2019 IEEE International Conference on Web Services (ICWS)

Block Chain-Based Data Audit and Access Control Mechanism in Service Collaboration

Chao Wang, Shizhan Chen, Zhiyong Feng, Yanan Jiang, Xiao Xue*
College of Intelligence and Computing
Tianjin University, Tianjin, China
{*taracw, shizhan, zyfeng*}*@tju.edu.cn, hntkjyn@163.com, jzxuexiao@tju.edu.cn*

***Abstract*—In the context of big data, cloud storage services provide users with the ideal data storage service. But external store of data causes cloud storage service providers gain control of the data. Therefore, our work should consider how to ensure the privacy of data and maintain the integrity of data when enjoying convenient services. This paper builds a blockchain-based solution through research on cloud storage service model and blockchain technology. And related protocols are built on the solution-based architecture. In our solution, the decentralized model solves the single point of trust problem in the traditional data auditing service model by collective trust. A public agreement enables auditors to efficiently build proof of data integrity without touching data. The protocol allows users to trace the history of their data, and examine whether the owner of the data guarantees the privacy of the data in an after-the-fact audit. In addition, our work constructed the prototype system of the scheme and revealed the effectiveness of the scheme through system testing.**

***Keywords*-Blockchain; Data Integrity; Data Privacy; Collective Trust; Public Auditability**

I. Introduction

Cloud Storage Service(CSS) [1] has unprecedented advantages: on-demand self-service, ubiquitous network access, location-independent resource pools, fast and flexible resource usage policies. As a disruptive technology with far-reaching impacts, it is changing the nature of enterprise storage resources. In this model, a fundamental shift is that data is being concentrated and stored in the cloud.

In the age of big data and service ecosystem [2], with external store of data, users can eliminate the burden of local data storage and maintenance. In contrast, external store of data causes cloud storage service providers gain control of their data, which leads to security challenges for data and private information [3] in turn. This kind of event that threatens the security of data may be caused by the following reasons: the cloud storage service may still convince the user that it owns the data even if the data stored by the user is completely or partially lost. Cloud storage server misconduct is diverse, including reclaiming storage space by maliciously discarding data that users have not yet or rarely accessed, or hiding data loss events (due to management errors, hardware failures, external or internal attacks). Furthermore, due to lack of supervision and more effective restrictions, cloud storage service providers will leak user private data due to interests and other reasons, for example, it was an incident that FaceBook user privacy data was disclosured recently.

Based on the above reasons, after studying the principle and characteristics of blockchain technology, we propose a data integrity audit and data privacy protection scheme based on blockchain technology. The scheme builds a collective trust model and a public data audit protocol with privacy protection. We solve the single-point trust problem in the traditional audit service model through collective trust model by a decentralized manner. A public audit protocol with privacy protection allows auditors to efficiently perform user data integrity audits without touching user data. Therefore, our main contribution of this paper is as follows. In this paper we:

- propose a blockchain-based data integrity and privacy protection scheme. The solution improves the stability of the service model by constructing a decentralized architecture to solve single point trust and malfunction in centralized services.
- have built a public data agreement. The agreement allows data transfer for the public recording. The transparency and traceability of this operation keeps the data state under public monitoring, which can effectively avoid data storage party fraud. In addition, we introduce a homomorphic tags to ensure that in the audit phase, only the homomorphic tags of the data needs to be audited, thus preventing the load problem caused by all data transmission, and ensuring that the audit node does not touch the user data.
- built the prototype system and tested it. This review verifies the effectiveness of our approach and the adaptability of the scenario.

The rest of the paper is organized as follows. Section II overviews the related work; In Section III, we detail the blockchain-based solution and the public data agreement; In Section IV, we built the prototype system in accordance with the solution described and the public data agreement in Section III, and performed system testing on the prototype system; Section V is the conclusion of this paper.

