Blockchain: Research and Applications 5 (2024) 100204

Contents lists available at ScienceDirect

Blockchain: Research and Applications

journal homepage: www.journals.elsevier.com/blockchain-research-and-applications

Research Article

Privacy-preserving pathological data sharing among multiple remote parties

Wei Wu [a], Fulong Chen [a,*], Pinghai Yuan [a], Taochun Wang [a], Dong Xie [a], Chuanxin Zhao [a], Chao Wang [a], Detao Tang [a], Jingtao Li [a], Ji Zhang [b]

[a] Anhui Provincial Key Laboratory of Network and Information Security, Anhui Normal University, 241002 Wuhu, China
[b] University of Southern Queensland, 4350 QLD, Australia

ARTICLE INFO

Keywords:
Digital remote pathology
Conditional proxy re-encryption
Blockchain
Data storage
Data sharing

ABSTRACT

The sharing of pathological data is highly important in various applications, such as remote diagnosis, graded diagnosis, illness treatment, and specialist system development. However, ensuring reliable, secure, privacy-preserving, and efficient sharing of pathological data poses significant challenges. This paper presents a novel solution that leverages blockchain technology to ensure reliability in pathological data sharing. Additionally, it employs conditional proxy re-encryption (C-PRE) and public key encryption with equality test technology to control the scope and preserve the privacy of shared data. To assess the practicality of our solution, we implemented a prototype system using Hyperledger Fabric and conducted evaluations with various metrics. We also compared the solution with relevant schemes. The results demonstrate that the proposed solution effectively meets the requirements for pathological data sharing and is practical in production scenarios.

1. Introduction

Pathological data are valuable resources that can be utilized for remote diagnosis, education, and the development of specialist systems. Pathological data encompass not only diagnosis reports but also examination and medical images, as well as pathological slices, which are vital for illness identification and treatment. Its broad sharing has the potential to benefit numerous entities and improve human life quality.

The advancements in technologies such as image processing, big data processing, and machine learning, particularly deep learning, have greatly facilitated the utilization of medical data, including pathological data [1,2]. Therefore, how to effectively utilize pathological data is a valuable research topic, and network-based sharing is considered an ideal solution to facilitate its broad utilization.

Due to the privacy-sensitive nature of pathological data and their potential use in remote diagnosis, ensuring the reliability of data-sharing nodes and the confidentiality of the sharing scheme are critical requirements. Cloud storage has been widely used in storing and retrieving pathological data [3–5]. Cloud servers cannot mitigate the problems of single points of failure and fully address the reliability problem.

Moreover, cloud-based pathology data storage and sharing rely on third-party servers and are vulnerable to security breaches in the event of an attack on the server or if the server is itself a malicious node. Naively, traditional encryption techniques can protect data security, but they may lack flexibility to preserve patients' privacy and suffer from frequent encryption and decryption challenges [6,7].

Recently, people have begun to explore blockchain to create distributed storage [8,9]. Blockchain's decentralized and tamper-proof nature provide transparency, openness, and traceability, making it well-suited for secure storage and sharing of medical data. Moreover, to further enhance the security and flexibility of data sharing, some schemes employ proxy re-encryption (PRE) technology. PRE can accomplish the conversion of ciphertext without exposing any kind of plaintext, thus it enhances data security. Moreover, because it can avoid the frequent encryption and decryption challenges of traditional asymmetric encryption during data sharing, it can also effectively improve system performance.

* Corresponding author.
E-mail addresses: ww1140@ahnu.edu.cn (W. Wu), long005@ahnu.edu.cn (F. Chen), wangtc@ahnu.edu.cn (T. Wang), xiedong@ahnu.edu.cn (D. Xie), zhaocx@ahnu.edu.cn (C. Zhao), wangchao@ahnu.edu.cn (C. Wang), cw4094@ahnu.edu.cn (D. Tang), lijintao@ahnu.edu.cn (J. Li), Ji.Zhang@usq.edu.au (J. Zhang).

https://doi.org/10.1016/j.bcra.2024.100204
Received 3 August 2023; Received in revised form 8 March 2024; Accepted 30 April 2024
Available online 9 May 2024

Note that a modern pathological data sharing system usually involves multiple parties (data requesters besides data owners), and specific data fields (of items) can only be accessed by authorized requesters and should be concealed from others. Therefore, the sharing scheme should have the capability to share data fields in terms of permissions plus conditions. In this scenario, PRE is stuck as pathological data contain privacy information.

This paper proposes a novel blockchain-based storage and sharing scheme specifically designed for pathological data. By eliminating the need for a centralized server and leveraging smart contracts through blockchain technology, our system ensures reliability. The incorporation of conditional proxy re-encryption (C-PRE) provides a secure and flexible approach to sharing pathological data. Data owners have control over the re-encryption authority, ensuring the privacy of their sensitive information. Additionally, the ciphertext equality test enables secure message matching without the need for decryption during the sharing process, further enhancing data security.

To evaluate the practicality of our proposed solution, we implemented a prototype system based on Hyperledger Fabric and evaluated its performance with various metrics. The evaluation results demonstrated that our system outperforms several recent solutions. Overall, our system presents a robust solution for privacy-preserving pathological data sharing, addressing the challenges of confidentiality, reliability, and flexibility.

The paper is structured as follows: Section 2 and Section 3 provide a review of related literature and theoretical foundations, respectively. In Section 4, we present the design of our proposed pathological data blockchain storage and sharing scheme, including its system architecture, workflow, and design objectives. The specific details of the scheme are elaborated in Section 5, followed by the security proof in Section 6 and the performance evaluation in Section 7. Finally, we conclude the paper in Section 8.

2. Related work

This section briefly summarizes some literature about C-PRE and blockchain-based data sharing.

Proxy re-encryption (PRE). PRE [10] is a system that enables a proxy, equipped with a conversion key granted by an authorizer, to convert the original ciphertext intended for an authorized individual into ciphertext intended for another authorized person. The latter recipient can then decrypt the converted ciphertext using his own private key. This approach resolves the issue of frequent encryption and decryption associated with traditional asymmetric encryption during data sharing, thereby enhancing security and flexibility.

However, traditional PRE schemes often suffer from significant overhead and low performance. To address these challenges, Yang and Ma [11] proposed a keyword searchable PRE scheme, enabling patients to delegate time-limited access to their records while facilitating search functionalities. Access and search privileges are automatically revoked once the designated validity period elapses.

In another track, Li et al. [12] combined PRE technology with the equality test, enabling secure and flexible search capabilities for medical record data encrypted with different public keys. This approach ensures the confidentiality of both the key and plaintext while facilitating efficient sharing and retrieval of medical record data.

Conditional proxy re-encryption (C-PRE). The concept of C-PRE was initially introduced by Tang [13] and Weng et al. [14]. In C-PRE, an authorizer possesses the capability to generate a conditional conversion key based on a predefined conditional expression. The proxy, who receives this conditional conversion key, can only convert ciphertext that satisfies the specified condition. This mechanism enables effective control over the proxy's authority.

Furthermore, Fimiani [15] proposed a fuzzy conditional identity-based PRE scheme. This scheme leverages biometric information to derive keys, ensuring secure and privacy-preserving exchange of medical documents. By incorporating biometric data, this approach enhances the protection of privacy in the re-encryption process.

