Physical Communication 58 (2023) 102048

Contents lists available at ScienceDirect

Physical Communication

journal homepage: www.elsevier.com/locate/phycom

Full length article

Blockchain aware proxy re-encryption algorithm-based data sharing scheme

Ismail Keshta [a,*], Yassine Aoudni [b], Mukta Sandhu [c], Abha Singh [d], Pardayev Abdunabi Xalikovich [e], Ali Rizwan [f], Mukesh Soni [g], Sachin Lalar [h]

[a] Computer Science and Information Systems Department, College of Applied Sciences, AlMaarefa University, Riyadh, Saudi Arabia
[b] Department of Computers and Information Technology, Faculty of sciences and arts, Turaif, Northern Border University, Arar 91431, Saudi Arabia
[c] Department of Computer Science and Engineering, SVSU, India
[d] Department of Basic Sciences, College of Sciences and Theoretical Studies, Dammam-branch, Saudi Electronic University, Riyad, Saudi Arabia
[e] Department of Scientific Research, Innovation and Training of Scientific and Pedagogical Personnel, Tashkent Institute of Finance, Tashkent, Uzbekistan
[f] Department of Industrial Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia
[g] Department of CSE, University Centre for Research & Development, Chandigarh University, Mohali, Punjab 140413, India
[h] Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, India

ARTICLE INFO

Article history:
Received 17 January 2023
Received in revised form 20 February 2023
Accepted 6 March 2023
Available online 10 March 2023

Keywords:
Blockchain
Proxy re-encryption algorithm
Data sharing scheme
Control sharing
Privacy preservation

ABSTRACT

The blockchain stores transaction data in a distributed shared global ledger. It is challenging to strike a balance between privacy protection and usefulness while sharing data. Moreover, the dynamic adjustment of blockchain data access rights is a challenging problem. To this end, this paper suggests a blockchain data-controlled sharing scheme based on proxy re-encryption. First, a proxy re-encryption algorithm is constructed based on SM2 and the blockchain. Blockchain data sharing can give businesses a secure way to store and share data. Since this network is decentralized, and data is transmitted across a peer-to-peer network under the protection of an unchangeable cryptographic signature. Blockchain makes it more difficult to alter or hack the data. The data-controlled sharing scheme uses proxy re-encryption to protect transaction data privacy and realize data security sharing. Secondly, a dynamic adjustment mechanism for user rights is proposed. Blockchain nodes divide labor and manage re-encryption key parameters separately to achieve user access rights determinism Update, the visibility of transaction data is dynamically adjusted. Finally, the performance and security evaluation demonstrate that this scheme can realize the dynamic sharing of blockchain data while protecting transaction privacy and has advantages in computing overhead, which is better applicable to the Controlled sharing of blockchain data. This research suggests a regulated blockchain-based data-sharing system that makes use of proxy re-encryption. They are developing a proxy re-encryption algorithm with SM2 in order to fully safeguard the privacy of transaction data and to achieve data access authority determination by controlling proxy re-encryption key parameters. It is suggested to employ a hybrid attribute-based proxy re-encryption method that enables the proxy server to change attribute-encrypted cypher texts into identity-based encrypted cypher texts so that users with limited resources can access the previously encrypted material.

1. Introduction

Realizing data security sharing in a distributed environment has always been a research hotspot. In 2008, the author, the pseudonym "Satoshi Nakamoto", published an article, "Bitcoin: A Peer-to-Peer Electronic Cash System" [1] district. Blockchain technology, which forms the backbone of Bitcoin, has seen widespread use. Blockchain is a distributed ledger technology maintained collaboratively by numerous parties. It is characterized by qualities of decentralization, traceability and non-tampering, making the application of Utilizing blockchain technology to encrypted currency as well as provides platform support for data security sharing. However, since the transaction information on the blockchain is open and transparent to all nodes in the network, each node can read the transaction data and the attacker through the blockchain ledger. The data analysis on the web poses a

* Corresponding author.
E-mail addresses: imohamed@mcst.edu.sa (I. Keshta), yassine.aoudni@nbu.edu.sa (Y. Aoudni), mukta.sandhu@gmail.com (M. Sandhu), asingh@sdataeu.edu.sa (A. Singh), pardayev_abdunabi@tfi.uz (P.A. Xalikovich), arkhan71@kau.edu.sa (A. Rizwan), mukesh.research24@gmail.com (M. Soni), sachin509@gmail.com (S. Lalar).

https://doi.org/10.1016/j.phycom.2023.102048
1874-4907/

threat to the user's transaction privacy and identity privacy [2]. Authors [3] once proposed a digital currency, Monero, based on ring signatures, using ring signatures to hide transaction amounts. Decentralization, traceability, and non-tampering are its defining characteristics, enabling the use of blockchain technology for encrypted currency and serving as a platform for data security sharing. However, each node in the network can access the transaction data and the attacker through the blockchain ledger since the transaction information on the blockchain is transparent and open to all nodes in the network. A digital currency based on ring signatures that uses ring signatures to mask transaction amounts has been developed, but data analysis on the web poses a threat to the user's transaction privacy and identity privacy. Still, the signature technology in the signing process, other users are required to complete the signature together, and the transaction information is easily leaked. The hidden transaction has been broken by tracking the private key of the user's signature. The privacy considerations of user identity information and transaction data on the blockchain have become increasingly popular among researchers. However, the importance of transaction data security sharing is facing significant challenges.

The privacy protection of user identity information and transaction data is a significant problem in the security of blockchain technology. Blockchains are the source of Bitcoin and other cryptocurrencies, and they contributed to the rise of distributed ledger technology (DLT). Public blockchains also help to solve several issues and challenges, such as centralization and security flaws. Blockchain creates new cost-saving efficiency while enhancing the provenance, safety, reliability, and openness of data shared throughout a business network. Authors [4] presented a blockchain-based anonymous electronic cash system on Bitcoin - Zerocoin, through zero-knowledge proof and RSA accumulator and other cryptography technologies to hide the address of the trader and interrupt the association between transactions, making the transaction untraceable, not disclosing the relevant information of the transaction, and providing high privacy protection for the blockchain. Still, the algorithm proves that the process is very slow and not very practical. Authors [5] first provided blockchain transaction privacy and programmability methods. A fundamental issue with the security of blockchain technology is the privacy protection of user identifying information and transaction data. A blockchain-based anonymous electronic cash system based on Bitcoin called uses zero-knowledge proof, the RSA accumulator, and other cryptography technologies to conceal the trader's address and prevent associations between transactions, rendering the transaction untraceable, withholding the pertinent transactional information, and offering the blockchain with high privacy protection. However, the technique, which also offers blockchain transaction privacy and programmability methods, shows that the process is extremely slow and not very practical. Based on smart contracts and zero coins, users send encrypted information to intelligent contracts; However, smart contracts results can be publicly verified, and the order of all transaction actions in the contract must be kept secret from the public. Authors [6] implemented a system for managing access policies based on the Bitcoin platform for access control and implement user transactions through transactions. The transfer of permissions between the access control strategy and permission conversion is publicly visible on the blockchain. Subsequently, Authors [7] improved the scheme by using smart contracts to achieve access control. Authors [8] passed the same State-of-the-art encryption technology, and blockchain intelligent contracts protect users' privacy. Still, only the transaction information is visible to both parties, which is not conducive to the efficient sharing of data by users. Authors [9] proposed a method using blockchain and certificates. The distributed data storage scheme of encryption technology uses blockchain miners to eliminate traditional centralized servers and miners use credentials cryptography to record audits. To transform data into ciphertext, encryption methods are utilized. The encrypted data appears random because the encryption key causes data to be altered in a predictable way. There are a variety of data encryption methods available. The three main types of encryptions are symmetric, asymmetric, and hashing, according to the majority of internet security (IS) experts. Authors [10] proposed a decentralized data-sharing model. Improve the transaction record format and consensus mechanism, and ensure data privacy with the help of secure multi-party computing and differential privacy technology. Authors [11] proposed a security based electronic medical records sharing concept, which uses data desensitization technology to sacrifice some data accuracy to avoid the problem of transaction privacy leakage. The above schemes mainly protect identity and transaction privacy through cryptographic technologies such as digital signatures, zero-knowledge proofs, and homomorphic encryption. However, blockchain data sharing is limited to pre-set data users, it is difficult to achieve efficient expansion, the availability is low, and it cannot meet the actual needs of data sharing. The encryption algorithm understands that sharing transaction data is safe and private. In order to allow for the dynamic adjustment of transaction data visibility, the update of data access authority is accomplished for transaction information access through the split management of agent re-encryption essential parameters. The solution can enable dynamic sharing of transaction data while safeguarding user privacy, according to security and performance studies. The data-controlled sharing method employs proxy re-encryption to ensure data security sharing while safeguarding transaction data privacy. A dynamic adjustment mechanism for user privileges is the second suggestion. Blockchain nodes divide labor and independently manage re-encryption key parameters to guarantee user access rights determinism. The visibility of the transaction data is dynamically adjusted. Bitcoin's foundational technology, blockchain, has been widely adopted. Blockchain is a distributed ledger technology that is jointly updated by many participants. It is distinguished by attributes including decentralization, traceability, and immutability, allowing the use of blockchain technology to encrypted cash as well as providing a platform for data security sharing. But, because the transaction data on the blockchain is visible to all nodes in the network and open, each node can access the transaction data and the attacker through the blockchain ledger. The user's transaction privacy and identity privacy are at risk from online data analysis.