978-1-7281-2717-0/19/$31.00 ©2019 IEEE
DOI 10.1109/ICWS.2019.00044

II. Related Work

Cloud storage service data integrity audit [4], [5], [6] is to verify whether the cloud storage server can guarantee the integrity of the data and avoid any tampering or deletion of user data. Therefore, the cloud storage service provider's misconduct can be restrained to some extent.

Ateniese et al.[5] firstly proposed a homomorphic tags-based random sampling method in their proposed "Provable Data Possession" (PDP) scheme. Data existence proof has been achieved by random sampling verification of file block sets stored in the cloud storage server. In the case of an audit agency or a user directly conducting a public audit, the program will reveal user privacy. Juel et al.[7] describe a "Proof Of Retrievability" (POR) model that uses sample check and correction code mechanisms to ensure the existence and retrievability of data files on remote data service systems. Although there have been good results with a sufficiently streamlined Merkle tree to build a public POR, this approach is only suitable for public encryption and is not suitable for public audit. In [8], Wang et al. considered similar support for partial dynamic data storage in distributed system, as well as additional features of data mislocalization. In the subsequent work, Wang et al. [8] proposed combining BLS-based homomorphic verifiers with MHT to support public auditability and all data dynamics storage. Wang [9] et al. proposed an improved method based on the [5] scheme to provide a more stringent user privacy protection strategy, and proposed a batch audit solution.

Zyskind [10] proposed a decentralized access control system to protect user data in the form of symmetric encryption, but this solution still cannot retrospect history in the case of user data loss. Liang[11] propose a decentralized and trusted cloud data provenance to verify data security. The provenance auditor verifies provenance data through information in the block. Liu[12] propose a blockchain-based framework for Data Integrity Service.

The solution that we constructed in this paper is not limited to data integrity protection, but also the user data transfer process, and ultimately guarantees user data privacy through post-audit.

III. Solution Based on Blockchain

A. What is a Blockchain

Essentially, a blockchain can be thought of as a distributed ledger: a chain of data blocks arranged in a logical chronological order, where each block contains a record of all valid network activity on the chain [13], [14], [15],[16], [17]. Each block can be defined as a set of encrypted information. In theory, anyone can add data to a blockchain by trading on the network, and anyone can view the data at any time, but no one can make changes to the data without sufficient authorization. Therefore, the blockchain has a complete and unalterable history of network activity that is shared among all nodes of the distributed network. The blockchain is the first technology that enables two or more entities that may not know each other or distrust each other to exchange value in a decentralized network without including a third-party audit and verification authority.

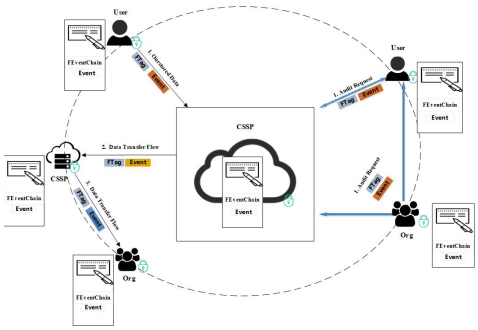

Figure 1. Blockchain-based Application Model.

B. Technical Advantages of Blockchain

The blockchain provides users with several key technical advantages that are the hallmark of their structural architecture. As described below:

1) *Robustness*: Decentralized networks eliminate single points of failure compared to centralized systems. The risk distribution between nodes makes the blockchain more robust than the centralized system and more suitable for blocking deter malicious access.
2) *Transparency*: Each node on the network participates in maintaining the only identical ledger of the blockchain. This ledger allows for real-time review and inspection of data sets. This transparency makes network activity and operations highly visible, reducing the need for trust.
3) *Immutability*: Because each block has a timestamp, the data stored in the blockchain is immutable. The nature allows the user to operate with the highest confidence, which means the data of the blockchain is immutable and accurate.
4) *Traceability*: The blockchain stores all historical data after the genesis block through the chain data structure, and any data on the blockchain can be traced back to the source through the chain structure.