Public key encryption with equality test (PKEET). The PKEET is a cryptographic scheme initially proposed by Yang et al. [16] that enables the testing of equality between encrypted data using either different public keys or the same public key. By reducing the number of encryption and decryption operations and enhancing data sharing efficiency, PKEET facilitates secure data sharing.

To address the authorization challenge associated with the equality test, Tang [17] introduced FG-PKEET, a ciphertext equality test scheme that provides fine-grained authorization. Through a negotiation process between two parties, an authorization trapdoor is generated, granting exclusive access to the trapdoor holder for performing the equality test.

As mentioned above, Li et al. [12] combined PRE technology with an equality test, leveraging both PRE and isometric testing. This integration enables users to search for required medical record data from data encrypted under different public keys while ensuring secure data sharing and reducing the need for multiple encryption and decryption steps. Consequently, the efficiency of data sharing is significantly improved. Over time, various schemes tailored to different scenarios have been proposed to support specific application requirements [18–20].

Blockchain-based data sharing. Since the blockchain concept was introduced by Nakamoto in 2008, many blockchain platforms like Bitcoin, Ethereum, and Hyperledger Fabric have emerged and found applications in diverse fields, such as finance, education, and healthcare. These platforms offer distinct features like decentralization, tamper-proofing, and traceability, making them highly suitable for various uses.

In the healthcare domain, Li et al. [21] introduced EHRChain, a blockchain-based electronic medical record system. They devised a secure and reliable storage scheme for electronic medical records utilizing blockchain technology and the InterPlanetary File System (IPFS). Additionally, they implemented secure sharing of large-capacity medical data through an attribute-based homomorphic encryption system.

To enhance the flexibility of medical data sharing on the blockchain, Wu et al. [22] proposed a secure electronic health record system that combines attribute cryptography and blockchain technology. This system employs attribute-based encryption (ABE) and identity-based encryption (IBE) to encrypt medical data on the blockchain and uses identity-based signatures (IBSs) for digital signatures.

Addressing the challenge of secure sharing of electronic medical records, Chen et al. [23] put forth an electronic medical record system based on a consortium chain and PRE. This scheme integrates electronic devices with the blockchain network, ensuring secure data access through the automated execution of chaincode.

The above solutions have some problems, such as centralized storage being vulnerable to single-point attacks, frequent encryption and decryption processes during sharing, and the inability to control the scope of sharing and patient privacy.

3. Preliminaries

3.1. Blockchain and smart contract

Blockchain is a decentralized database where each participant in the network maintains its own ledger, ensuring data consistency throughout the network. By employing cryptographic algorithms, consensus mechanisms, and the inherent characteristics of the blockchain, transactions are secured, transparent, and immutable. There are three main types of blockchains: public chains, private chains, and consortium chains. Public chains are openly accessible to anyone, private chains are limited to specific organizations, and consortium chains fall somewhere in between, typically utilized by members with designated roles. In the context of storing and sharing pathological data, a consortium chain built on Hyperledger Fabric was selected as our development platform.

Hyperledger Fabric offers several advantageous features, including configurable consensus protocols, a flexible trust model, a modular ar-

chitecture, and the ability to create private channels. These attributes make it highly suitable for various scenarios, particularly those involving sensitive data.

The Hyperledger Fabric platform utilized in this work incorporates the following notable features:

- Three components: Hyperledger Fabric consists of three primary components, namely Fabric-CA, peer nodes, and ordered nodes. These components collectively handle crucial tasks like receiving transaction requests, managing identities, and executing transactions transparently within the blockchain network.
- Practical Byzantine fault tolerance (PBFT) Consensus: PBFT consensus has solved the problem of the inefficiency of the original Byzantine fault tolerance (BFT) algorithm and is applied to transactions where there is trust between the parties. "Fault tolerance" means that this mechanism is used to tolerate the existence of a certain number of malicious nodes so that they will not affect the normal achievement of the entire consensus.
- Smart contract: A smart contract is an autonomous contract that can execute transactions and enforce agreements automatically. Its programmability allows it to execute actions based on predetermined conditions, thereby exhibiting its "smartness". Once deployed on a blockchain platform, the contract code becomes immutable, safeguarding it against modifications. In theory, smart contracts have the potential to perform various computational tasks.

3.2. Computational hardness assumption

This work proposes a conditional re-encryption scheme that relies on the foundational assumptions of computational hardness and complexity assumptions in bilinear groups. We introduce the bilinear groups and then the complexity assumptions.

Let p be a safe large prime number with l bits in length, $(G,+)$ and $(G_T,\cdot)$ are both p-order groups. The scale (G,G_T) is a **symmetric bilinear group** if there exists a map $e: G\times G\rightarrow G_T$ satisfying the following three properties:

- Bilinearity. For $\forall P,Q\in G$ and $\forall a,b\in Z_p$, the equation $e(aP,bQ)=e(P,Q)^{ab}$ holds.
- Nondegeneracy. Scale value $e(g_1,g_2)^{ab}\neq 1_{G_T}$, where 1_{G_T} is the multiplicative identity element of G_T.
- Computability. There are valid algorithms that can calculate the value of $e(P,Q)$ for $\forall P,Q\in G$.

Bilinear Decisional Diffie–Hellman (BDDH) assumption [24], in short, is a problem in which logarithm is hard to calculate but exponential is easy to calculate. Let G and G_T be primes of order q, then the BDDH problem on (G,G_T) is as follows: Given $\left(g,g^a,g^b,g^c,Z\right)\in\{G^4\times G_T\}$ for unknown $a,b,c\in Z_q^*$, determine whether $Z=e(g,g)^{abc}$ is true. In general, for a polynomial-time adversary $\mathcal{A}$, its **advantage** (predominance) against the BDDH problem on the group G is defined as:

$$Adv^{BDDH}_{(G,G_T),\mathcal{A}}=\left|\mathcal{P}\left[\mathcal{A}\left(g,g^a,g^b,g^c,e(g,g)^{abc}\right)=1\right]-\mathcal{P}\left[\mathcal{A}\left(g,g^a,g^b,g^c,e(g,g)^{z}\right)=1\right]\right| \tag{1}$$

The probability $\mathcal{P}()$ takes into account all the random cases used by adversary $\mathcal{A}$. For any polynomial t-time adversary $\mathcal{A}$, if his advantage is less than ε, then the (t,ε)-BDDH hypothesis on the group G is said to be true.