Aiming to realize the safe sharing of transaction data, Authors [12] suggested a blockchain-based method for sharing information and traceability scheme, which adopts a double-chain structure to realize data secure sharing and traceability of data sources. This is due to the fact that one chain stores original data, whereas the other chain stores duplicate data is used to store transaction data. Authors [13] proposed an encryption scheme based on traceable attributes, using blockchain technology to maintain data integrity and non-repudiation. Pre-encryption technology enables the rapid generation of the Ciphertext. Following that, the Bloom attribute filter is used to send the hidden strategy to the blockchain. The blockchain features a data tracking algorithm. and created a block using the algorithm for the policy change. Structure, to understand the dynamic updating of the block content's visibility, and to discover the active sharing of traceability information while safeguarding transaction privacy. The Ciphertext is quickly generated through pre-encryption technology. Then, the hidden strategy is sent to the blockchain using the Bloom attribute filter. Authors [14] proposed a Based-on attribute encryption, the blockchain has an algorithm for tracking

data. and designed a block based on the policy update algorithm. Structure, to realize the dynamic update of the visibility of the block content, and to learn the active sharing of traceability information while protecting transaction privacy. However, when the attributes on the block are updated, the data traceability complexity will increase. Authors [15] will divide Combining layer attribute ciphering with linear secret sharing is proposed with mechanism for controlling data privacy access based on accessible attribute encrypted. To ensure data security sharing and protect the privacy of transaction data, the data-controlled sharing method employs proxy re-encryption. Second, a dynamic user rights adjustment system is suggested. To achieve determinism of user access rights, blockchain nodes split work and independently maintain re-encryption key parameters. Update, transaction data visibility is dynamically changed. The verification node implements the user's access control, which avoids the need to submit Private keys and network access mechanisms for the blockchain. Authors [16] suggested a blockchain-based strategy for managing and distributing data access, leveraging attribute-based encryption to control and distribute enterprise data and realize fine-grained access control and secure data sharing. In the above scheme, when the access rights of blockchain transaction data are changed, the data needs to be repeatedly encrypted. Therefore, efficient and safe sharing of blockchain data has become a serious issue needing resolve.

By using proxy re-encryption technology, a partially trusted agent can change a ciphertext that only one user can decrypt into one that only another user can decrypt that another user with the same plaintext can solve. There is no need to re-encrypt data when authorization is changed, and the user only needs to calculate the re-encryption key. Data sharing can be completed, which has application value in the blockchain environment. At present, proxy re-encryption technology is widely used in data sharing. Authors [17] proposed a deterministic update scheme for cloud data access authorization based on proxy re-encryption, managing the re-encryption key separately and realizing the deterministic update of the key when the authorization changes. Subsequently, Authors [18] proposed a trusted authorization method based on proxy re-encryption for IoT cloud nodes. A proxy server, usually referred to as a ''proxy'' or a ''application-level gateway'', is a computer that serves as a bridge between a local network (for instance, all the computers at one business or in one building) and a wider network like the internet. Proxy servers improve security and performance. Authors [19] realized the controlled sharing of Using identity proxy re-encryption, we can keep our social cloud data safe. Authors [20] To achieve efficient data sharing, A hybrid attribute-based proxy re-encryption scheme is proposed, which allows the proxy server to convert attribute-encrypted cipher texts into identity-based encrypted Cipher text so that resource-constrained users can effectively access the previously encrypted data. Authors [21] proposed a cloud computing joint security data sharing and query framework, adopting a combination the combination of encryption algorithms and proxy re-encryption to prevent. The revoked user leaks unauthorized data when rejoining the system. This scheme assumes that the two servers do not collude to achieve data security sharing. The above method mainly uses proxy re-encryption technology in the cloud environment or big data platform. It is unsuitable for efficient and stable user privacy data sharing on the blockchain.

Due to the foregoing evaluation, this paper recommends a controlled a blockchain-based data-sharing mechanism that uses proxy re-encryption. They are using SM2 to construct a proxy re-encryption algorithm to complete transaction data privacy protection and to realize data access authority determination through managing proxy re-encryption key parameters. The following are the study's main contributions:

(1) Construct a proxy re-encryption algorithm suitable for the controlled sharing of blockchain data. Design a proxy re-encryption algorithm based on the SM2 encryption algorithm to achieve controlled sharing of blockchain data while protecting transaction privacy—a balance between privacy protection and usability.

(2) Propose a dynamic adjustment mechanism for user permissions for transaction data sharing. Blockchain miner nodes divide the work into agents and manage the proxy re-encryption key parameters separately. Authorized management nodes verify user data access permissions to achieve deterministic updates of user access permissions, allowing the visibility of transaction data to be dynamic adjustment, and there is no need to re-encrypt data when authorization changes.

(3) The performance analysis and security analysis establish that the proposal provided in this research may not only implement dynamic data sharing, but also improve performance. while protecting transaction privacy but also better adapt to the data access in the open environment of the blockchain network in terms of functionality and computational overhead—control sharing. Healthcare professionals and researchers have paid close attention to the security and privacy of EHRs. Many encryption and decryption techniques, as well as key management systems, have been created to ensure security. However, additional security measures we will propose as a result of sharing and scalability problems. However, these technologies also lead to other challenges, such efficiency problems. To reduce computational overhead, blockchain-based EHR administration systems will be developed. EHRs may, however, be leaked to the company because the majority of blockchain systems are deployed by outsourcing firms.

2. Prepare knowledge

2.1. Blockchain

Blockchain is a special data format generated by merging data blocks in chains in sequential sequence based on a verifiable and trustworthy consensus method in a peer-to-peer network setting. It cryptographically confirms that its data cannot be altered with, Unforgeable, traceable, decentralized, and trustless distributed shared ledger system. After the transactions within a certain period are verified, the blocks are formed and linked to the blockchain database with time stamps. The number of blocks continues to increase, creating a data storage structure from the first block to the most recent block, and the data can be traced back by time. In comparison to earlier study plans, the computing efficiency has also increased. Security, functionality, and computing overhead are all areas where it excels. The privacy protection and usability of blockchain data sharing are effectively balanced. The alliance chain's partial decentralization and transaction data privacy are both addressed by this solution. It applies to applications where private information, legal papers, electronic medical records, and other sensitive data are shared utilizing blockchain-distributed databases. As a result, the study that comes after will focus on creating an effective and reasonable data-sharing system that uses blockchain technology in particular application situations.

Blockchains are classified into three types: public, alliance, and private [22]. Among them, the public chain is an entirely decentralized, permissionless blockchain which can be joined by any organization or individual. For example, Bitcoin and Ethereum

are public. The consortium chain is a partially decentralized permission chain, which is usually jointly participated by multiple institutions. The joining of nodes requires the consent of other consortium members. The blockchain data-controlled sharing scheme proposed in this paper applies to consortium chains; private chains are usually used for the organization controls the internal system of a unit or organization and the read and write authorization of its data.

2.2. Proxy re-encryption technology

Proxy Re-Encryption (PRE) supports the transfer of decryption authority based on public key encryption. It was first proposed by Authors [23] at the European Cryptography Annual Conference in 1998. The agent of the letter (proxy) converts the cipher text that Alice can decrypt into the same plaintext that Bob can solve, and the semi-trusted agent cannot receive any information regarding the data's plaintext. Proxy re-encryption puts the encryption and decryption work in data sharing. During this process, the user completes the first encryption, and the agent re-encrypts different shared users based on the first cipher text. The data owner does not need to repeat the encryption operation when sharing data, and the encryption work is handed over to the proxy server, which reduces the Workload. According to the conversion direction of the ciphertext, One-way and two-way proxy re-encryption are two types of proxy re-encryption. One-way proxy re-encryption can only realize the ciphertext conversion from Alice to Bob.

Two-way proxy re-encryption, on the other hand, may recognize the decryption conversion from Alice to Bob and discover the decryption translation from Bob to Alice. This work presents a system for one-way proxy re-encryption. At present time, some scholars have proposed the PRE scheme based on keyword search [24]; the identity-based PRE scheme [25] takes identity, attributes, and key types for fine-grained management as essential parameters of PRE keys [17], which can provide a necessary basis for the research of ciphertext access control. Several data-sharing designs have been proposed for the ecosystem of data-sharing depending on 3rd providers, including cloud computing and edge computing [26]. Massive data mutuality between the cloud and IoT is addressed by the cloud-assisted IoT [27] architecture, which adopts a conditional identity-based broadcast proxy re-encryption method (CIBPRE) to provide secure data exchange. Users can collect, store, and distribute IoT data in a confidential manner using this framework. An ABKS approach is proposed by Yin et al. [28] to enable precise and effective data access control over encrypted data. These gadgets create enormous amounts of data every day around the globe [29]. The fundamental idea behind Internet of Things technology is data, and this data have immeasurable worth for a variety of applications. IoT advancements have created new security and privacy concerns, despite the fact that they initially appear to be highly appealing. As a result, it is crucial to confirm the dependability and safety of IoT data sharing [30].

2.3. SM2 encryption algorithm

The provable security of the national secret algorithm SM2 has reached the highest security level of the cryptographic algorithm, and its implementation efficiency is equivalent to or slightly higher than the international standard cryptographic algorithm. The international standard cryptographic algorithm's implementation efficiency is equal to or slightly higher than the national secret algorithm SM2, whose proved security has advanced to the highest security level of any cryptographic method. Security, stability, and great efficiency are benefits of the universal common encryption algorithm, or ECC. The SM2 Encryption algorithm and proxy re-encryption technology manages the generation parameters of the re-encryption key under the deterministic update of the re-encryption key, dynamically update the access rights of the blockchain data, and achieve controlled sharing of the blockchain data. Finally, we will evaluate this scheme's ability to enable ciphertext data access control by comparing it to other research techniques now in use. The global common encryption algorithm ECC has the advantages of security, stability, and high efficiency. The SM2 algorithm is made up of the various algorithms.

(1) Initialization. Given the security parameter κ, generate elliptic curve parameters params = {p, q, E, G}, where p is a large prime number, representing the scale of the finite field, and E represents the elliptic curve defined on the limited field F_p, G represents the generator point of elliptic curve E with order q.

(2) Key generation. The user randomly selects d ϵ [1, q - 1], input the system public parameters params, calculates P = dG, private key d, and publicizes, public keyP.

(3) Encryption algorithm. Given the parameter params, the bit string M of the message, and the length is klen, use the public key P to encrypt according to the following steps:
(a) Utilize a random generator used to produce an arbitrary number k ϵ [1, q - 1];
(b) Compute the elliptic curve point $C_1 = [\ k\ ]G = (x_1, y_1)$, and translate the data type of C_1 into a bit string;
(c) Compute [k] P = (x2, y2);
(d) Compute t = KDF(x_2|| y_2, klen);
(e) Compute M⊕t=C2;
(f) Compute C3 using Hash (x_2|| M| | y_2);
(g) Ciphertext C = (C1, C2, C3) is return as output.

