

Blockchain-based Enterprise Community Data Access Control Methods and Applications

Yang Yang^a, Li Wang^b, Tingyezhi Hu^c, Yuan Wang^d

School of Economics and Management, Beihang University, Beijing, China;

^ayy15388128182@buaa.edu.cn, ^bwlbh@vip.163.com, ^cHTYeahZ1002@buaa.edu.cn,

^dbuaaerwy@126.com

Abstract. Data, as a fundamental resource in the digital economy of enterprise communities, can significantly enhance digital development and enterprise growth if shared securely and orderly. Despite providing detailed control, traditional attribute-based data access control methods face challenges related to privacy protection and data ownership, potentially causing data loss and trust issues. To tackle these challenges, this paper presents a blockchain-based data sharing management system, which merges the decentralized and traceable aspects of blockchain technology with attribute-based access control.

Keywords: Blockchain, Data Sharing Management, Data Access Control Methods

1. Introduction

Data serves as a pivotal resource in the digital economy and is a key competitive factor for enterprises. Its commercial value is increasingly recognized, and secure, orderly data sharing within enterprise communities can significantly unlock this value, fostering digital economy development and enterprise growth[1]. However, most enterprises manage data in centralized, isolated systems, creating data silos that hinder effective sharing, coordination, and cooperation within communities[2].

Recent advances in information technologies, particularly the Internet, have spurred innovation and upgrades in enterprise communities, leading to the emergence of numerous data sharing platforms. These platforms, offering both online and offline services, facilitate inter-enterprise data sharing. Typically, enterprise communities utilize attribute-based access control mechanisms that offer detailed permissions. However, significant issues persist in current data sharing solutions. Firstly, enterprise data often includes sensitive corporate and user information, such as browsing histories and preferences, posing privacy risks if improperly shared. Existing solutions inadequately enable data owners to set precise sharing permissions, heightening privacy breach risks. Secondly, the lack of publicly recorded access logs complicates data ownership verification and traceability, potentially resulting in data asset losses, reduced inter-enterprise trust, and hindered collaborative development[3].

This paper tackles these issues by introducing a blockchain-based data sharing management solution tailored for enterprise communities. We develop a platform that enhances privacy protection and data ownership verification, aiming to advance the digital construction of enterprise communities and improve inter-enterprise data sharing.

2. Background

2.1 Enterprise Community Data Sharing

Most enterprise data is generated from daily operations and financial transactions, making it a crucial asset in the data economy. For enterprise communities, data from various sources enables multifaceted user profiling, allowing companies to iteratively develop products that attract users and generate profits. Additionally, data can help identify the causes of anomalies occurring on specific days or during specific periods. However, acquiring, storing, and transforming data incurs costs. If

inter-enterprise data sharing management can be achieved securely, transparently, and efficiently, storing data in a shared system that allows easy access to one's own data and partial access to publicly available data from other enterprises can reduce the unit cost of data storage. Furthermore, enterprise data sources are frequently constrained, and the available data's quantity and quality may fall short of the demands of modern analytical techniques like machine learning and deep learning. By sharing data with other companies, the accuracy of data analysis can be enhanced, leading to more precise business decisions[4].

Traditional inter-enterprise data sharing management uses attribute-based access control techniques to achieve detailed data access control.. Key entities involved include enterprise users (acting as both data providers and accessors), blockchain certificate authorities (CAs), blockchain networks, third-party audit institutions, and distributed databases like IPFS. The process involves system initialization, user registration, third-party review, issuance of digital certificates, attribute assignment, and access provisioning. However, reliance on third-party institutions for user evaluation introduces significant concerns:

Firstly, the evaluation process lacks transparency. Flaws in evaluation may grant inappropriate access permissions, compromising data privacy. Secondly, enterprise users, as data providers, are unaware of other nodes' attributes, complicating the setting of access permissions. Additionally, the non-disclosure of data access records leaves users unaware of who accessed their data. This opacity can lead to unauthorized data ownership claims and illegal activities like data resale, impacting data ownership verification and trust among enterprises.

In conclusion, existing solutions that involve third-party institutions for enterprise user evaluation may result in data privacy breaches, and the lack of public access to data access records can lead to problems with data ownership verification, thereby affecting trust between enterprises.