A variant of the BDDH is the **3-weak Bilinear Decisional Diffie–Hellman Inversion** (3-wBDDHI) assumption [25]. Let G and G_T be primes of order q, then the problem on (G,G_T) is as follows: Given a 3-wBDDHI tuple $\left(g,g^{\frac{1}{a}},g^a,g^{a^2},g^b,Z\right)\in\{G^5\times G_T\}$ for unknown $(a,b)\in Z_q^*$, determine whether $Z=e(g,g)^{\frac{b}{a^2}}$ holds. In general, for a polynomial adversary $\mathcal{A}$, its **advantage** against 3-wBDDHI problems on (G,G_T) is defined as:

$$Adv^{3-wBDDHI}_{(G,G_T),\mathcal{A}}=\left|\mathcal{P}\left[\mathcal{A}\left(g,g^{\frac{1}{a}},g^a,g^{a^2},g^b,e(g,g)^{\frac{b}{a^2}}\right)\right]-\mathcal{P}\left[\mathcal{A}\left(g,g^{\frac{1}{a}},g^a,g^{a^2},g^b,Z\right)\right]\right| \tag{2}$$

The probability $\mathcal{P}()$ takes into account the random selectivity of (a,b) and Z, and all random cases used by the adversary. The (t,ε)-3-wBDDHI on the group G is said to be true if for any polynomial t-time adversary $\mathcal{A}$, the advantage is less than ε.

Table 1
Pathological data structure in blockchain storage. Privacy data must remain concealed from unauthorized parties.

Name	Type	Interpretation	Privacy
CaseID	string	A unique ID for this examination	false
PatName	string	Patient's name	**true**
PatID	string	Patient's ID-card number	**true**
Gender	string	Patient's gender	false
Birth	date	Patient's birthday	false
PhoneNo	string	Patient's phone NO. for contacting	**true**
Area	string	Location of examination (lesion)	false
PathType	string	Datatype of this pathology data	false
Seen	text	Generally seen with eyeball	false
Complaint	text	Complaints from patient	false
Introduce	text	Introduction to the condition	false
SliceAddr	string	URL address to a file stored in IPFS	false
DocName	string	Doctor's name	**true**
Time	date	Diagnosis time of this examination	false
Result	text	Examination result	false

3.3. Privacy in pathological data

Given the heterogeneous nature of pathological data stemming from multiple sources, the absence of unified standards poses challenges in facilitating the effective sharing of such data. Additionally, the presence of private information, including patient names and IDs, necessitates encryption or re-encryption procedures to safeguard privacy, thereby hindering the sharing of the private portion of data. To address these concerns, this paper utilizes the C-PRE.

To leverage the features of C-PRE, the paper establishes standardization for the fields of pathological data, as outlined in Table 1. Patient information associated with privacy, such as names and ID numbers, is marked as private fields that remain unaltered during the PRE process, ensuring that the data consumers cannot decrypt this information using their private key. However, other pathological diagnostic information can be accessed and decrypted. Medical images and pathology slices are stored on an IPFS, and data consumers can access these data by decrypting the corresponding URL addresses within the pathological data.

4. System design

This section describes the system workflow and security assumptions and discusses our design goals.

4.1. System architecture

This paper presents a novel scheme for privacy-preserving sharing of pathological data using blockchain storage and C-PRE. Fig. 1 illustrates the system architecture from various perspectives, showing four distinct layers that outline the flow of data processing and utilization. Beginning at the bottom, the *generation layer* represents the production of pathological data. Moving up, the *transfer layer* demonstrates the encryption and transmission of data among different entities. The *data*

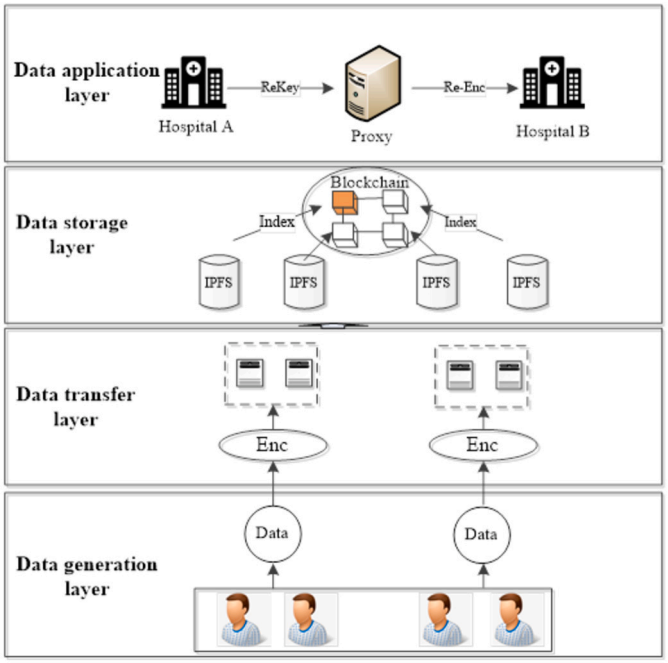

Fig. 1. System architecture illustrated from different perspectives, outlining the flow of data processing and utilization.

storage layer highlights the storage of data on a blockchain for sharing purposes, while larger pathological files are stored separately on IPFS servers. Finally, the *application layer* at the top shows how data are utilized and consumed by different entities within the system.

Meanwhile, internally, the system involves four distinct entities, namely, the data owner, data consumer (requester), proxy re-encrypter, and blockchain.

Data owner: Typical data owners are hospitals. They are responsible for providing medical treatment to patients and, consequently, possess comprehensive pathological data related to the patients. These data are valuable for various purposes, including pathological diagnosis, educational activities, and sharing with authorized data requesters.

Data requester: The entity that initiates data requests to the blockchain network is referred to as the data requester. In other words, data requesters are entities seeking to obtain data from data owners. Generally, owners are data producers, while requesters are data consumers. Sometimes, a data requester can also be a data owner if he launches requests to access data that he himself does not possess.

Proxy re-encrypter: The proxy re-encrypts the data owned by the data owner, converting it into a ciphertext specific to pathological data, which is then provided to the data requester. This enables data requesters to utilize their private key to access and obtain the shared data while not needing to touch any kinds of plaintext.

Blockchain: Pathological data are stored using blockchain technology. Pathological structured data are directly stored on the blockchain, while large files like medical images and pathology slices are stored on a distributed file system—IPFS. Indexes (URLs indeed) for accessing large files, serving as attributes of pathological structured data objects, are stored on the blockchain. Note that the IPFS network functions as a supportive storage solution for the blockchain; thus, it is considered an integral part of the blockchain entity.

4.2. System workflow

The system workflow operates as follows: Taking the sharing of data between two hospital instances as an example, when Hospital B requests to obtain pathological data, it sends a request ciphertext to the blockchain network. Then the blockchain network calls the ciphertext equality test algorithm to match the corresponding ciphertext data and initiates a sharing request to the owner of the ciphertext data, Hospital A. After Hospital A agrees to the request, it invokes the re-encryption key generation algorithm to generate a C-PRE key and sends it to the proxy re-encrypter (acting as a third party). The proxy re-encrypter calls the re-encryption algorithm to re-encrypt the non-private data that meet the requirements through the re-encryption key, and sends the re-encrypted data to Hospital B. Finally, Hospital B decrypts the obtained ciphertext through its own private key and obtains the plaintext of the shared pathology data (including the indexes for accessing pathological images stored on the IPFS).

The collaboration between the IPFS and blockchain facilitates seamless completion of the data storage process, ensuring secure storage while alleviating the burden on the blockchain. The transaction process is executed through the blockchain's *consensus mechanism*, guaranteeing the security and immutability of the transaction.

Based on the described workflow, the system encompasses nine fundamental functionalities.