C. Blockchain-based Application Model

Fig. 1 is an overview of the application of the blockchain in user data transfer. We propose a decentralized solution based on blockchain technology. The solution collects, stores, and manages key information for each block of data throughout its transfer process through the blockchain. As user data transfers between entities, it is owned by a variety of participants. The blockchain creates a secure, shared transfer record for each block of each user. These events are

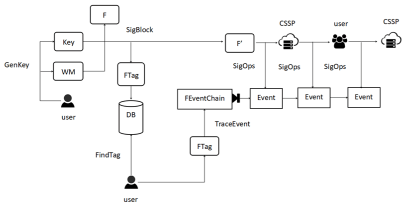

Figure 2. Public Protocol and Data Traceability Protocol.

recorded in the blockchain in a chronological order by the participants. Each user data has a unique event management chain. According to the events recorded in the chain, we can trace the transfer events and build other proof information. For example, the important scenes in this paper are traced back through data transfer to prove the privacy of user data. Verify the integrity of user data by combining the relevant evidence provided by the cloud storage service.

The blockchain technology provides robustness, transparency and immutability that are the underlying guarantees provided by the blockchain foundation services. Essentially, we can directly enjoy these underlying services through the smart contract mechanism. According to the characteristics of the smart contract mechanism, we have developed a set of protocols for building data integrity certification and user data privacy protection.

D. Public Protocol and Data Traceability Protocol

In this scheme, we record the transfer process of a user's data F (usually expressed in the form of a file) in chronological order in the form of an event. For example, when user A saves the data segment F to the cloud storage service provider B, then according to the agreement, the data transfer event signed by the two parties will be recorded in the blockchain through a chain structure, which is the content saved in FEventChain. We can trace the transfer path of user data blocks through the data transfer event traceable protocol. Combining the public process and data integrity certification process, we describe the working logic shown in the Fig. 2 . First, the user combines the blockchain system to generate his own key (sk) information and store it locally, which is done by GenKey protocol algorithm. In the user data transfer, the user constructs a file label (FTag) for each file F (considering the user's data segment as a file), and adds watermark information to the external file through a digital watermark algorithm to form a watermarked file. This step is done by SigBlock protocol algorithm. A transferable event occurs when a watermarked file is transferred between entities that join the blockchain. The blockchain system records the event in the corresponding data transfer event chain when the data transfer event is triggered (the trigger condition is that the parties involved in the file transfer jointly sign the transfer event), and the blockchain completes the step through the SigOps protocol algorithm. Users or authorized organizations can trace the event of a particular file in the blockchain system through the file tag, where the user can combine the FindFTag protocol algorithm and TraceEvent protocol algorithm to complete the traceback process. Through this way, we can complete an after-the-fact trace to prevent the owner of the data cheating.We describe each of algorithm as follows.

1) *GenKey* - When an entity joins the network, the system creates an account information such as a password. The output of the protocol algorithm is the user's key pair (pk, sk).
2) *SigBlock* - Extract the file label of a data block.
3) *SigOps* - This event will be recorded when user data is migrated from one account to another.
4) *FindFTag* - Find file tag information of a piece of data $(FTag)$ in the local database.
5) *TraceEvent* - Retrieve the details of the data block transfer event from the blockchain based on the file label of the data block.

E. Privacy Protection Audit Agreement

Combined with the public process and data integrity certification process, we construct a set of audit business logic, as shown in the Fig. 3 . In summary, the first step is the startup phase, the user (U) generates a key pair and broadcasts the public key (pk) to other cloud storage service providers (CSSP) and auditors (A). The step two, the user (U) computes a homomorphic verifiable digest (HVD) for each data block in file F, and finally each file is constructed as an aggregation of the homomorphic verifiable digest (HVD). Then, the user (U) stores the file (F) and the aggregation in the cloud storage service provider (CSSP) and the audit node (A) respectively. In the verification phase, the audit node (A) requests proof of ownership of the subset of data blocks in the file (F). In the third step, the audit node (A) first generates a verification challenge (ch) according to the homomorphic verifiable digest set (T); Then, the audit node (A) sends the challenge to the cloud storage service provider (CSSP); For the four step, the cloud storage service provider (CSSP) builds the data block possession certificate (V) and sends it back to the audit node (A). At the last step, the results of this challenge are verified by the audit node (A) which is in combine with the verification challenge (chal). This phase can be performed indefinitely to determine if the cloud storage service provider (CSSP) still owns the selected block.We describe each of algorithm as follows.