2.2 Blockchain-based data sharing solution

Blockchain[5] can be categorized into private chains, consortium chains, and public chains. Private chains do not allow external nodes to join, resulting in weaker decentralization and are used by a limited number of nodes. Public chains, in stark contrast to private chains, are characterized by complete decentralization and do not restrict node participation. Consortium chains possess a semi-open nature, maintained by multiple nodes and enforcing strict identity admission mechanisms[6]. Currently, most scholars domestically and internationally choose consortium chains combined with attribute-based encryption data access control methods to develop data sharing platform systems. For instance, Wang utilized multiple consortium chains and adopted an off-chain data storage approach (IPFS), with on-chain storage of data addresses located in IPFS, thereby implementing an inter-enterprise and intra-enterprise data access control system that addresses data sharing issues and the problem of large blockchain data storage[7].

Since this paper aims to propose a data sharing management solution for enterprise communities, where shared data may involve enterprise users or their user privacy, the developed platform should ensure that shared data circulates only among authorized enterprise users and should be semi-open. A consortium blockchain with an admission mechanism not only protects the data of on-chain enterprise users from external leaks but also leverages the transparency and immutability of blockchain to safeguard enterprise data ownership. Although storing data files on a blockchain consumes significant space, this solution employs IPFS distributed databases for data file storage, storing only the data storage addresses and corresponding encryption keys on the blockchain, thus resolving the issue of blockchain data file storage. Hence, blockchain technology aligns well with the proposed solution outlined in this paper.

3. System Design

3.1 Design Architecture of the Data Sharing Management Plan

To meet the needs of resolving data privacy protection and ownership concerns, this section proposes an overall architecture design for enterprise community data sharing management based on blockchain. The design architecture of the data sharing management plan is shown in Fig 1. The main stakeholders/participants in our system are the data uploader and data requester. Comparing to traditional plans, the blockchain key generation center of the proposed plan does not need to perform the initialization functions of the ABE mechanism as typical blockchain CA entities do during the initialization of the blockchain system. Instead, it waits until a new user has obtained approval from existing enterprise users. Based on the approval results, the new user is assigned an identity number (EID) and a set of random numbers (RPN), and an identity key (UK) is generated for them. Subsequently, the newly registered user can act as a data owner to encrypt and upload shared data, or as a data requester to request access to data uploaded by other users based on their own attributes.

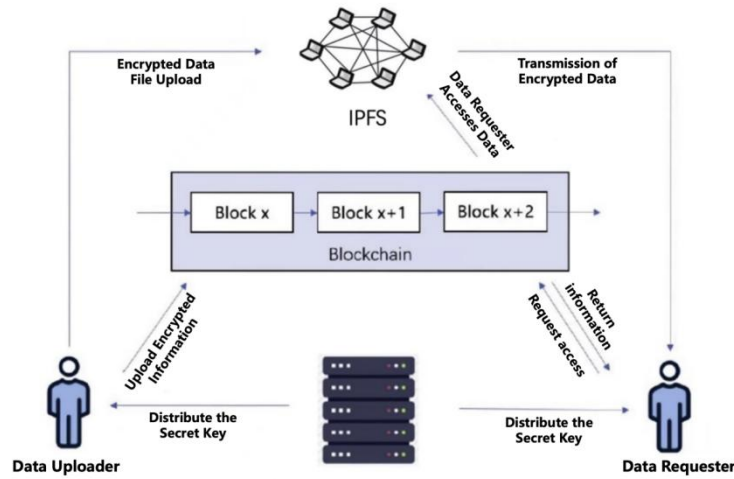


Fig. 1 Design Architecture of the Data Sharing Management Plan

The specific processes and algorithms for enterprise user registration approval, data upload, and data request are as follows:

3.1.1 Enterprise User Registration Review

Initially, a potential enterprise user submits pertinent supporting documents, including balance sheets from the previous three years, project involvement data from the same period, and specifics on sharable data. The prospective user also selects the attributes they wish to apply for. Second, existing enterprise users review the materials submitted by the prospective enterprise user and vote on the attributes applied for based on the review results. If the materials do not pass the review, the prospective enterprise user is denied access. If the materials pass the review, the user is granted access. Finally, once the prospective enterprise user is granted access, the data sharing platform assigns the corresponding permissions to the new user based on the voting results from existing enterprise users regarding the attributes applied for. The platform generates an identity key and a digital signature for the new enterprise user based on their enterprise user identity number and a set of random numbers. Upon gaining access to the enterprise community data sharing platform, the user's materials undergo review and attribute allocation, with all related data securely recorded on the blockchain.