- $\boldsymbol{Setup(k) \rightarrow (par)}$: To initiate a data accessing procedure, the system generates a global public parameter par using the specified security parameter k.
- $\boldsymbol{KeyGen(par) \rightarrow (pk_i, sk_i)}$: Utilizing the public parameter par as input to generate a public–private key pair (pk_i, sk_i) for the user (data owner or data requester).
- $\boldsymbol{ReKeyGen(par, sk_i, pk_j, w) \rightarrow rk}$: Create conditional proxy re-encryption key rk by combining the data owner's private key sk_i, the data requester's public key pk_j, and the condition w.
- $\boldsymbol{Enc\left(par, pk_i, m, w\right) \rightarrow C_i}$: Under the condition w, encrypt the plaintext m using the data owner's public key pk_i to produce ciphertext C_i.
- $\boldsymbol{Trapdoor(C, sk) \rightarrow td}$: Create an authorization trapdoor td using the ciphertext C and the private keys sk.
- $\boldsymbol{Test(C_i, C_j; td_i, td_j)}$: Perform the ciphertext equality test. This function returns *TRUE* if $\{C_i = C_j \wedge td_i = td_j\}$; otherwise, it returns *FALSE*.
- $\boldsymbol{ReEnc(rk, C_i) \rightarrow C_{i \rightarrow j}}$: This PRE algorithm uses key rk to convert the ciphertext C_i of the data owner into the ciphertext $C_{i \rightarrow j}$ of the data requester.
- $\boldsymbol{Dec_2(C_i, sk_i) \rightarrow m}$: The data owner uses his private key sk_i to decrypt the ciphertext C_i and retrieve the plaintext m.
- $\boldsymbol{Dec_1(C_{i \rightarrow j}, sk_j) \rightarrow m}$: The data requester executes this algorithm using his private key sk_j to decrypt the re-encrypted ciphertext $C_{i \rightarrow j}$ to retrieve the plaintext m.

These functionalities work closely together to establish a context and encrypt and re-encrypt keys for data sharing.

4.3. Threat model

In this scheme, we define the following five oracles that can characterize the ability of adversary $\mathcal{A}$.

KeyGen oracle $\mathcal{O}_k$: The challenger C calls the algorithm $KeyGen$ to generate a public–private key pair (pk_j, sk_j), and then sends (pk_j, sk_j) to the adversary $\mathcal{A}$.

ReKeyGen oracle $\mathcal{O}_{rk}$: The challenger C calls the algorithm $ReKeyGen$ to generate a C-PRE key rk, and sends rk to the adversary $\mathcal{A}$.

Enc oracle $\mathcal{O}_e$: The challenger C calls the algorithm Enc to produce a ciphertext C_i and sends C_i to the adversary $\mathcal{A}$.

ReEnc oracle $\mathcal{O}_{re}$: The challenger C calls the algorithm $ReEnc$ to generate the re-encrypted ciphertext $C_{i \rightarrow j}$ and sends $C_{i \rightarrow j}$ to the adversary $\mathcal{A}$.

Dec oracle $\mathcal{O}_d$: The challenger $\mathcal{C}$ calls the algorithm Dec_1 to decrypt the re-encrypted ciphertext $C_{i\to j}$, retrieve the plaintext m, and then sends m to the adversary $\mathcal{A}$.

4.4. Security assumptions

The proposed system is built upon the Hyperledger Fabric blockchain platform, which leverages a distributed Fabric-CA (Certificate Authority) server for user registration. As a benefit, it can mitigate the risk of a single point of attack, ensuring enhanced security. Within the system, users are considered trustworthy entities and are required to undergo certification by the Fabric-CA during the registration process, verifying their authenticity and integrity.

Meanwhile, smart contracts play a crucial role in the system as they are a set of predefined, self-executing programs that are widely trusted. These smart contracts ensure that the execution of predetermined actions is restricted by predefined rules, maintaining the integrity and consistency of the system. As in previous works, we also assume that adversaries cannot inject smart contracts into the blockchain, and all data transactions are processed with authorized smart contracts.

This system assumes that adversaries are external entities that can access all ciphertext data and re-encryption keys. These adversaries are considered to be polynomial-time entities, possessing a certain level of computational power. We acknowledge the existence of these adversaries and design our system with robust security measures to safeguard sensitive information from unauthorized access or extraction.

4.5. Design goals

Our design goals encompass both functional and security requirements, aiming to guarantee the following properties.

Correctness: Ensuring the correctness of our scheme is paramount. This entails enabling users to accurately employ their private keys corresponding to their respective public keys for decrypting ciphertexts or proxy re-encrypted ciphertexts. Moreover, our *ciphertext equality test algorithm* must output *TRUE* only when two ciphertexts contain the same message and the user is authorized to access that message.

Confidentiality: The security of our scheme heavily relies on preserving the confidentiality of private keys belonging to both the data owner and the data consumer (requester). It is imperative to protect these keys from potential attacks launched by adversaries. Furthermore, during the ciphertext equality test process, it is guaranteed that no sensitive information will be compromised or leaked.

Distribution: The scheme operates in a distributed manner, eliminating the dependence on centralized servers or certain third parties. With such a system architecture, this work can eliminate the assumption that the server must always be honest. This distributed architecture enhances the trustworthiness of our scheme by ensuring robustness and reliability.

By prioritizing correctness, confidentiality, and distribution, this work aims to build a secure and resilient system that fulfills the functional and security requirements of our design.

5. System construction

Our scheme incorporates a dedicated blockchain storage solution for pathological data and develops a sharing mechanism through the utilization of C-PRE.

5.1. Algorithm essentials

This subsection provides a comprehensive overview of the algorithmic intricacies underlying the scheme.

1) $\boldsymbol{Setup(k)\to(par)}$: The TrustCenter executes this algorithm and generates the necessary parameters through the following steps. Step 1: The algorithm begins by selecting a bilinear pairing group (G, G_T), in which both sets of order prime q. This group is chosen to satisfy the condition $e: G\times G\to G_T$. Step 2: In addition, the algorithm selects a group element $g\in G$ and incorporates five secure hash functions:

- $H_1: \{0,1\}^l \to Z_q^*$
- $H_2: G_T \to \{0,1\}^l$
- $H_3: G\times\{0,1\}^* \to G$
- $H_4: G\times\{0,1\}^l\times G \to G$
- $H_5: G_T \to G$

Assuming that the data space of plaintext is represented as $\{0,1\}^{l_1}$, the algorithm would produce a tuple of global parameters $(q, G, G_T, g, H_1, H_2, H_3, H_4, H_5, l, l_1)$ as its output.

2) $\boldsymbol{KeyGen(par)\to(pk_i, sk_i)}$: By selecting a random number $x_i\in Z_q^*$, this algorithm generates a public–private key pair $(pk_i, sk_i) = (g^{x_i}, x_i)$ for the user.

3) $\boldsymbol{ReKeyGen(par, sk_i, pk_j, w)\to rk}$: Using the data owner's private key sk_i, the data requester's public key pk_j, and under condition w, this algorithm randomly selects $s\in Z_q^*$ to generate a C-PRE key pair $rk = (rk_1, rk_2) = (pk_j^{\frac{1}{x_i}} H_3(pk_i, w)^s, pk_i^s)$.

