pages 3–18. Springer, Cham, August 2015.

[3] T. Ermakova and B. Fabian. Secret sharing for health data in multi-provider clouds. In *Proc. of the IEEE 15th Conference on Business Informatics (CBI'13), Vienna, Austria*, pages 93–100. IEEE, July 2013.

[4] D. D. F. Maesa, P. Mori, and L. Ricci. Blockchain based access control. In *Proc. of the 17th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS'17), Neuchâtel, Switzerland*, volume 10320 of *Lecture Notes in Computer Science*, pages 206–220. Springer, Cham, June 2017.

[5] P. McCorry, S. F. Shahandashti, D. Clarke, and F. Hao. Authenticated key exchange over bitcoin. In *Proc. of the 2015 International Conference on Research in Security Standardisation (SSR'15), Tokyo, Japan*, volume 9497 of *Lecture Notes in Computer Science*, pages 3–20. Springer, Cham, December 2015.

[6] S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. `https://bitcoin.org/bitcoin.pdf` [Online; accessed on August 20, 2018], 2008.

[7] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman. Towards a novel privacy-preserving access control model based on blockchain technology in iot. In *Proc. of the 2017 Europe and MENA Cooperation Advances in Information and Communication Technologies*, pages 523–533. Springer, Cham, 2017.

[8] J. Poon and T. Dryja. The bitcoin lightning network: Scalable off-chain instant payments, January 2016. draft version 0.5.9.2.

[9] S. Ruj, M. Stojmenovic, and A. Nayak. Decentralized access control with anonymous authentication of data stored in clouds. *IEEE transactions on parallel and distributed systems*, 25(2):384–394, February 2014.

[10] C. Sur, C. D. Jung, Y. Park, and K. H. Rhee. Chosen-ciphertext secure certificateless proxy re-encryption. In *Proc. of the 2010 IFIP International Conference on Communications and Multimedia Security (CMS'10), Linz, Austria*, volume 6109 of *Lecture Notes in Computer Science*, pages 214–232. Springer, Berlin, Heidelberg, May 2010.

[11] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani. Medshare: Trust-less medical data sharing among cloud service providers via blockchain. *IEEE Access*, 5:14757–14767, July 2017.

[12] Q. Xia, E. B. Sifah, A. Smahi, S. Amofa, and X. Zhang. Bbds: Blockchain-based data sharing for electronic medical records in cloud environments. *Information*, 8(2):44, April 2017.

[13] S. Yu, C. Wang, K. Ren, and W. Lou. Achieving secure, scalable, and fine-grained data access control in cloud computing. In *Proc. of the 2010 IEEE International Conference on Computer Communications, San Diego, California, USA*, pages 1–9. IEEE, March 2010.

Author Biography

Siwan Noh received his M.S. degree in Interdisciplinary Program of Information Security from Pukyong University, Republic of Korea in 2018. He is currently a doctoral course student of Pukyong National University. His research interests are related with blockchain, applied cryptography, and communication security.

Hynwoo Kim received his B. S. degree in Information and Communication Engineering from Dongseo University, Republic of Korea in 2016. He is currently a master course student of Pukyong National University. His research interests are related with blockchain, applied cryptography, and network security.

Sang-Uk Shin received his M.S. and Ph.D. degrees in Department of Computer Science from Pukyong National University, Republic of Korea in 1997 and 2000, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Republic of Korea from 2000 to 2003. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests are related with cryptographic protocol, mobile network security, digital forensic and e-discovery.

Kyung-Hyune Rhee received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Republic of Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Republic of Korea from 1985 to 1993. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Republic of Korea. His research interests center on applied cryptography, communication and network security, and blockchain.