1) *PregenProof* - Generate corresponding data integrity audit challenges(chal) based on local file tag information(FTag).
2) *GenProof* - The cloud storage service builds the audit results(V) of this challenge based on the verification challenge combined with the locally stored data block information set.

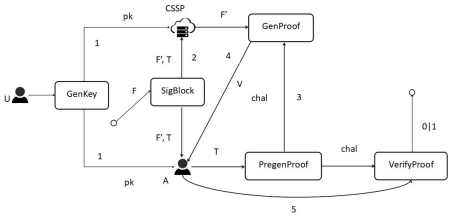

Figure 3. privacy protection audit algorithms.

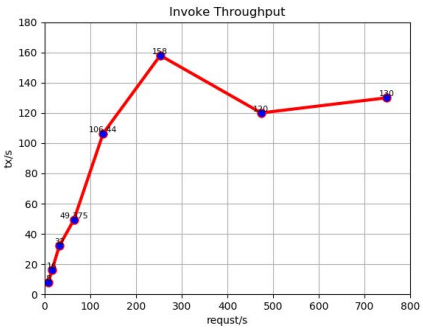

Figure 4. Invoke Throughput Test.

3) *VerifyProof* - The auditor combines the audit results(V), the audit challenges(chal), and the key information(sk) of the audit nodes to construct the verification results (Result)of this audit challenge.

IV. System Performance Evaluation

In the experiment, we built the system-adaptive blockchain platform as Hyperledger Fabric 1.1.0, adopting the most basic distributed system configuration: 1 Orderer 4 Peers network topology, in which the consensus network adopts the single-node solo mode. Due to the distributed system characteristics of the blockchain, in terms of hardware, we use the AWS EC2 cloud host, whose basic configuration is 4CPU, 2.3GHz, 16G RAM. The algorithm uses a pairing-based cryptography (PBC) library version 0.5.14, and the homomorphic summary length is 512 bytes. All the test results in the experiment are the average of the results of the 20 sets of experimental tests.

The stress test of the blockchain is carried out in two aspects that we are most concerned about: 1) blockchain write stress test; 2) blockchain query stress test.

The read/write test is output/input in the form of a Key-Value pair. The size of the Key is 8 bytes, and the value of the Value is 512 bytes.

In the next section, we show more detailed invoke and query test results. We send requests with different send rate. As we showed in Fig. 4, we can see that as the amount of requests per unit time increases, the throughput of platform invoke operations increases first and then decreases, and peaks in Request/s = 253 in our tests. The subsequent section

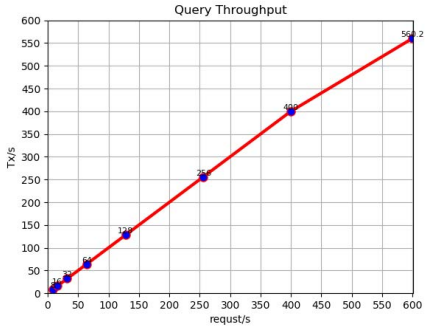

Figure 5. Query Throught Test.

Table I
Invoke Throught

Request/s	Tx/s	Failed
8	8	0
16	16	0
32	32	0
64	46.375	0
128	106.44	0
253	158	0
474	120	2∼4
749	130	3∼7

Table II
Query Throught

Request/s	Tx/s	Failed
8	8	0
16	16	0
32	32	0
64	64	0
128	128	0
256	256	0
400	400	0
600	560.2	3∼6

gradually showed a downward trend. And from the Table I, we can find that when Request/s $\geq$ 474, the test result begins to appear request failure.