4) $\boldsymbol{Enc_2(par, pk_i, m, w)\to C_i}$: With the public key pk_i, condition w, and plaintext $m\in\{0,1\}^{l_1}$, and choosing a configuration $\{r = H_1(m, r_1), r_1\in\{0,1\}^{l-l_1}, r_2, r_3\in Z_q^*$, a keyword $\theta\in G\}$, this algorithm computes the following values:

- $C_1 = pk_i^r$
- $C_2 = H_2(e(g,g)^r)\oplus(m||r_1)$
- $C_3 = H_3(pk_i, w)^r$
- $C_4 = H_4(C_1, C_2, C_3)^r$
- $C_5 = \theta^{r_2} H_5(e(pk_i^{r_2}, g)^{r_3})$
- $C_6 = g^{r_2}$
- $C_7 = g^{r_3}$

The final output is a second layer ciphertext denoted as $C_i = (C_1, C_2, C_3, C_4, C_5, C_6, C_7)$.

5) $\boldsymbol{Enc_1(par, pk_j, m)\to C_j}$: For plaintext m, using the requester's public key pk_j and a configuration $\{ r = H_1(m, r_1), r_1\in\{0,1\}^{l-l_1}\}$, this algorithm computes the following values:

- $C_1' = e(pk_j, g)^r$
- $C_2' = H_2(e(g,g)^r)\oplus(m||r_1)$

The resulting ciphertext $C_j = (C_1', C_2')$ is then outputted.

6) $\boldsymbol{Trapdoor(C, sk)\to td}$: Using a ciphertext C and the user's private key sk, calculate the user's authorization trapdoor td. For users i and j, whose trapdoors are represented as td_i and td_j, respectively, their authorization trapdoors are as follows:

- $td_i = C_{i,6}^{x_i}$
- $td_j = C_{j,6}^{x_j}$

7) $\boldsymbol{Test(C_i, C_j; td_i, td_j)}$: Using ciphertext pair (C_i, C_j) and trap pair (td_i, td_j) uploaded by user i and user j, respectively, the following equality tests are performed:

$$\begin{cases} V_i = \frac{C_{i,5}}{H_5(e(td_i, C_{i,7}))} \\ V_j = \frac{C_{j,5}}{H_5(e(td_j, C_{j,7}))} \end{cases} \tag{3}$$

If $e(C_{i,6}, V_j) = e(C_{j,6}, V_i)$ is true, returns 1; otherwise, it returns $\perp$.

8) $\boldsymbol{ReEnc(rk, C_i)\to C_{i\to j}}$: First, check if the following equations hold:

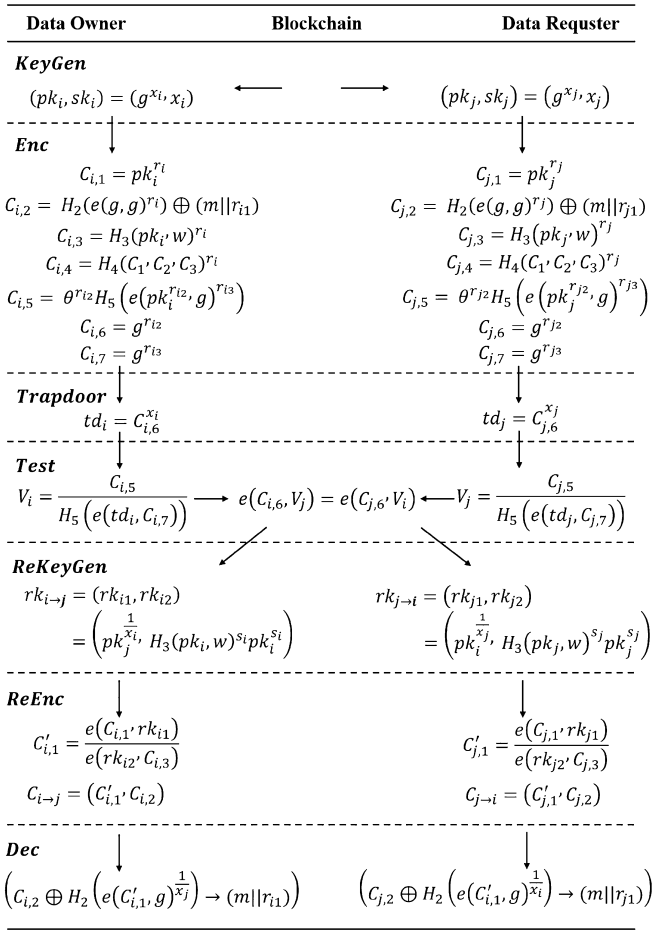

Fig. 2. The workflow of the pathological data blockchain sharing procedure based on conditional proxy re-encryption.

$$\begin{cases} e(C_1, H_3(pk_i, w)) = e(pk_i, C_3) \\ e(C_1, H_4(C_1, C_2, C_3)) = e(pk_i, C_4) \end{cases}$$

If not, it returns $\perp$; otherwise, it performs the following calculation with a re-encryption key rk and a ciphertext C_i:

$$C_1' = \frac{e(C_1, rk_1)}{e(rk_2, C_3)}$$

The resulting first layer ciphertext is $C_{i\to j} = (C_1', C_2)$.

9) $\boldsymbol{Dec_2(C_i, sk_i) \to m}$: A data owner uses his private key sk_i to decrypt the second layer ciphertext C_i to obtain the plaintext m. Specifically, it first calculates

$$(C_2 \oplus H_2(e(C_1, g)^{\frac{1}{x_i}}) \to (m||r_1))$$

and then determines whether ${pk_i}^r = C_1$ is true. If yes, it outputs plaintext m by extracting the first l_1 bits; otherwise, it returns $\perp$.

10) $\boldsymbol{Dec_1(C_{i\to j}, sk_j) \to m}$: A data requester uses his private key sk_j to decrypt the first layer ciphertext $C_{i\to j}$ to obtain the plaintext m. Specifically, it first calculates

$$(C_2 \oplus H_2(e(C_1', g)^{\frac{1}{x_j}}) \to (m||r_1))$$

and then determines whether ${pk_j}^r = C_1$ is true. If not, it returns $\perp$; otherwise, it outputs plaintext m by exextracting the first l_1 bits.

5.2. Pathological data sharing procedure

Fig. 2 presents a comprehensive overview of the sharing procedure for pathological data on the blockchain using C-PRE. The process begins with system initialization, followed by users generating their public–private key pairs using the *KeyGen()* algorithm. Subsequently, the data owners can encrypt their data using the *Enc()* algorithm and store data on the blockchain and IPFS. The *Trapdoor()* algorithm enables the generation of authorization trapdoors for data owners and data requesters, while the *Test()* algorithm performs a ciphertext matching process through the blockchain. Upon successful matching, data owners execute the *ReKeyGen()* algorithm to generate the C-PRE key. The blockchain network then employs the *ReEnc()* algorithm to convert the ciphertext and deliver it to data requesters. Finally, data requesters can utilize their private keys to decrypt the re-encrypted ciphertext and retrieve the shared information (pathological data).