(4) Decryption algorithm. Given the parameter paramswith the private key d: the ciphertext C = (C1, C2, C3) decrypt it
(a) Compute [d]C_1 = (x2, y2);
(b) Compute t = KDF(x_2|| y_2, klen);
(c) Compute M = C_2⊕t;
(d) Compute $C_3' =$ Hash (x_2|| M| | y_2), if $C_3' = C3$, output M.

3. System design

3.1. System model

The blockchain data-controlled sharing scheme based on proxy re-encryption includes 4- types of participating entities, namely data owners, data users, authorized managers, and miner nodes maintaining the blockchain. The model of the system is depicted in Fig. 1, including The entities and their functions described below. Distribution and control of data access rights using the blockchain consensus process, finish the node's registration, and safeguard the key. Check the user's access privileges in accordance with the list of authorizations provided by the data owner, and then send the results of the verification to the other miners.

(1) Data owner. Encrypt the shared transaction data to generate the initial ciphertext, specify the access rights of the data, decide the revocation and re-entry of user permissions, construct the proxy re-encryption key, and attach the initial ciphertext. Proxy re-encryption key transactions are broadcast to the blockchain network. The data owner can be a miner or a user who conducts transactions on the blockchain.

(2) Data users. Request access to transaction data on the blockchain and use its private key and decryption parameters to decrypt the re-encrypted ciphertext to obtain shared data.

(3) Authorization manager. To realize the decentralization of the entire system, the authorized manager designated in the

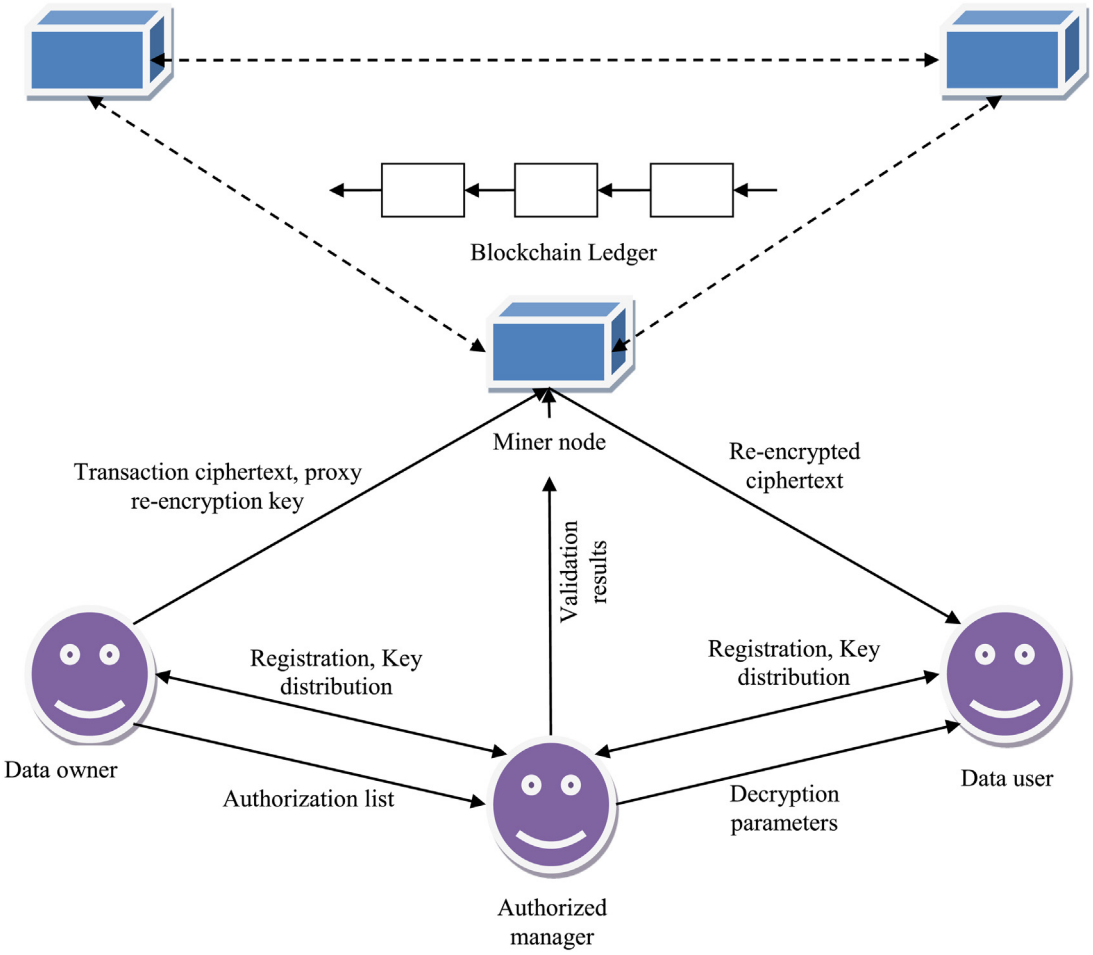


Fig. 1. System Model.

blockchain system can be an authorized management node selected through the blockchain consensus mechanism, complete the node's registration, and secure the key—distribution and management of data access rights. Data that has been stored cannot be modified or removed as long as blockchain is in use. The blockchain thereby prevents data leakage across the whole network. Each network user can verify any data were entered into the blockchain because of its decentralized structure. Blockchain costs more money than a traditional database does. As well, companies must correctly plan and implement the blockchain connection into their workflow. Blockchain technology makes it impossible to simply update data that has already stored; to do so, all of the blocks' codes must be rebuilt, which requires time and money. Verify the user's access rights according to the authorization list given by the data owner, and send the verification results to the rest of the miners. Send decryption parameters to legitimate data users, interact with the data owner, and implement through the management of the authorization list Updates to blockchain data access permissions.

(4) Miner nodes that maintain the blockchain. For legal users, The proxy re-encryption key supplied by the data owner is used to re-encrypt the initial transaction ciphertext, and the re-encrypted decryption of the transaction data is transmitted to the data user. Illegal users will reject user requests. Broadcast the data transaction records within a period, and other nodes will verify the blocks and add them to the blockchain ledger.

The authorized manager is one of them who may be trusted. The miners in the maintenance blockchain system are "honest and curious", yet there is a chance for miner node attacks in this plan. Those who desire can do so thanks to the blockchain's incentive system. After getting the transaction ciphertext and proxy re-encryption key from the blockchain network, the rewarded nodes will finish proxy re-encryption operation at the same time, may engage in cryptanalysis and collusion assaults, exposing user's information.

3.2. Model of security design

Security Design model is as follows:

Attacker A can interrogate the process of essential generation, decryption, proxy re-encryption key generation, re-encryption, etc.

Initialization: Challenger C selects parameters and generates initial system coefficients pp.

Phase 1: Attacker A asks for any process of KeyGen, ReKeyPara, ReKeyGen, ReEncrypt, and Decrypt. The key is used when asking ReKeyPara, ReKeyGen, and ReEncrypt; KeyGen generates Decrypt.

Challenge: After the attacker A completes the phase 1 inquiry, he outputs the equal-length plaintext $(m_0, m_1) \epsilon$ M, the decryption parameter α^*, the re-encryption parameter β^* generated by ReKeyPara and the public key pk* of the attacked target, where KeyGen generates pk*, and the private key has not been leaked. When attacker A asks the ReKeyGen function with $(\beta^*, \beta', \alpha^*)$, the private key corresponding to β' is kept secret, the challenger C chooses b $\epsilon\{0, 1\}$ is used as random bits to calculate the ciphertext $C_b = \text{Encrypt}(m_b, \text{pk}^*)$ for challenge inquiry.

Phase 2: Attacker A continues the query in phase 1 while satisfying the following conditions.

(1) When attacker A inquires ReKeyGen with (β^*, β', α^*), the private key of β' is kept secret;

(2) When A inquires ReEncrypt with (C_b, β^*, β', α^*), the private key of pk ′ is kept secret;

(3) When A queries ReKeyGen with (β^*, β', α^*), C_b' cannot be used to query Decrypt, where C_b' is ReEncrypt(C_b, β^*, β', α^*) valid output.

Guess: Attacker A guesses b′ ϵ {0, 1}, if b′= b, the challenge is successful.

If the advantage of attacker A to win the above challenge is defined as ε, then ε =| Pr[b'=b]−1/2|, The challenge scheme is ciphertext secure (CCA).

4. Blockchain data controlled sharing scheme

4.1. Program overview

First, the overall flow chart of the scheme in this paper is given, as shown in Fig. 2.

The scheme in this paper includes four stages: system establishment, uploading transaction information, data access, and user authorization update stage.

(1) System establishment phase. Initialize system parameters and generate public–private key pairs.

(2) Upload transaction information stage. The data owner encodes the transaction data with its public key to generate the initial transaction ciphertext, divides the transaction data access rights, constructs the proxy re-encryption key, proxy re-encryption key to the blockchain network, and broadcasts the initial transaction ciphertext and proxy re-encryption key to the blockchain network and sends the authorization list at the same time authorized managers for the blockchain system.

(3) Data access stage. The user requests data access from the blockchain and authorizes the management node to verify the user's access rights. If the user is legitimate, the blockchain miner node uses the proxy re-encryption key uploaded by the data owner to encrypt the initial transaction encryption. Then, Text to get the clear text of the transaction data. If it is an illegal user, the user request will be rejected.

(4) User authorization update stage. The data owner interacts with the blockchain system authorization manager, completes the update of transaction data access rights by updating the authorization list, and realizes the revocation or rejoin of data users so that the blockchain data Visibility is adjusted dynamically.

4.2. Scheme construction

Phase 1: System Establishment

Contains two steps system initialization and key generation.

(1) System initialization: Setup (κ) →pp

The security parameter κ, get the prime numbers p, q, E, and G of κbit. p represents the size of the finite field, E represents the elliptic curve defined on the finite field F_p, define P as a point on the elliptic curve E, and set it As the generator of the group G, G is a cyclic group of order q. Define the hash function H_1: {0, 1}*→ {0, 1}^{l}, H_2: G →Z_q^*, H_3: {0,1}*→ G, H_4: {0,1}*→ G. Publish parameter pp ={ H1, H2, H3, H4,p,q,E,G,P}.