In Fig. 5, we show the results of the query throughput test for the current blockchain platform in detail. We can see that as the number of requests per unit time increases, the platform query throughput shows a trend of increasing lines, we can know from Table II, after request/s >400, the test start respond to failures.

V. Conclusion

In this paper, we propose a public data auditing scheme based on collective trust by studying blockchain technology. We include cloud storage service providers and users

in the blockchain network, which makes the behavior of user data processing, including data change records, data audit records, etc., be honestly recorded by the blockchain network. Therefore, we constructed a data audit model that replaces single-point trust with collective trust. On this model, we constructed public data auditing protocol with privacy protection, which enables audit nodes to complete the audit of the integrity of user data without touching the user's private information.

ACKNOWLEDGEMENT

This work is supported by the National Key R&D Program of China grant No.2017YFB1401201, the National Natural Science Foundation of China grant No.61572350 and the Shenzhen Science and Technology Foundation (JCYJ20170816093943197).

REFERENCES

[1] Seny Kamara and Kristin Lauter. Cryptographic cloud storage. In *International Conference on Financial Cryptograpy & Data Security*, 2010.

[2] X. Xue, S. Wang, L. Zhang, Z. Feng, and Y. Guo. Social learning evolution (sle): Computational experiment-based modeling framework of social manufacturing. *IEEE Transactions on Industrial Informatics*, pages 1–1, 2018.

[3] Kaiqi Xiong and Mufaddal Makati. Assessing end-to-end performance and security in cloud computing. In *Symposium on Applied Computing*, 2017.

[4] M. Venkatesh, M. R. Sumalatha, and C. Selvakumar. Improving public auditability, data possession in data storage security for cloud computing. In *International Conference on Recent Trends in Information Technology*, 2012.

[5] Giuseppe Ateniese, Randal C. Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary N. J. Peterson, and Dawn Song. Provable data possession at untrusted stores. In *Acm Conference on Computer & Communications Security*, 2007.

[6] M Shaik Saleem and M. Murali. Privacy-preserving public auditing for data integrity in cloud. 2018.

[7] Ari Juels and Burton S. Kaliski, Jr. Pors: Proofs of retrievability for large files. In *Proceedings of the 14th ACM Conference on Computer and Communications Security*, CCS '07, pages 584–597, New York, NY, USA, 2007. ACM.

[8] Qian Wang, Cong Wang, Kui Ren, Wenjing Lou, and Jin Li. Enabling public auditability and data dynamics for storage security in cloud computing. *IEEE Transactions on Parallel & Distributed Systems*, 22(5):847–859, 2011.

[9] C. Wang, Q. Wang, K. Ren, and W. Lou. Privacy-preserving public auditing for data storage security in cloud computing. In *2010 Proceedings IEEE INFOCOM*, pages 1–9, March 2010.

[10] Guy Zyskind, Oz Nathan, and Alex 'Sandy' Pentland. Decentralizing privacy: Using blockchain to protect personal data. In *IEEE Security & Privacy Workshops*, 2015.

[11] Xueping Liang, Sachin Shetty, Deepak Tosh, Charles Kamhoua, Kevin Kwiat, and Laurent Njilla. Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability. In *Proceedings of the 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, CCGrid '17, pages 468–477, Piscataway, NJ, USA, 2017. IEEE Press.

[12] B. Liu, X. L. Yu, S. Chen, X. Xu, and L. Zhu. Blockchain based data integrity service framework for iot data. In *2017 IEEE International Conference on Web Services (ICWS)*, pages 468–475, June 2017.

[13] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4:2292–2303, 2016.

[14] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

[15] Ahmed Kosba, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *Security & Privacy*, 2016.

[16] Y. Yuan and F. Y. Wang. Blockchain: The state of the art and future trends. *Acta Automatica Sinica*, 2016.

[17] Michael Mainelli and Alistair Milne. The impact and potential of blockchain on securities transaction lifecycle. *Social Science Electronic Publishing*, 2016.