5.3. Smart contract design

Algorithms 1 and 2 present the detailed design of the smart contract used in our proposed scheme. Within the smart contract, three structures are defined: *Cipher_Tuple*, *Role_Info*, and *Test_Tuple*. Additionally, three methods are implemented: *Query()*, *Insert()*, and *Test()*.

Algorithm 1 The structure of smart contract.

```
Structure Cipher_Tuple {
    C_1, C_2, C_3, C_4, C_5, C_6, C_7 string
}
Structure Role_Info {
    ID string
    Public_key string
}
Structure Test_Tuple {
    Role string
    C Cipher_Tuple
    Trapdoor string
}
```

Algorithm 2 The functions of smart contract.

```
//-------------------------------------------------------
Query(key string):
    return D[key] or null
//-------------------------------------------------------
Insert(id, data string,flag int):
    cdata Cipher_Tuple
    if flag == 0 then
        //Convert data to Cipher_Tuple object
        cdata = toCipherTuple(data)
        //Upload cdata to the blockchain
        uploadCipherTuple(cdata)
    else
        //Update cdata by id
        updataCipherTuple(id)
    end if
//-------------------------------------------------------
Test(cdatai, cdataj, tdi, tdj string, ti, tj Test_Tuple):
    if ti == null or tj == null then
        return null
    end if
    Parse ti.C into (Ci1, Ci2, Ci3, Ci4, Ci5, Ci6, Ci7)
    Parse tj.C into (Cj1, Cj2, Cj3, Cj4, Cj5, Cj6, Cj7)
    //Computes Vi and Vj
    Vi = Ci5/H5(e(tdi, Ci7))
    Vj = Cj5/H5(e(tdj, Cj7))
    if e(Ci6, V[j]) = e(Cj6, Vi) then
        return 1
    else
        return null
    end if
```

The *Cipher_Tuple* structure represents a tuple that stores ciphertext obtained from the *Enc()* operation. It is utilized for storing encrypted data. The *Role_Info* structure captures the role information of a user, including his ID and public key. On the other hand, the *Test_Tuple* structure contains information related to the ciphertext equality test, including the user's role (data owner or data requester), the ciphertext to be tested, and the authorization trapdoor.

The *Query()* function is responsible for searching and retrieving data from the blockchain based on a given keyword. It enables data retrieval operations. The *Insert()* function handles the insertion of the user's ciphertext tuple into blockchain for storage. If the data item already exists, this method performs an update operation. *Test()* performs ciphertext equality testing to determine whether two ciphertexts exactly match. This is necessary for the re-encryption process.

These structures and methods collectively contribute to the functionality of the smart contract, allowing for efficient data storage, retrieval, and ciphertext matching on the blockchain.

6. Security proof

This section proves the fulfillment of our system's design goals, namely, correctness, confidentiality, and distribution.

6.1. Correctness of data sharing

To verify the correctness of the equality test, we infer the values of V_i and V_j and ensure that the equation $e(C_{i,6}, V_j) = e(C_{j,6}, V_i)$ is satisfied. Referring to the definitions shown in Equation (3), we can make the following inference:

$$\begin{aligned} V_i &= \frac{C_{i,5}}{H_5(e(td_i, C_{i,7}))} \\ &= \frac{\theta_i^{r_{i2}} H_5(e(pk_i^{r_{i2}}, g)^{r_{i3}})}{H_5(e(g^{r_{i2}x_i}, g^{r_{i3}}))} \\ &= \frac{\theta_i^{r_{i2}} H_5(e(pk_i^{r_{i2}}, g)^{r_{i3}})}{H_5(e(g^{r_{i2}x_i}, g^{r_{i3}}))} \\ &= \frac{\theta_i^{r_{i2}} H_5(e(g,g)^{r_{i2}x_i r_{i3}})}{H_5(e(g,g)^{r_{i2}x_i r_{i3}})} \\ &= \theta_i^{r_{i2}}, \end{aligned} \tag{4}$$

$$\begin{aligned} V_j &= \frac{C_{j,5}}{H_5(e(td_j, C_{j,7}))} \\ &= \frac{\theta_j^{r_{j2}} H_5(e(pk_j^{r_{j2}}, g)^{r_{j3}})}{H_5(e(g^{r_{j2}x_j}, g^{r_{j3}}))} \\ &= \frac{\theta_j^{r_{j2}} H_5(e(pk_j^{r_{j2}}, g)^{r_{j3}})}{H_5(e(g^{r_{j2}x_j}, g^{r_{j3}}))} \\ &= \frac{\theta_j^{r_{j2}} H_5(e(g,g)^{r_{j2}x_j r_{j3}})}{H_5(e(g,g)^{r_{j2}x_j r_{j3}})} \\ &= \theta_j^{r_{j2}}. \end{aligned} \tag{5}$$

To verify that the re-encrypted ciphertext obtained in the first layer is identical to the ciphertext obtained by the data requester using his private key, we provide the following proof.

$$\begin{aligned} C_1' &= \frac{e(C_1, rk_1)}{e(rk_2, C_3)} \\ &= \frac{e(pk_i^r, pk_j^{\frac{1}{x_i}}) e(pk_i^r,\ H_3\left(pk_i, w\right)^s)}{e(pk_i^s, H_3\left(pk_i, w\right)^r)} \\ &= e(pk_j, g)^r. \end{aligned} \tag{6}$$

6.2. Confidentiality of the system

Confidentiality can be understood in two aspects. The first aspect relates to the inability to extract the original plaintext message from the information contained in the ciphertext. The second aspect pertains to the authorized trapdoor, which should also not reveal any information about the corresponding plaintext message.

To establish the confidentiality of our proposed scheme, two distinct (types of) games are introduced: a ciphertext indistinguishability game and an authorization trapdoor privacy game. These games involve the interaction between the adversary $\mathcal{A}$ and the challenger C. By analyzing the outcomes of these games, we can demonstrate the confidentiality guarantees.

6.2.1. Ciphertext indistinguishability

The major execution steps of these games are listed below.

(a) ***Setup***: The challenger C initiates the process by selecting a secure parameter k and executing the $Setup()$ algorithm to generate a public parameter par. Subsequently, the $KeyGen()$ algorithm is invoked to generate public–private key pairs for the data owner and the data requester, denoted as (pk_i, sk_i) and (pk_j, sk_j), respectively. The tuple (par, pk_i, pk_j) is then transmitted to the adversary $\mathcal{A}$.

(b) ***Phase1***: The challenger C has the ability to flexibly initiate different query operations to the following oracles, which is under the control of a challenger.

KeyGen oracle $\mathcal{O}_k$: By selecting a random number $x_j \in Z_q^*$, the algorithm $KeyGen$ generates a public–private key pair $(pk_j, sk_j) = (g^{x_j}, x_j)$ for the adversary $\mathcal{A}$.

ReKeyGen oracle $\mathcal{O}_{rk}$: Using private key sk_i, public key pk_j, and under condition w, the algorithm $ReKeyGen$ randomly selects $s \in Z_q^*$ to generate a C-PRE key rk for the adversary $\mathcal{A}$.

Enc oracle $\mathcal{O}_e$: With a public key pk_i, this oracle executes the Enc algorithm to produce a ciphertext $C_i(m)$ for a given message m and sends it to the adversary $\mathcal{A}$.