(2) Key generation: KeyGen (pp) → (sk_A, pk_A) input system public parameter pp, select random number $x \epsilon Z_q^*$, private key sk_A = x, public key pk_A = xP.

Phase 2: Upload transaction information

Contains initial encryption of transaction information, generation of proxy re-encryption key parameters, construction of proxy re-encryption key and authorization list.

The data owner encrypts the transaction data with its public key, generates the initial ciphertext, creates the proxy re-encryption key rkA → B and the authorization list L, and broadcasts the transaction ciphertext and proxy re-encryption key rkA → B to the blockchain network, where the transaction is verified by the blockchain's miner nodes. Send the authorization list L to the blockchain system's authorized manager, and update the access authority of the blockchain data through the management of L. The transaction data records are initially encrypted by the data owner, and the generated proxy re-encryption key is calculated as follows.

(1) Initial encryption: Encrypt(M, pk_A) → C. The data owner uses his public key pk_A to encrypt the message M, the length of M is l, select i ϵ G, and the encryption operation is as follows:

$$r = H_2(i) \tag{1}$$

$$C_1 = rP = (x_0, y_0) \tag{2}$$

$$rpk_A = (x_A, y_A) \tag{3}$$

$$t = H_1(x_A \parallel y_A) \tag{4}$$

$$C_2 = M \oplus t \tag{5}$$

$$C_3 = H_3(x_A \| M \| y_A) \tag{6}$$

$$C_4 = H_4(M \| C_1 \| C_3) \tag{7}$$

$$C = (C_1, C_2, C_3, C_4) \tag{8}$$

The transaction data is originally encrypted and posted to the blockchain for broadcasting before being verified by miners. Then, the data owner can access the uploaded transaction data and decrypt the transaction ciphertext with its private key. The decryption operation is as follows:

$$S = sk_A C_1 = (x_A \parallel y_A) \tag{9}$$

$$t = H_1(x_A \parallel y_A) \tag{10}$$

$$M = C_2 \oplus t \tag{11}$$

$$C_3' = H_3(x_A \| M \| y_A) \tag{12}$$

If $C_3' = C3$, the data plaintext M is obtained.

(2) Proxy re-encryption key parameter generation: RekeyPara (r, pk_A, pk_B) → β. The data owner constructs a proxy re-encryption key parameter for data user B, that is, β = {rpk_A, rpk_B}.

(3) Proxy re-encryption key generation: RekeyGen (α, β) → $rk_{A\to B}$. The data owner calculates the proxy re-encryption for data user B through the proxy re-encryption key parameter β and the authorization parameter α defined by itself. The key $rk_{A\to B}$ is uploaded to the blockchain network, namely

$$rk_{A\to B} = H_1(rpk_A) \oplus H_1(rpk_B \parallel \alpha) \tag{13}$$

Phase 3: Data Access

The data user sends a data request to the blockchain, and the authorization management node checks whether the requesting user has access rights according to the authorization list L uploaded by the data owner. If the user is illegal, the user request is rejected. After the transaction ciphertext is re-encrypted, the re-encrypted ciphertext is sent to the requesting user.

(1) Proxy re-encryption: ReEncrypt(C, $rk_{A\to B}$) → C′. For legal users, the proxy re-encryption calculation of transaction ciphertext C by miner nodes on the blockchain system is as follows:

$$C_1' = C_1 \tag{14}$$

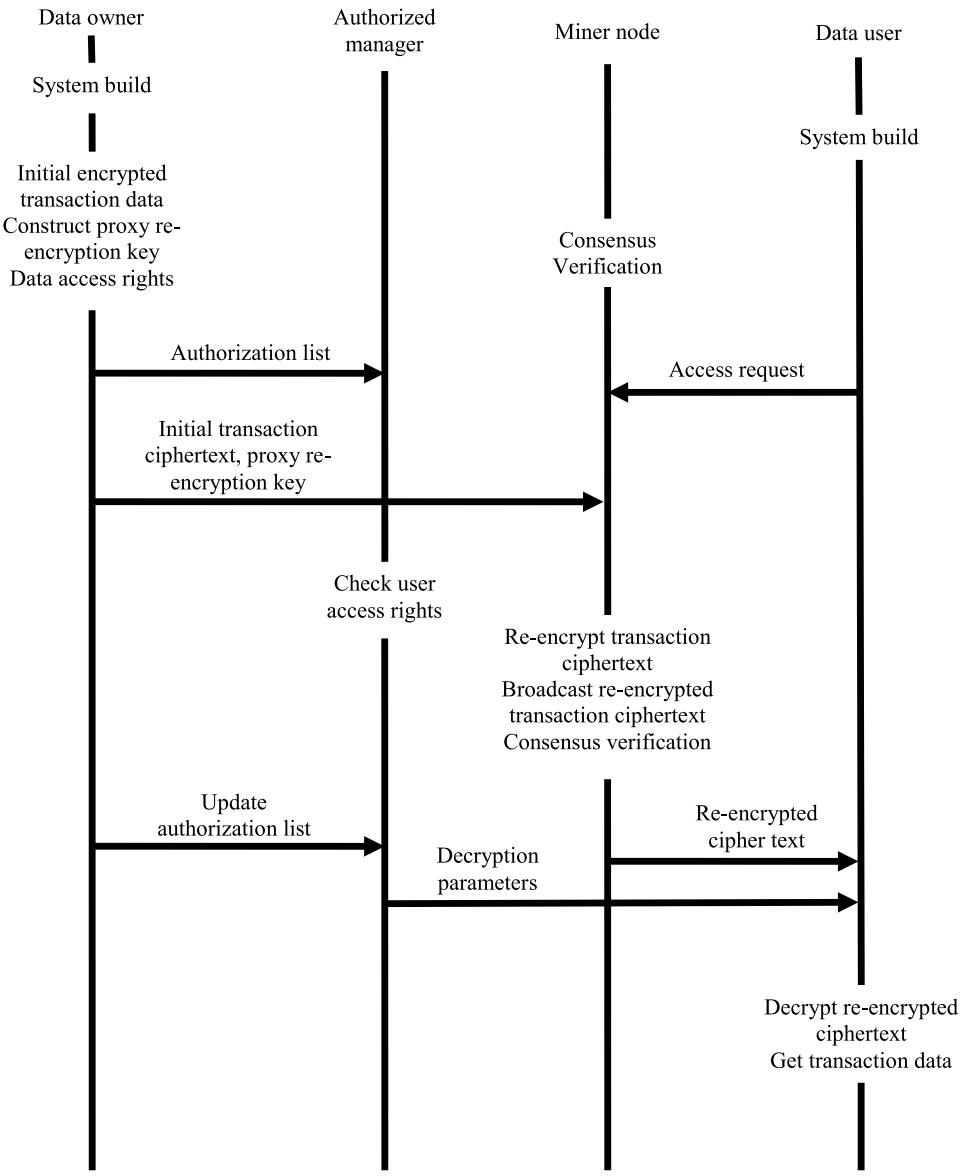


Fig. 2. Scheme flow chart.

$$C_2' = rk_{A\to B} \oplus C_2 \quad (15)$$

$$C_3' = C_3 \quad (16)$$

$$C_4' = C_4 \quad (17)$$

$$C' = (C_1', C_2', C_3', C_4') \quad (18)$$

(2) Decryption: Decrypt(sk_B, C′, α)→ M. After the legal user obtains the re-encrypted ciphertext from the blockchain, he can use his private key and the decryption parameter α sent by the blockchain authorized manager to decrypt and get for transaction data, the user interprets and re-encrypts the ciphertext and calculates it as follows:

$$M' = C_2' \oplus H_1(sk_B C_1' \parallel \alpha) \quad (19)$$

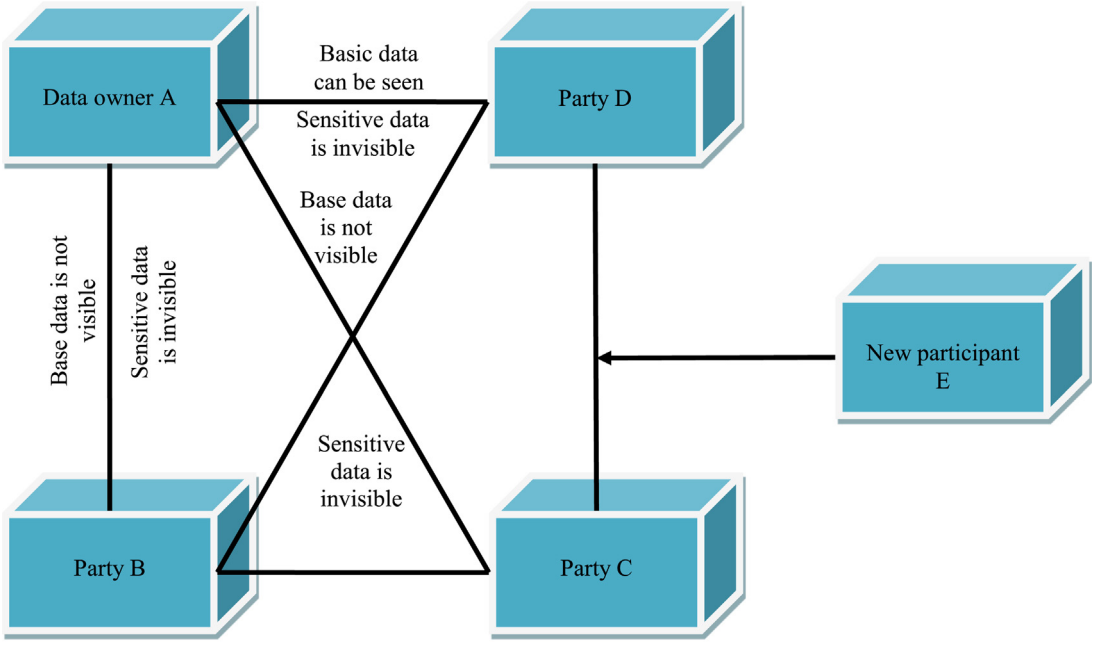
$$k = H_4(M' \| C_1' \| C_3') \quad (20)$$

If k $= C_4'$, then M = M', the legitimate user gets the plaintext of the transaction data.