3.1.2 Enterprise Data File Encryption and Upload

Upon obtaining access to the enterprise community data sharing platform, enterprise users simultaneously act as data providers and data requester. During data upload, after the data owner

submits a file, the system encrypts the file using the SM4 encryption algorithm. The encrypted file is subsequently uploaded and stored in the distributed database IPFS, where access permission attributes policy (A, F) is designated for the file. Following this attribute policy (A, F), the system generates and furnishes a corresponding public and private key pair (APK, ASK), encrypts plaintext (M) - comprising addresses and summaries - into ciphertext (CT). Here, A denotes an attribute matrix, with each column representing an attribute as an m-dimensional column vector, and F signifies a function mapping each column of the matrix to an attribute. A random prime number p is generated during the encryption process.

The encryption process can be expressed as:

$$\text{Enc}(p, M(A, F), \{\text{APK}\}) \rightarrow \text{CT}$$

After the data file is encrypted, the system sends the data address (hash value) and the SM4 encryption key, with set attribute access permissions, along with the data summary, to the blockchain for secure storage.

3.1.3 Enterprise Data File Access Authorization

When a data requester (access subject) applies for data access permissions based on a data summary, the platform first checks the attributes of the data requester against the data access policy. Utilizing the outcomes of this verification alongside the identity key, the system produces a private key (SK) and a transformation key (TK). During the pre-decryption step, the system verifies the attribute permissions of the data requester. If the permissions meet the requirements, the decryption is completed, and the plaintext (PM) is obtained. With the storage address of PM and the SM4 symmetric encryption key, the system accesses and decrypts the original data file.

The data file decryption process can be expressed as:

$$\text{Dec}(p, \text{CT}, \{\text{TK}, \text{EID}\}) \rightarrow \text{M}$$

As a result, unauthorized users are unable to acquire the accurate address and encryption key, thereby being prevented from accessing the data file.

3.2 Design of the Data Sharing Management System Architecture

This paper introduces a data sharing management plan based on blockchain technology, which decentralizes the attribute-based data access control method.. By applying a mechanism for decentralized approval of new enterprise users and storing data access records on the blockchain, the scheme addresses the issues of shared data privacy protection and data ownership verification. The layered model of the blockchain-based data sharing management system is shown in Fig2.

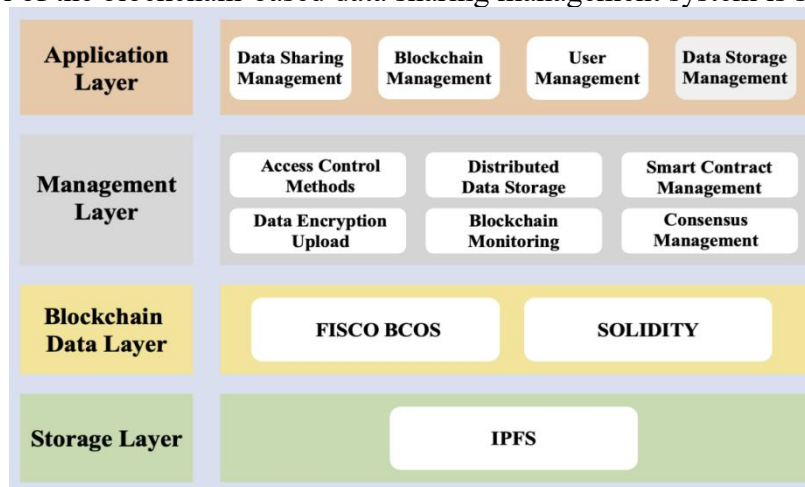


Fig. 2 Layered Model Diagram of the Data Sharing Management System

3.2.1 Storage Layer

The storage layer of the Data Sharing Management System leverages IPFS for distributed storage, forming its foundational component. Smart contracts replace traditional data operation elements, enabling the storage of data files and audit documentation unsuitable for blockchain

storage in a distributed manner via IPFS. Compact information such as addresses and summaries are stored on the blockchain, ensuring data security, traceability, and integrity.