ReEnc oracle $\mathcal{O}_{re}$: With a re-encryption key rk and a ciphertext $C_i(m)$, this oracle executes the $ReEnc$ algorithm to produce a first layer ciphertext $C_{i\to j}$ and sends it to the adversary $\mathcal{A}$.

Dec oracle $\mathcal{O}_d$: With a private key sk_j, this oracle executes the Dec_1 algorithm to decrypt the first layer ciphertext $C_{i\to j}$ and obtain the plaintext m for the adversary $\mathcal{A}$.

(c) ***Challenge***: $\mathcal{A}$ selects two messages m_0 and m_1 for the challenge. The challenger C randomly chooses a bit $b \in \{0, 1\}$, computes the ciphertext $C_i(m_b) = Enc(par, pk_i, m_b, w)$, and sends $C_i(m_b)$ to $\mathcal{A}$.

(d) ***Phase2***: $\mathcal{A}$ can continue to make queries to the oracles as described above, with the restriction that neither m_0 nor m_1 can be submitted to the oracle.

(e) ***Guess***: $\mathcal{A}$ outputs a bit $b' \in \{0,1\}$, he wins the game if and only if $b' = b$. The advantage of $\mathcal{A}$ in winning the game is defined as follows:

$$Adv_{\mathcal{A}}^{IND} = \left| \mathcal{P}[b' = b] - \frac{1}{2} \right|.$$

Theorem 1. *Under the assumption that the 3-wDBDHI assumption holds and the hash functions used in our scheme are collision-resistant under the random oracle model, our scheme achieves ciphertext indistinguishability.*

Proof 1. We establish the proof by conducting a sequence of games, where each game builds upon the previous one with slight modifications, while maintaining indistinguishability for the adversary. In the final game, the ciphertext becomes completely independent of the original message and contains no information about the underlying plaintext message.

Game 1: This game demonstrates the ciphertext indistinguishability as mentioned above. The challenger C initiates the game by generating the common parameter $par = (q, G, G_T, g, H_1, H_2, H_3, H_4, l, l_1)$ and two pairs of public–private keys $(pk_i, sk_i) = (g^{x_i}, x_i)$ and $(pk_j, sk_j) = (g^{x_j}, x_j)$. The public information (par, pk_i, pk_j) is published, while the private keys $(sk_i = x_i, sk_j = x_j)$ are kept secret. By querying the ciphertext oracle, C generates a ciphertext

$$C_i(m) = \begin{cases} C_{i1} = pk_i^r \\ C_{i2} = H_2(e(g,g)^r) \oplus (m||r_1) \\ C_{i3} = H_3(pk_i, w)^r \\ C_{i4} = H_4\left(C_1, C_2, C_3\right)^r \\ C_{i5} = \theta^{r_2} H_5(e\left(pk_i^{r_2}, g\right)^{r_3}) \\ C_{i6} = g^{r_2} \\ C_{i7} = g^{r_3} \end{cases}$$

corresponding to the message m, where $r_1 \in \{0,1\}^{l-l_1}$, $r = H_1(m, r_1)$, $r_2, r_3 \in Z_q^*$, and $\theta \in G$. By querying the authorized trapdoor oracle, C calculates the trapdoor $td_i = C_{i6}^{x_i}$ corresponding to the ciphertext $C_i(m)$. This process simulates the adversary's ability to obtain a trapdoor. By repeating the queries, the adversary $\mathcal{A}$ attempts to guess the challenge ciphertext based on the available information, and its advantage in winning the game is defined as:

$$Adv_{\mathcal{A}}^{Game1} = Adv_{\mathcal{A}}^{IND}.$$

Game 2: This game is similar to Game 1 but with a slight modification. Instead of using standard hash functions H_1, all hash functions are modeled as random oracles. In this game, when the challenger C receives an input x, he first checks the hash list and returns the corresponding value y if it exists. If there is no value corresponding to x, the challenger C randomly selects a value and sets $H_1(x) = y$. The purpose of this modification is to ensure that this game remains indistinguishable, as otherwise, an adversary could exploit any distinguishability to compromise the security of the hash function. Therefore, the challenger C wins this game with the same advantage as $\mathcal{A}$ in Game 1:

$$Adv_C^{Game2} = Adv_{\mathcal{A}}^{Game1}.$$

Game 3: This game is identical to Game 2, with the only difference being that the challenge ciphertext is not considered during the encryption phase of $C_{i2} = H_2(e(g,g)^r) \oplus m||r_1$. Instead, the challenger C selects a random string T to calculate a new challenge ciphertext $C_{i2}^* = H_2(T) \oplus m||r_1$. The goal of this modification is to determine if the adversary $\mathcal{A}$ can distinguish between the original ciphertext C_{i2} and the modified ciphertext C_{i2}^*.

If $\mathcal{A}$ can distinguish, it implies that $\mathcal{A}$ can determine the relationship between T and $e(g,g)^{\frac{b}{a^2}}$ based on the tuple $(g, g^{\frac{1}{a}}, g^a, g^{a^2}, g^b)$, where $a, b \in Z_q^*$. However, from the adversary's perspective, the tuple $(g, g^{\frac{1}{a}}, g^a, g^{a^2}, g^b)$ forms a 3-wDBDHI problem, as (a, b) is unknown and random. Therefore, the adversary cannot distinguish between the original ciphertext C_{i2} and the modified ciphertext C_{i2}^*. In conclusion, the challenger C's advantage in winning this game is as follows:

$$|Adv_C^{Game3} - Adv_C^{Game2}| \leq Adv_{\mathcal{A}}^{3-wDBDHI}.$$

Game 4: In this game, the challenger C selects a random string $C_{i2}^{**} \leftarrow \{0,1\}^*$ as a replacement for the calculation $C_{i2}^* = H_2(T) \oplus (m||r_1)$. Since the randomness of T, thus C_{i2}^{**} and C_{i2}^* are indistinguishable for $\mathcal{A}$, our advantage in winning this game remains the same as that in Game 3, denoted as:

$$Adv_C^{Game4} = Adv_C^{Game3}.$$

Meanwhile, it is observed that the challenge ciphertext is completely independent of the message m. As a result, the probability of adversary $\mathcal{A}$ winning in the game is $1/2$. Therefore, our advantage in this game is $Adv_{\mathcal{A}}^{Game4} = |1/2 - 1/2| = 0$, since

$$\begin{cases} Adv_C^{Game4} = 0 \\ Adv_{\mathcal{A}}^{IND} \leq Adv_{\mathcal{A}}^{3-wDBDHI} \end{cases}$$

where $Adv_{\mathcal{A}}^{3-wDBDHI}$ is negligible as 3-wDBDHI holds.

6.2.2. Trapdoor indistinguishability

Theorem 2. *Under the assumption that the BDDH assumption holds and the hash function is collision-resistant under the random oracle model, our scheme can achieve trapdoor indistinguishability.*

Analysis: In this scenario, the adversary $\mathcal{A}$ possesses additional trapdoor information corresponding to the ciphertext. Our objective is to demonstrate that this trapdoor information does not provide any advantage to the adversary in deducing the plaintext message. In other words, the trapdoor is designed to be independent of the plaintext and should not reveal any information about its content.