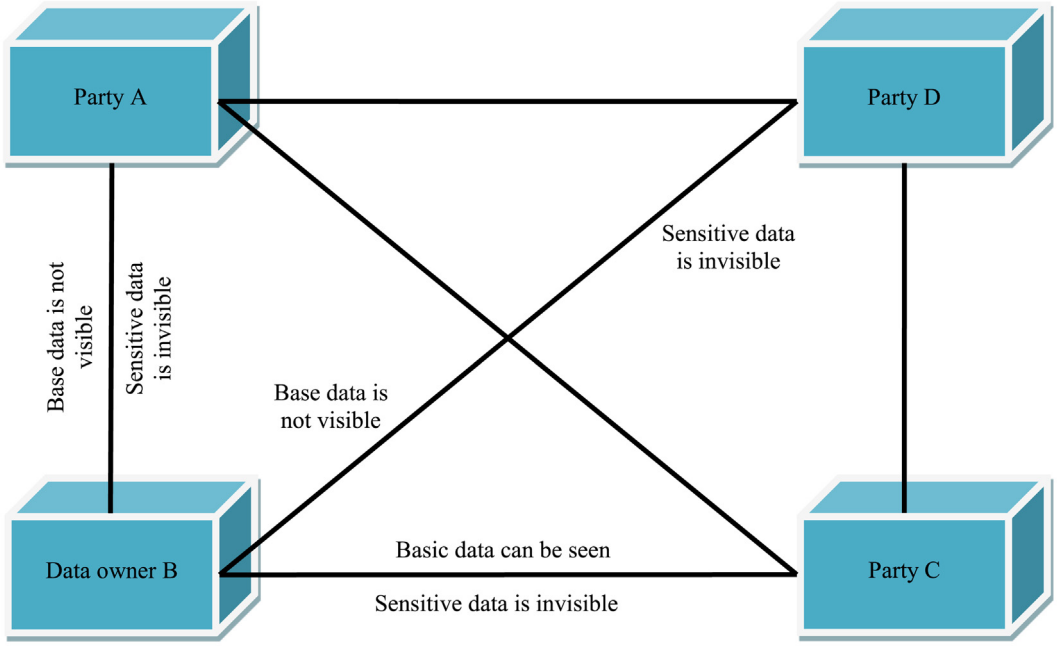
Phase 4: User Authorization Update

The authorization manager on the blockchain stores the authorization list L and updates the data access rights according to the requirements of the data owner. The miner node on the blockchain stores the original transaction ciphertext and the proxy re-encryption key. According to the verification of the authorized management node, The result determines whether to re-encrypt the transaction ciphertext and share it with the data users. Finally, the data owner interacts with the blockchain authorization manager to revoke the data requester's access to the data records by deleting the authorization list L and granting New access rights to data requesters.

To describe the privacy protection requirements of blockchain transaction data sharing further specifically, this paper abstracts a blockchain system data sharing platform, as shown in Fig. 3, and initializes four participants, A, B, C, and D. Define the privacy of transaction data. The protection requirements are shown in Table 1. The data owner can specify the transaction data because the visibility of transaction data should be chosen in accordance with the individual business needs of the data owner. The data owner may identify transaction data as primary and sensitive



(A)Participant A shares data



(B)Participant B shares data

Fig. 3. Blockchain data-sharing platform.

data since the visibility of transaction data should be chosen in accordance with the specific business needs of the data owner. The transaction initiator can only describe the transaction data. It is visible to authorized users, and unauthorized participants can only see the ciphertext of transaction data. For example, the primary data in the transaction initiated by participant A is visible to itself and authorized users B and D, but only participant C is invisible. On the other hand, the sensitive data in the transaction initiated by participant A can only be seen by participant A and authorized user B, and neither participant C nor D can access it; only ciphertext and hash values can be seen. Data owner and blockchain system authorization management The user interacts updates the authorization list L, and dynamically adjusts the visibility of transaction data. The specific steps are as follows.

(1) The data owner A wants to authorize the sensitive data in the shared transaction data to the participant D, who initially has no access rights, then the data owner A interacts with the authorized manager of the blockchain system, updates the

Table 1
Transaction data privacy protection requirements table.

Data ownership By	Transaction data type	Data user			
		Party A	Party B	Party C	Party D
Party A	Basic data	V	V	I	V
	Sensitive data	V	V	I	I
Party B	Basic data	V	V	V	I
	Sensitive data	I	V	V	I

V: visible, I: invisible

authorization list L, and adds Authorization parameter of user D, and construct the corresponding agent at the same time the re-encryption key $rk_{A\to D}$ is broadcast to the blockchain network.

(2) Data owner A wants to authorize the sharing of transaction data to new participant E. Data owner A interacts with the authorized manager of the blockchain system, updates the authorization list L, and adds authorization parameters for data user E, and at the same time Construct the corresponding proxy re-encryption key $rk_{A\to E}$ and broadcast it to the blockchain network.

(3) If the data owner B wants to revoke the access rights of the original legitimate data user C, then B interacts with the authorization manager to delete the corresponding authorization list parameters.

In the above authorization update process, after the data owner initially encrypts the transaction data, the subsequent update of the access authority does not need to encrypt the data repeatedly. It only needs to define a new authorization list, send it to the authorization manager, and construct the corresponding proxy re-encryption. Finally, the encryption key is broadcast to the blockchain network.

5. Scheme analyses

5.1. Correctness analysis

Initial ciphertext $C_1 = rG = (x_0, y_0)$, $C_3 = H3(x_A||\ M|\ |\ y_A)$, $C_4 = H4(M|\ |\ C_1||\ C_3)$, after re-encryption calculation, $C_1' = C1$, $C_3' = C3$, $C_4' = C4$. When decrypting, we get M′, $k = H_4(M'||C_1'||C_3')$, so as long as $k = C_4'$ is verified, then there is M = M′. Therefore, whether the decryption result is correct can be judged by verifying whether k is equal to C_4'.

If the calculation process is correct in the following analysis, M = M′.

$$M' = C_2' \oplus H_1(sk_B C_1'\|\alpha) = rk_{A\to B} \oplus C_2 \oplus H_1(sk_B C_1'\|\alpha)$$
$$= H_1(rpk_A) \oplus H_1(rpk_B\|\alpha) \oplus C_2 \oplus H_1(sk_B C_1'\|\alpha) \quad (21)$$

Among them, $sk_B C_1' = sk_B C_1 = sk_B(rP) = r(sk_B P) = rpk_B$, then $H_1(sk_B C_1'|\,|\alpha) = H_1(rpk_B\ \|\ \alpha)$, then we have

$$M' = H_1(rpk_A) \oplus C_2 = H_1(rpk_A) \oplus M \oplus H_1(rpk_A) = M \quad (22)$$

Therefore, under the condition that the calculation process is correct, there is M = M′, so the scheme in this paper is right.

5.2. Safety analysis

Definition 1. DDH complexity assumption: For any a,b ϵ Z_q^*, given a set of elements P, aP, bP, $T \epsilon$ G on a cyclic group G, it is not easy to judge whether the equation $T = abP$ holds.

Theorem 1. *Under the random oracle model, the proxy re-encryption mechanism given in this paper is CCA secure if the DDH complexity condition holds true for group G.*

There are four hash calculations in the proof algorithm. The following analysis process simulates them as four different random oracles. The input value space of the hash calculation process in these four calculations is extra and can be automatically distinguished into various hash operations. The four hash operations in the algorithm can use the secure hash algorithm SM3 to replace the random oracle. For example, to prove Theorem 1 is to verify that attacker A challenges with the advantage ε, then $\varepsilon = |\ Pr[\ b' = b] - 1/2\ |$ It can be ignored. Define the challenge game G_i (i = 0,1,2,…,5), and the challenger C defines T_i to represent the event that b′= b in G_i.

(1) G_0: Challenger C truthfully answers the random inquiry of attacker A and, at the same time, initializes H_i^{list} (i = 1,2,3,4), let $\delta_0 = Pr[\ b' = b]$, then $|\ \delta_0 - 1/2| = \varepsilon$.

(2) G_1: Challenger C plays the game with G_0, except for the following content. Challenger C randomly selects $\tau \epsilon$ {1,2,…, p + 1} and asks H_1 for τ times. When C receives the attacker After A's challenge if the attacker A inquires about H_1, the challenger terminates the game, the probability of success of the challenger C is at least 1/(p + 1), and in G_1, $\delta_1 = Pr[\ b' = b]$, then $Pr[T_1\ ] = \delta_1/(p + 1)$.

(3) G_2: Challenger C plays the game with G_1, except H_i collides; since the hash function is a standard random process, $|\ Pr[T_1\ ] - Pr[T_2\ ]||$ can be ignored.

(4) G3: Challenger C plays the game with G2; the only difference is when Decrypt is called if the input is (C, β^*, α^*), attacker A does not use ($\beta^*|\ |\alpha^*$) query to H_1, the challenger terminates the game, otherwise the challenger C returns the decryption result to the attacker A. Since the encryption and decryption algorithm process is determined, and the hash function used is a random process, $|\ Pr[T_2\ ] - Pr[T_3\ ]||$ can be neglected.

(5) G_4: Challenger C plays the game with G_3, except there is a difference when calling ReKeyGen and ReEncrypt. In calling ReKeyGen, challenger C uses (β, α) proposed by attacker A to query the re-encryption key list; if there is a result, challenger C will feed back $rk_{A\to B}$ to attacker A. If there is no result, challenger C will query according to β and α in the key list and calculate $rk_{A\to B} = H1(rpk_A) \oplus H_1(rpk_B||\ \alpha)$. The challenger feedback is terminated if the user's private key is leaked. In calling ReEncrypt, challenger C uses (β, α, C_i) proposed by attacker A to calculate the decryption parameters in ReEncrypt. C feedback terminates; otherwise, perform a key query in the key and re-encryption key list and feedback on the ciphertext for attacker A. If the pk_i used by attacker A in ReKeyGen is not obtained through KeyGen, the challenger terminates the game. $|\ Pr[T_3\ ] - Pr[T_4\ ]||$ can be ignored.