3.2.2Blockchain Data Layer

The system utilizes FISCO BCOS, a widely recognized and applied open-source commercial-grade blockchain in China. FISCO BCOS provides comprehensive scripts for secondary development, enhancing system security. Smart contracts, developed in SOLIDITY using the REMIX online environment, are integrated with JAVA and the SPRING BOOT framework in an offline IDEA environment, and deployed onto the FISCO BCOS consortium chain. The system exclusively uses blockchain for data operations, with no interaction with traditional databases, relying on smart contracts for data uploading, storage, and access.

3.2.3Management Layer

The management layer oversees all underlying system functions and the administration of blockchain smart contracts. This includes the implementation of data access control policies, encrypted data file uploads, IPFS storage methods, data access permission granting, blockchain performance monitoring, smart contract management, and consensus mechanism management.

3.2.4Application Layer

The application layer encompasses several modules, such as the data sharing management, blockchain management, user management, and data storage management modules.

4.System Performance Analysis

4.1System Security Analysis

The proposed scheme implements a decentralized process for assessing and attributing new enterprise users. Compared to existing data sharing schemes, this approach offers two primary security advantages:

a.Transparent and Secure Evaluation: All enterprise users can evaluate new users based on submitted materials, with the resulting attribute assignments stored on-chain. This enhances transparency and security while allowing data providers to control access permissions, thereby improving data privacy protection.

b.Comprehensive Data Tracking: All data uploads and access records are stored on-chain, enabling users to monitor their data's flow and addressing data ownership concerns.

Theoretical Analysis of Common Blockchain Attacks and Mitigation Strategies:

1.Replay Attack[8]: Attackers resend captured communication data. The system uses timestamps in transaction requests to verify and resist such attacks.

2. Man-in-the-Middle Attack[9]: Malicious nodes alter transaction information. Digital signatures assigned during node registration and verified through consensus prevent tampering.

3. Sybil Attack[10]: Nodes forge multiple identities to flood the network with invalid transactions. A strict identity verification mechanism requiring social credit codes and financial reports prevents such attacks.

4. Collusion Attack[11]: Multiple nodes combine privileges to decrypt files. The system ensures that only attribute sets matching the required permissions can access data, preventing unauthorized decryption.

In summary, the proposed scheme enhances data privacy and ownership, effectively countering replay, man-in-the-middle, Sybil, and collusion attacks, thereby demonstrating robust security.

4.1 Upload and Download Efficiency Analysis

In this study, the size of both the attribute set and data files influences the efficiency of encryption and decryption algorithms in the proposed scheme. Through experimentation involving

varied numbers of attributes and data file sizes, we contrast the encryption and decryption rates with those of existing schemes, highlighting the enhanced time efficiency of our approach.

The encryption process proposed begins by utilizing the SM4 symmetric encryption algorithm to encrypt the data files. Subsequently, the encrypted files are stored on IPFS, with the resulting address and the SM4 encryption key encrypted using attribute-based encryption and then stored on the blockchain. For comparison, we select two common schemes:

Scheme A: Uses the CP-ABE algorithm to directly encrypt files.

Scheme B: Uses the AES algorithm to encrypt data files, stores the encrypted files on IPFS, and then encrypts the IPFS address and AES key using the CP-ABE algorithm.

1.Experiment 1: Varying File Sizes

This experiment keeps the number of attributes constant while varying the file size, testing 50 times with files of 500KB, 1000KB, 2000KB, 3000KB, 5000KB, and 10000KB. We compare the average encryption and decryption times of the three schemes. Results show that for files smaller than 5000KB, Scheme B consumes fewer resources and exhibits higher time efficiency. For files larger than 5000KB, the time required by Scheme B increases, whereas the Proposed Scheme demonstrates higher time efficiency. Given that the target data files are generally larger than 5000KB, our proposed scheme exhibits better time efficiency in practical applications compared to other schemes.

Table 1. Upload Time of the Three Schemes

File Size/KB	500	1000	2000	3000	5000	10000
Scheme A	213	216	225	232	251	309
Scheme B	140	153	165	183	225	348
Proposed Scheme	156	180	192	201	206	211

Table 2. Decryption Time of the Three Schemes

File Size/KB	500	1000	2000	3000	5000	10000
Scheme A	47	48	50	49	51	53
Scheme B	45	52	69	97	117	193
Proposed Scheme	60	66	70	71	73	77

2.Experiment 2: Varying Number of Attributes

This experiment tests the impact of attribute quantity on the upload and download times across different schemes. For this experiment, a data file size of 5000KB was chosen, using the same schemes as in Experiment 1. As illustrated in Figure 4.7, the proposed scheme shows superior upload and download efficiency, particularly when handling a larger number of attributes. Given the complex data sharing management requirements in enterprise communities, which necessitate a higher number of attributes, the proposed scheme exhibits better time efficiency in practical applications.