(6) G_5: Challenger C plays the game with G_4, except that after receiving the challenge from attacker A (m_0, m_1, α), challenger C calculates the first decrypted ciphertext, selects i ϵ G, and calculates r = H_2(i), C = (C_1, C_2, C_3, C_4), C_1 = rP = (x_0, y_0), $rpk_A = (x_A, y_A)$, t = $H_1(x_A||\ y_A)$, C_2 = M⊕t, $C_3 = H3(x_A||\ M|\ |\ y_A)$, $C_4 = H4(M|\ |\ C_1||\ C_3)$. The difference between G_5 and G_4 is whether to query H_2, and the difficulty of querying H_2 is based on the DDH problem, so$|\ Pr[T_4] - Pr[T_5]||$ Can be ignored. Since the Hash function is a random process, $Pr[T_5] = 1/2(p + 1)$. Based on the above analysis,

$$|\ Pr[T_1] - Pr[T_5]| = \left|\frac{\delta_0}{p+1} - \frac{1}{2(p+1)}\right| = \left|\frac{\delta_0 - \frac{1}{2}}{p+1}\right| = \frac{\varepsilon}{p+1}$$

can be ignored; that is, ε is negligible. After proof

Theorem 2. *If the proxy re-encryption algorithm in this paper satisfies CCA security, then the blockchain data-controlled sharing scheme has privacy protection.*

Proof. First, any effective operation in the blockchain system will be recorded on the block as a transaction. Both parties to the transaction share data through the address. Therefore, the blockchain account has anonymity. Even if the attacker obtains the transaction record, tx cannot extract the user's real identity through the transaction ciphertext so this scheme can protect the user's identity privacy. Secondly, the transaction data is uploaded to the blockchain after initial encryption. Due to the blockchain incentive mechanism, if you want to, the rewarded miner nodes will proxy re-encrypt the initial transaction ciphertext and share it with legitimate data users. According to the proxy re-encryption algorithm in this paper, it can be seen that even if the attacker obtains the transaction ciphertext, he cannot obtain information about the plaintext from it. Any valid information, so this scheme can protect transaction privacy. In summary, the blockchain data-controlled sharing scheme in this paper has the privacy protection property of protecting identity privacy and transaction privacy.

Theorem 3. *If the authorized management node in this paper is credible, then the blockchain data has dynamic sharing.*

It proves that the transaction information tx is initially encrypted by the data owner A with its public key before being uploaded to the chain. The access rights of the data are divided, and the proxy re-encryption key $rk_{A\to B}$ for the legitimate user B is constructed. The authorization list L is sent to the Blockchain authorized managers; data users who meet the access rights can share transaction information from the transaction ciphertext C′. This scheme can select credible authorized management nodes through the consensus mechanism on the Blockchain to realize data access rights deterministic updates. When the access rights of transaction information change, the authorization manager updates the authorization list to L′, and the miner nodes on the Blockchain generate new proxy re-encrypted ciphertexts based on the verification results of the authorization manager. Therefore, users of old data cannot Valid transaction information obtained through its private key and α, and new data users can use their private key and decryption parameter α' to access the new re-encrypted transaction ciphertext effectively. Therefore, blockchain data has dynamic sharing. Proof complete

Theorem 4. *If the proxy re-encryption algorithm in this paper satisfies CCA security, the scheme can resist collusion attacks prove that in this scheme, the collaboration between users can only obtain the data they are authorized to access. The partnership between approved user B and the agent can only get the information that user B is allowed to access. Therefore, the following mainly refers to illegal users or undo the collaboration between the user and the proxy party.*

First, the miner nodes on the blockchain perform proxy re-encryption operations on the initial transaction ciphertext, and the attacker must conspire with more than half of the nodes on the blockchain system, that is, a 51% attack, which is obviously very expensive.

Secondly, even if the attacker successfully colludes with the miner nodes on the blockchain system and obtains the initial transaction ciphertext $C = (C_1, C_2, C_3, C_4)$ and the proxy re-encryption key $rk_{A\to B} = H1(rpk_A) \oplus H_1(rpk_B||\alpha)$, mastering the initiative of transaction ciphertext conversion, can calculate re-encrypted ciphertext $C' = (C_1', C_2', C_3', C_4')$, $C_1' = C1$, $C_2' = rk_{A\to B} \oplus C_2$, $C_1' = C3$, $C_1' = C4$. Still, the transaction data is always transmitted in the blockchain network through encryption because the proxy re-encryption algorithm in this paper satisfies CCA security, it can be seen that the plaintext of transaction data will not be leaked. This paper combines proxy re-encryption technology and blockchain technology to construct a distributed data-sharing scheme, entrusting the task of ciphertext conversion to a decentralized Completed by the blockchain system, it has the property of anti-collusion attack and can realize the safe sharing of encrypted data information. The proof is completed

6. Performance analyses

6.1. Scheme comparison

Authors [3] proposed a confidential transaction scheme based on ring signatures, which hides transaction amounts through ring signatures and protects transaction privacy and identity privacy. Authors [14] proposed a blockchain data traceability algorithm based on attribute encryption and designed A policy update algorithm suitable for blockchains that realizes dynamic protection of transaction privacy. In Authors [15] presented a blockchain data encryption based on searchable characteristics that integrated hierarchical attribute encryption with linear secret sharing. Privacy protection control scheme hides transaction information from nodes without access rights. Authorized blockchain nodes can query useful transaction information through trapdoor keywords. This addresses the issue of privacy exposure in regular blockchain transactions. Authors [31] proposed a model for blockchain to protect private data, using blockchain smart contracts to judge user access rights and symmetric or asymmetric methods to encrypt and store data. Only one-to-one secure data transmission Authors [12] proposed a blockchain-powered data sharing plan, introduced a Blockchain structure with two chains, combined with proxy re-encryption technology, and realized safe and reliable data sharing. The system in this paper combines the SM2 Encryption algorithm and proxy re-encryption technology splits and manages the generation parameters of the re-encryption key under the deterministic update of the re-encryption key, dynamically updates the access rights of the blockchain data, and realizes the controlled sharing of the blockchain data. Next, we will compare this scheme with existing research schemes in terms of whether it supports ciphertext data access control, whether it can resist collusion attacks, whether the authority update is deterministic, whether it is re-encrypted, and the data traceability of the blockchain. The comparative analysis of functional characteristics is shown in Table 2.

6.2. Calculation efficiency analysis

To illustrate the operational efficiency of this scheme, this paper lists the existing research schemes for comparison. The definition symbol T_E represents exponential operation, T_P represents bilinear pairing operation, T_H represents hash operation, T_M represents point multiplication operation of elements in a group, and XOR represents XOR operation. Table 3 shows the performance calculation comparison analysis between this scheme and Authors [18] and other projects.

In the above scheme's operation, bilinear pairing and exponential function will consume more computing resources. Authors [18] uses the cloud server to store data and perform proxy re-encryption to complete the data transfer between IoT cloud nodes. As a result, they are shared large amounts of calculation. The IoT cloud nodes to finish the data transfer, the cloud server will store data and carry out proxy re-encryption. They are therefore given access to a lot of computation. In order to realize data exchange of lightweight devices like sensors, the system also makes use of a proxy server to complete the proxy re-encryption procedure. Decentralized data sharing is made possible by a plan that uses blockchain smart contracts to perform proxy re-encryption. Data access rights still are not dynamically

Table 2
Comparison between the scheme of this paper and the existing research scheme.

Plan	Ciphertext access control	Anti-collusion attack	Authorize deterministic updates	Re-encryption	Traceable
Authors [3]	No	Yes	No	No	Yes
Authors [14]	Yes	No	Yes	No	Yes
Authors [15]	Yes	Yes	No	No	Yes
Authors [31]	Yes	No	No	No	No
Authors [12]	Yes	No	No	Yes	Yes
This article	Yes	Yes	Yes	Yes	Yes

Table 3
Performance calculation comparison of schemes.

Plan	Initial encryption	Re-encryption	Decrypt
Authors [18]	$3T_E + T_P + 4T_H$	$2T_E + 2T_P + 3T_H$	$3T_E + 2T_P + 3T_H$
Authors [32]	$3T_E + 2T_P + T_H$	$T_E + T_P$	$T_E + T_P + T_H$
Authors [12]	$2T_E + T_P + 3T_H$	$T_P + 4T_H$	$T_E + T_P$
Authors [33]	$4T_M + 2T_H$ + XOR	XOR	$5T_M + 2T_H$ + XOR
This article	$2T_M + 4T_H$ + XOR	XOR	$T_M + 2T_H$ + XOR

Table 4
Elliptic curve parameters of SM2 Public Key Cryptography Algorithm.

Curve parameters	Value
P	FFFFFFFEFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF00000000FFFFFFFFFFFFFFFF
A	FFFFFFFEFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF00000000FFFFFFFFFFFFFFFC
b	28E9FA9E9D9F5E344D5A9E4BCF6509A7F39789F515AB8F92DDBCBD414D940E93
n	FFFFFFFEFFFFFFFFFFFFFFFFFFFFFFFF7203DF6B21C6052B53BBF40939D54123
x_G	32C4AE2C1F1981195F9904466A39C9948FE30BBFF2660BE1715A4589334C74C7
y_G	BC3736A2F4F6779C59BDCEE36B692153D0A9877CC62A474002DF32E52139F0A0

updated. The scheme of Authors [32] also uses a proxy server to complete the proxy re-encryption process, realizing data sharing of lightweight devices such as sensors. Chen et al. [33] scheme using blockchain smart contracts to complete proxy re-encryption, the amount of calculation is small, and decentralized data sharing is realized. Still, there is no dynamic update of data access rights. This scheme constructs a lightweight proxy re-encryption algorithm based on SM2; the calculation overhead is small, which can meet the actual needs of blockchain transaction data sharing and realize the dynamic adjustment of blockchain data access rights.

6.3. Experimental analysis

The configuration of the experimental host in this paper is 3.20 GHz, i7-8700 CPU, 8 GB RAM, the system is Windows 11, and the programming tool is Python 3.7.4. Furthermore, the 256-bit elliptic curve in the prime number domain recommended in the SM2 elliptic curve public essential encryption algorithm standard is adopted, namely $y^2 = x^3 + ax + b$; Table 4 illustrates the curve parameters.