Table 3. Decryption Time of the Three Schemes

File Size/KB	500	1000	2000	3000	5000	10000
Scheme A	47	48	50	49	51	53
Scheme B	45	52	69	97	117	193
Proposed Scheme	60	66	70	71	73	77

This chapter presents a theoretical analysis to show how the proposed scheme effectively counters four common blockchain attack vectors, highlighting the system's strong security

performance. Additionally, through simulation experiments and comparisons with existing literature, this chapter verifies the efficiency of the proposed approach's encryption and decryption processes.

4. Summary

The management of data sharing within enterprise communities has certain hardware and software foundations. However, current data sharing solutions face challenges related to data privacy protection and ownership, which impede trust and collaboration between enterprises. To address these issues, we propose a blockchain-based data sharing management solution for enterprise communities. By integrating SM4 encryption with the ABE data access control algorithm, our solution improves upon existing third-party audit and evaluation methods, enabling transparent and decentralized on-chain auditing and permission granting for new enterprise users. This enhances enterprises' ability to protect their data privacy and ensures controlled data sharing. Additionally, by storing data files off-chain and data addresses, audit data, and access process data on-chain, we resolve the challenges of data ownership verification.

Furthermore, we designed and implemented a data sharing management system for enterprise communities. Experimental and theoretical analysis confirms the feasibility and security of our proposed solution and system, addressing concerns about data privacy protection and ownership verification. This optimizes data sharing management, fosters a collaborative and cooperative enterprise community environment, and promotes the digital construction and development of enterprise communities.

Acknowledgements

This research is supported by the National Natural Science Foundation of 2019YFB1404603

References

- [1] Xiaoxiao S ,Yijie W ,Hujun S .Blockchain-based collaborative business process data sharing and access control[J].Journal of Reliable Intelligent Environments,2023,10(1):3-17.
- [2] Abdulaziz A ,Abdulmajeed A ,Hamad A , et al.Data Mesh Meets Blockchain[J].International Journal of Computational Intelligence Systems,2024,17(1):
- [3] DeFranco F J ,Roberts J ,Ferraiolo D , et al.An infrastructure for secure data sharing: a clinical data implementation.[J].JAMIA open,2024,7(2):ooae040-ooae040.
- [4] Henning B ,Ann T ,Patrick W , et al.Cooperative Approaches to Data Sharing and Analysis for Industrial Internet of Things Ecosystems[J].Applied Sciences,2021,11(16):7547-7547.
- [5] Nakamoto S, et al. Bit coin: A peer-to-peer electronic cash system [J]. 2008.
- [6] Geng Boyun, Yan Shuming, Wang Chao. A Review of Key Blockchain Technologies[J]. Information Systems Engineering, 2021(10): 93-97.
- [7] Wang Xiuli, Jiang Xiaozhou, Li Yang. Data Access Control and Sharing Model Using Blockchain[J]. Journal of Software, 2019, 30(6): 9.
- [8] Ma Junming. A Design Scheme to Prevent Replay Attacks[J]. Network Security Technology & Application, 2022(02): 15-17.
- [9] Hu Yang, Han Zengjie, Ye Guohua, Yao Zhiqiang. A Certificate-Free Signature Scheme to Resist DNS Man-in-the-Middle Attacks[J]. Journal of Network and Information Security, 2021, 7(06): 167-177.
- [10] Li Biaoqi, Fu Xiaodong, Yue Kun, Liu Li, Liu Lijun, Feng Yong. Preventing Sybil Attacks in P2P Reputation Systems Using Proof of Work[J]. Journal of Small & Microcomputer Systems, 2022, 43(01): 137-143.
- [11] Zhou Jian, Qu Ran. A Blockchain Private Key Management Scheme Resistant to Collusion Attacks[J]. Computer Engineering, 2020, 46(11): 23-28. DOI:10.19678/j.issn.1000-3428.0057591.