Analyze and compare the operation efficiency of this scheme with the Authors [33] system, and analyze the calculation efficiency of this scheme by changing the size of the data plaintext. The average value of the algorithm running 100 times is shown in Table 5, Table 6 and Fig. 4. The encryption stage includes initial encryption and proxy re-encryption, and the decryption stage represents the decryption of the re-encrypted ciphertext. A "proxy re-encryption" approach is a superior option for safely transferring an encrypted communication in the cloud. In these types of methods, a semi-trusted proxy can re-encrypt ciphertext specified for one data aggregate as one assigned for another without needing to first decode it. Cloud-based IoT data outsourcing services using a secure proxy re-encryption protocol. It can be seen from Table 5, Table 6 and Fig. 4 that the operation efficiency in this paper is higher than [33] in both the encryption and decryption phases. For example, when the data size is 128 B, the running time of the encoding phase of this scheme is 40.25 ms, and the running time of the decryption phase is 9.82 ms. ms, the Authors [33] method requires 50.43 ms for encryption and 72.08 ms for deciphering. On the other hand, when the data size is 1 024 B, the scheme in this paper takes 158.09 ms for encoding and 40.88 ms for decipherment, and the calculation cost is relatively small, which can meet the actual needs of blockchain transaction data sharing.

To completely simulate the uplink and data access of the complete transaction information, this work employs four hosts to develop the PBFT consensus algorithm in Python in order to replicate the blockchain data sharing operation. In addition, the consensus nodes use SQL Server2008 to store local block data to improve block Storage and Data Access efficiency. The lightweight proxy re-encryption method created by this scheme is based on SM2, has a low computation overhead, and can be adjusted dynamically to meet the needs of blockchain data access rights and share transaction data on the blockchain. A transaction's initial encryption, the creation of a proxy re-encryption key, writing to local blocks, and consensus verification are the four processes that make up the transaction information upload process. The transaction information upload process is divided into four steps: initial encryption of transaction data, construction of proxy re-encryption keys, writing to local blocks and, consensus verification. The data access process is divided into authority verification, proxy re-encryption, and consensus verification, and there are four steps to decrypt the re-encrypted transaction ciphertext. In Table 7 and Fig. 5 that when the data size is within 1 024 B, the execution time of the above process operation is maintained within 700 ms, and the time cost is linear with the increase of the data size growth relationship.

Table 5
Encryption phase.

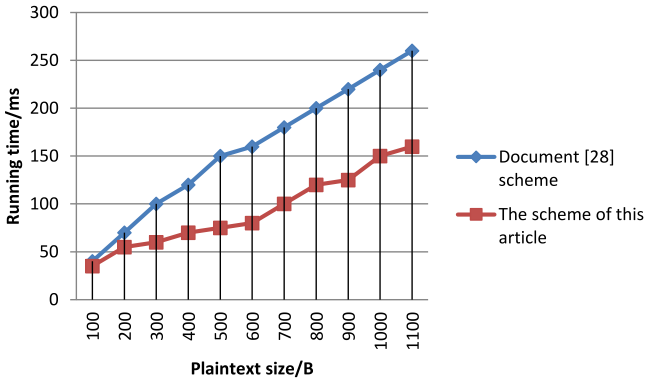
Plane text size	100	200	300	400	500	600	700	800	900	1000	1100
Document [28] scheme	40	70	100	120	150	160	180	200	220	240	260
The scheme of this article	35	55	60	70	75	80	100	120	125	150	160

Table 6
Decryption stage.

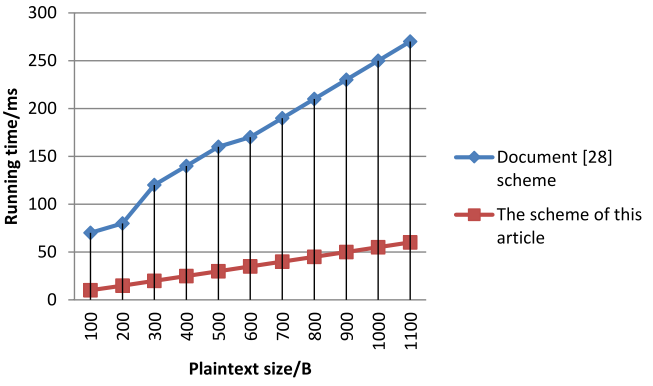
Plane text size	100	200	300	400	500	600	700	800	900	1000	1100
Document [33] scheme	70	80	120	140	160	170	190	210	230	250	270
The scheme of this article	10	15	20	25	30	35	40	45	50	55	60

Table 7
Data sharing execution time cost.

Plane text size	100	200	300	400	500	600	700	800	900	1000	1100
Data access	350	370	375	380	390	400	405	410	415	420	425
Transaction on chain	470	500	525	550	575	580	600	620	640	660	680



(a) Encryption phase



(b) Decryption stage

Fig. 4. Algorithm running time comparison.

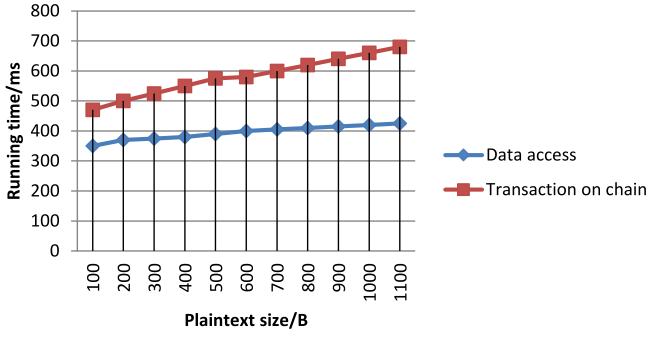


Fig. 5. Data sharing execution time cost.

7. Conclusions

This study presents a blockchain data-controlled sharing method on proxy re-encryption to address the privacy and security challenges associated with blockchain data sharing. The encryption algorithm realizes the privacy safety of transaction data sharing. In transaction information access, the update of data access authority is completed through the split management of agent re-encryption critical parameters so that the visibility of transaction data can be dynamically adjusted. Finally, security and performance analysis show that this method can achieve dynamic sharing of transaction data while protecting user privacy. Proxy re-encryption is used in the data-controlled sharing scheme to provide data security sharing while protecting transaction data privacy. The second suggestion is a dynamic adjustment mechanism for user privileges. To provide user access rights determinism, blockchain nodes split labor and independently handle re-encryption key parameters. A dynamic adjustment is made to transaction data visibility. The computational efficiency has also been improved compared with existing research schemes. It has noteworthy advantages in security, functionality, and computational overhead. An effective balance is achieved between the privacy protection and usability of blockchain data sharing. This solution meets the requirements of partial decentralization of the alliance chain and protecting transaction data privacy. It applies to applications where sensitive data is shared using blockchain-distributed databases, such as personal privacy data, legal documents, electronic medical records, etc. Therefore, the following research will be on designing an efficient and reasonable data-sharing scheme in combination with blockchain technology in specific application scenarios.

CRediT authorship contribution statement

Ismail Keshta: Conceptualization, Data curation, Supervision, Project administration, Methodology, Writing – original draft. **Yassine Aoudni:** Conceptualization, Data curation, Methodology, Writing – original draft. **Mukta Sandhu:** Conceptualization, Investigation, Methodology, Writing – original draft. **Abha Singh:** Formal analysis, Investigation, Methodology, Writing – original draft. **Pardayev Abdunabi Xalikovich:** Data curation, Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing. **Ali Rizwan:** Resources, Software, Validation, Visualization, Writing – review & editing. **Mukesh Soni:** Resources, Software, Visualization, Methodology, Writing – original draft. **Sachin Lalar:** Validation, Visualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] Z. Ullah, B. Raza, H. Shah, S. Khan, A. Waheed, Towards blockchain-based secure storage and trusted data sharing scheme for IoT environment, IEEE Access 10 (2022) 36978–36994, http://dx.doi.org/10.1109/ACCESS.2022.3164081.

[2] K.O.-B.O. Agyekum, Q. Xia, E.B. Sifah, C.N.A. Cobblah, H. Xia, J. Gao, A proxy re-encryption approach to secure data sharing in the Internet of Things based on blockchain, IEEE Syst. J. 16 (1) (2022) 1685–1696, http://dx.doi.org/10.1109/JSYST.2021.3076759.

[3] W. Yang, Z. Guan, L. Wu, X. Du, M. Guizani, Secure data access control with fair accountability in smart grid data sharing: An edge blockchain approach, IEEE Internet Things J. 8 (10) (2021) 8632–8643, http://dx.doi.org/10.1109/JIOT.2020.3047640.

[4] T. Li, H. Wang, D. He, J. Yu, Blockchain-based privacy-preserving and rewarding private data sharing for IoT, IEEE Internet Things J. 9 (16) (2022) 15138–15149, http://dx.doi.org/10.1109/JIOT.2022.3147925.

[5] L. Xue, et al., Blockchain-based data sharing with key update for future networks, IEEE J. Sel. Areas Commun. 40 (12) (2022) 3437–3451, http://dx.doi.org/10.1109/JSAC.2022.3213312.

[6] Y. Wang, A. Zhang, P. Zhang, Y. Qu, S. Yu, Security-aware and privacy-preserving personal health record sharing using consortium blockchain, IEEE Internet Things J. 9 (14) (2022) 12014–12028, http://dx.doi.org/10.1109/JIOT.2021.3132780.

[7] S. Wang, D. Zhang, Y. Zhang, Blockchain-based personal health records sharing scheme with data integrity verifiable, IEEE Access 7 (2019) 102887–102901, http://dx.doi.org/10.1109/ACCESS.2019.2931531.

[8] Y. Wang, A. Zhang, P. Zhang, H. Wang, Cloud-assisted EHR sharing with security and privacy preservation via consortium blockchain, IEEE Access 7 (2019) 136704–136719, http://dx.doi.org/10.1109/ACCESS.2019.2943153.

[9] K. Xue, X. Luo, H. Tian, J. Hong, D.S.L. Wei, J. Li, A blockchain based user subscription data management and access control scheme in mobile communication networks, IEEE Trans. Veh. Technol. 71 (3) (2022) 3108–3120, http://dx.doi.org/10.1109/TVT.2021.3138203.

[10] Z. Guan, X. Zhou, P. Liu, L. Wu, W. Yang, A blockchain-based dual-side privacy-preserving multiparty computation scheme for edge-enabled smart grid, IEEE Internet Things J. 9 (16) (2022) 14287–14299, http://dx.doi.org/10.1109/JIOT.2021.3061107.

[11] K. Fan, et al., A secure and verifiable data sharing scheme based on blockchain in vehicular social networks, IEEE Trans. Veh. Technol. 69 (6) (2020) 5826–5835, http://dx.doi.org/10.1109/TVT.2020.2968094.

[12] G. Wu, S. Wang, Z. Ning, J.L. records, Blockchain-enabled privacy-preserving access control for data publishing and sharing in the internet of medical things, IEEE Internet Things J. 9 (11) (2022) 8091–8104, http://dx.doi.org/10.1109/JIOT.2021.3138104.

[13] Z. Pang, Y. Yao, Q. Li, X. Zhang, J. Zhang, Electronic health records sharing model based on blockchain with checkable state PBFT consensus algorithm, IEEE Access 10 (2022) 87803–87815, http://dx.doi.org/10.1109/ACCESS.2022.3186682.

[14] X. Yang, T. Li, X. Pei, L. Wen, C. Wang, Medical data sharing scheme based on attribute cryptosystem and blockchain technology, IEEE Access 8 (2020) 45468–45476, http://dx.doi.org/10.1109/ACCESS.2020.2976894.

[15] L. Guo, X. Yang, W.-C. Yau, TABE-DAC: Efficient traceable attribute-based encryption scheme with dynamic access control based on blockchain, IEEE Access 9 (2021) 8479–8490, http://dx.doi.org/10.1109/ACCESS.2021.3049549.

[16] G. Wu, S. Wang, Z. Ning, B. Zhu, Privacy-preserved electronic medical record exchanging and sharing: A blockchain-based smart healthcare system, IEEE J. Biomed. Health Inf. 26 (5) (2022) 1917–1927, http://dx.doi.org/10.1109/JBHI.2021.3123643.

[17] Y. Pu, C. Hu, S. Deng, A. Alrawais, R^2PEDS: A recoverable and revocable privacy-preserving edge data sharing scheme, IEEE Internet Things J. 7 (9) (2020) 8077–8089, http://dx.doi.org/10.1109/JIOT.2020.2997389.

[18] S. Niu, L. Chen, J. Wang, F. Yu, Electronic health record sharing scheme with searchable attribute-based encryption on blockchain, IEEE Access 8 (2020) 7195–7204, http://dx.doi.org/10.1109/ACCESS.2019.2959044.

[19] J. Tao, L. Ling, Practical medical files sharing scheme based on blockchain and decentralized attribute-based encryption, IEEE Access 9 (2021) 118771–118781, http://dx.doi.org/10.1109/ACCESS.2021.3107591.

[20] Y. Lu, X. Huang, K. Zhang, S. Maharjan, Y. Zhang, Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles, IEEE Trans. Veh. Technol. 69 (4) (2020) 4298–4311, http://dx.doi.org/10.1109/TVT.2020.2973651.

[21] H. Jin, Y. Luo, P. Li, J. Mathew, A review of secure and privacy-preserving medical data sharing, IEEE Access 7 (2019) 61656–61669, http://dx.doi.org/10.1109/ACCESS.2019.2916503.

[22] M. Zhao, W. Liu, K. He, Research on data security model of environmental monitoring based on blockchain, IEEE Access 10 (2022) 120168–120180, http://dx.doi.org/10.1109/ACCESS.2022.3221109.

[23] Y. Jiang, Y. Zhong, X. Ge, IIoT data sharing based on blockchain: A multileader multifollower stackelberg game approach, IEEE Internet Things J. 9 (6) (2022) 4396–4410, http://dx.doi.org/10.1109/JIOT.2021.3103855.

[24] K. Yu, L. Tan, M. Aloqaily, H. Yang, Y. Jararweh, Blockchain-enhanced data sharing with traceable and direct revocation in IIoT, IEEE Trans. Ind. Inform. 17 (11) (2021) 7669–7678, http://dx.doi.org/10.1109/TII.2021.3049141.

[25] H. Yin, E. Chen, Y. Zhu, C. Zhao, R. Feng, S.S. Yau, Attribute-based private data sharing with script-driven programmable ciphertext and decentralized key management in blockchain Internet of Things, IEEE Internet Things J. 9 (13) (2022) 10625–10639, http://dx.doi.org/10.1109/JIOT.2021.3124016.

[26] M.B. Mollah, M. Azad, A. Vasilakos, Secure data sharing and searching at the edge of cloud-assisted Internet of Things, IEEE Cloud Comput. 4 (2017) 34–42.

[27] W. Wei, X. Peng, L.T. Yang, Secure data collection, storage and access in cloud-assisted IoT, IEEE Cloud Comput. 5 (2018) 77–88.

[28] H. Yin, Y. Xiong, J. Zhang, L. Ou, S. Liao, Z. Qin, A key-policy searchable attribute-based encryption scheme for efficient keyword search and fine-grained access control over encrypted data, Electronics 8 (2019) 265.

[29] J. Fu, Y. Liu, H. Chao, B. Bhargava, Z. Zhang, Secure data storage and searching for industrial IoT by integrating fog computing and cloud computing, IEEE Trans. Ind. Inform. 14 (2018) 4519–4528.

[30] T. Alladi, V. Chamola, B. Sikdar, K.R. Choo, Consumer IoT: Security vulnerability case studies and solutions, IEEE Consum. Electron. Mag. 9 (2020) 17–25.

[31] W. Dai, S. Tuo, L. Yu, K.-K.R. Choo, D. Zou, H. Jin, HAPPS: A hidden attribute and privilege-protection data-sharing scheme with verifiability, IEEE Internet Things J. 9 (24) (2022) 25538–25550, http://dx.doi.org/10.1109/JIOT.2022.3197708.

[32] L. Zhang, Y. Zhang, Q. Wu, Y. Mu, F. Rezaeibagha, A secure and efficient decentralized access control scheme based on blockchain for vehicular social networks, IEEE Internet Things J. 9 (18) (2022) 17938–17952, http://dx.doi.org/10.1109/JIOT.2022.3161047.

[33] S. Wang, Y. Zhang, Y. Zhang, A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems, IEEE Access 6 (2018) 38437–38450, http://dx.doi.org/10.1109/ACCESS.2018.2851611.

Ismail Keshta received his B.Sc. and the M.Sc. degrees in computer engineering and his Ph.D. in computer science and engineering from the King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia, in 2009, 2011, and 2016, respectively. He was a lecturer in the Computer Engineering Department of KFUPM from 2012 to 2016. Prior to that, in 2011, he was a lecturer in Princess Nourah Bint Abdul Rahman University and Imam Muhammad Ibn Saud Islamic University, Riyadh, Saudi Arabia. He is currently an assistant professor in the computer science and

information systems department of AlMaarefa University, Riyadh, Saudi Arabia. His research interests include software process improvement, modeling, and intelligent systems. He can be contacted at email: imohamed@mcst.edu.sa.

Yassine Aoudni received the Ph.D. degree in computer system engineering from the University of Sfax, Tunisia in 2010. the M.S degree in information technologies and electrical engineering from university of Sfax in 2003 and 2002, respectively. Since 2014, he is an Associate Professor, in Department of Computers and information technologies, Art and sciences college, Northern Borders University, Kingdom Saudi Arabia. From 2014 to 2010, he was an associate professor in electrical department in National Engineering school in Sfax. 2007, he became a permanent member in Computer and Embedded System Research Laboratory of Sfax university. In 2003, he participated in CMCU research project between university of Sfax, Tunisia and Southern Brittany University, France. His research interests include rapid prototyping of software and hardware applications on FPGA devices, application code generation for multiprocessing SoC and Biometric applications.

Dr. Mukta Sandhu is fervent educational professional. She has vast and diverse experience of more than decade in the field of Computer Science and Engineering for teaching theories, mentoring projects both in India and USA. She has served as Associate Professor in Bunker Hill College, Boston (USA) and worked as adjunct faculty in North Eastern University (USA). She has published numerous research papers in various journals of repute like IEEE Xplore, Springer, Elsevier to name few. She has 4 patent to her name. She is heading the Entrepreneurship cell of the university as well center head for monitoring employability of the university.

Dr. Abha Singh is working as Assistant Professor in Saudi Electronic University in Department of Basic Science, College of Science and theoretical studies. She has received her Ph.D. in Mathematics subjects. She has more than 14 years experience in teaching and research. Her research interests are AI, Fractal, Mathematics and its Applications and Statistics. She has visited many universities for presenting the research papers. Several research papers of her work published in the leading National and International Journals.

Pardayev Abdunabi Xalikovich is Professor, Doctor of Science in Economics, The Head of the Department of Scientific Research, Innovation and Training of Scientific and Pedagogical Personnel, Professor of "Accounting" department of Tashkent Institute of Finance, Tashkent, Uzbekistan.

Ali Rizwan, Ph.D. is working as an Assistant Professor in the Department of Industrial Engineering at King Abdulaziz University, Jeddah, Saudi Arabia. He did his Ph.D. in Knowledge Management and Leadership. He has more than 25 years of professional experience in the fields of education, entrepreneurship and consultancy. His areas of interest include pedagogy, soft skills, emotional intelligence, knowledge management, industrial applications, artificial intelligence, IOT, machine learning, health care management, human development and organizational development.

Mukesh Soni working as an Assistant Professor in Department of CSE, University Centre for Research & Development Chandigarh University, Mohali, Punjab, 140413, India. I Have completed my Bachelor's in Information Technology from Gyan Ganga Institute of Technology & Management, Bhopal, India in 2011, and a Masters in Computer Science & Engineering from MANIT, Bhopal, India in 2015. I am associated with NPTEL (IIT Project) as a Quality Control Person since 2019. He is also a member of many International and National professional bodies like IEEE, Asia Society of Researcher, Scientific and Technical Research Association (STRA), "International Association of Engineers Institute for Engineering Research and Publication, Scholars Academic & Scientific Society.

Dr. Sachin Lalar is working as Assistant Professor in Kurukshetra University in Department of Computer Science and Applications. Currently, he has received his Ph.D. in Computer Science and Engineering. He has more than 14 years' experience in teaching and research. His research interests are Computer Networks, Data Structure and Programming. He has visited many universities for presenting the research papers. Several research papers of his work published in the leading National and International Journals.