Proof 2. Given the trapdoor $td_i = C_{i6}^{x_i}$ associated with ciphertext $C_i(m)$, adversary $\mathcal{A}$ can compute an intermediate ciphertext $V_i = {C_{i,5}}/{H_5\left(e\left(td_i, C_{i,7}\right)\right)}$. Because of the randomness of r_2 and r_3, the adversary is unable to determine the relationship between T and $e(g,g)^{abc}$ based on the values in the tuple (g, g^a, g^b, g^c, T). Consequently, even if the adversary possesses the trapdoor information, he cannot gain any knowledge about the plaintext through a keyword guessing attack.

6.3. Distribution property

In our solution, the test algorithm is deployed on the blockchain as a smart contract rather than being implemented on a centralized server. This deployment ensures that the test results are publicly available and verifiable by others. The consensus mechanism of the blockchain ensures the correct execution of each test operation, providing a trustworthy and reliable environment. Moreover, the nature of blockchain technology enables multiple participants to access and contribute to the system in a distributed fashion.

6.4. Fault tolerance

The scheme in this paper is based on the Hyperledger Fabric platform, and the consensus mechanism adopted is PBFT. It allows no more than $f/3 - 1$ malicious nodes to exist in the system (f is the number of all nodes in the network). When all kinds of nodes in the blockchain network meet the above rules, even if some malicious nodes exist, they still cannot affect the transactions in the network.

7. Evaluation

This section compares the proposed scheme with four related articles in terms of practicality and communication cost. Additionally, to demonstrate its practicality, we have implemented a prototype system on a consortium chain and evaluated its performance with various metrics.

7.1. Practicality feature comparison

To assess the practicality of the solution, we focus on comparing four important characteristics: distribution, flexibility, equality testing, and C-PRE.

1) **Distribution**: A distributed scheme eliminates the reliance on centralized servers. This not only mitigates the risk of system failure due to a single node failure but also reduces management costs.

2) **Flexibility**: A flexible scheme allows for versatile authorization and sharing mechanisms, reducing the need for complex intermediate steps.

3) **Equality test**: An equality test scheme facilitates the matching of ciphertext between data requesters and data owners, effectively preserving user privacy.

Table 2
Comparison of key features of practicality among schemes.

Schemes	Distribution	Flexibility	Equality test	C-PRE
He et al. [26]	✓	×	×	×
Isshiki et al. [27]	×	✓	×	×
Yao et al. [28]	×	✓	×	✓
Chen et al. [29]	×	✓	✓	×
This work	✓	✓	✓	✓

4) **C-PRE:** This feature enhances the protection of users' sensitive data by restricting access to shared information to semi-honest third parties.

By comparing the proposed scheme with existing literature in these aspects, we can assess its practicality and determine its advantages and disadvantages.

Table 2 provides a comprehensive comparison of our scheme with He et al. [26], Isshiki et al. [27], Yao et al. [28], and Chen et al. [29] in terms of four key practicality characteristics.

As illustrated, our scheme outperforms the mentioned schemes in terms of the key features of practicality. While He et al.'s scheme exhibits distributed and PRE features, it lacks flexible authorization and equality test advantages. Additionally, although PRE is employed, it lacks conditions to control the re-encryption scope and safeguard data owner privacy. Isshiki et al.'s scheme does not utilize blockchain technology, equality tests, or other relevant techniques. It achieves some degree of flexibility through the PRE scheme but fails to incorporate conditions for protecting sensitive data. Yao et al.'s scheme achieves flexible authorization sharing and incorporates C-PRE technology, offering partial privacy protection for patients and enhancing sharing efficiency. However, it does not leverage blockchain technology or equality test techniques, making it vulnerable to single point attacks and potential privacy issues during plaintext searches. Chen et al.'s scheme utilizes equality test techniques to prevent privacy leakage during searches and provides flexibility. However, it does not utilize distributed blockchain technology or C-PRE. In comparison, our scheme combines all four key features of practicality, making it a robust solution for privacy-preserving data sharing and access control.

Our proposed scheme represents a significant advancement in the secure and efficient sharing of pathological data by seamlessly integrating blockchain, equality test, and C-PRE technology. This integration enhances the flexibility of data sharing, safeguards user privacy, controls the scope of re-encryption, and ensures high practicality.

Furthermore, our scheme is not limited to pathological and medical scenarios. Its applicability extends to various domains that demand secure and efficient data sharing while maintaining privacy. This versatility makes our scheme suitable for a wide range of industries and use cases.

7.2. Communication cost comparison

Table 3 summarizes the communication costs of our proposed scheme and the four articles. The communication costs are represented by various operations and parameters: t_p represents the bilinear pairing operation, t_s represents the scalar multiplication operation on group G, t_{e1} and t_{e2} represent the power operations on groups G and G_T, respectively, and t_{me} represents the double exponential operation. In the keyword search scheme, k is used to represent the size of the keyword set, s for the attribute of the data user, and l for the number of properties of the accessed structures.

In terms of the communication costs, our scheme demonstrates favorable efficiency in various aspects.

1) **Encryption** $Enc()$: Our scheme performs efficiently in the encryption stage, which is crucial for a smooth user experience. It outperforms Yao et al.'s scheme in terms of encrypted communication cost, is comparable to He et al.'s and Isshiki et al.'s schemes in efficiency, and exhibits lower costs compared with Chen et al.'s scheme. Essentially, the efficiency of our scheme is achieved by incorporating relevant content in the ciphertext for subsequent trapdoor generation and equality tests, striking a balance between functionality and efficiency.

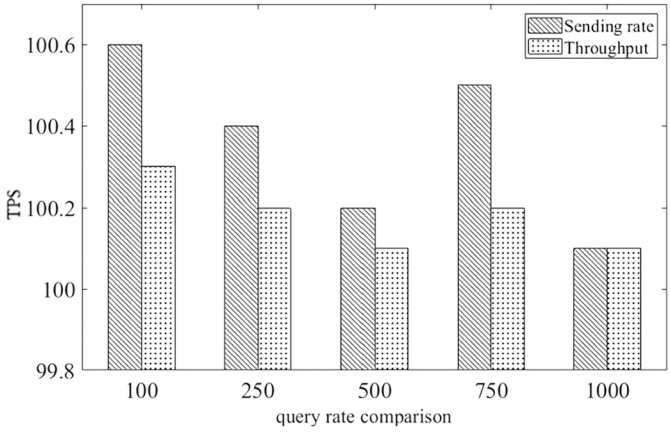

Fig. 3. Comparison of sending rate and throughput of query operations among schemes.

2) **First-layer Decryption** $Dec_1()$ and **Second-layer Decryption** Dec_2: Our solution significantly reduces the communication cost in the first-layer decryption algorithm compared with other schemes, enhancing the efficiency of data requesters in decrypting and accessing data. Moreover, the communication cost of the second-layer decryption algorithm in our scheme is considerably lower than that of Isshiki et al.'s scheme.

3) **Trapdoor Generation** $Trapdoor()$ and **Equality Test** $Test()$: Our scheme exhibits lower communication costs in the trapdoor generation and equality test algorithms compared with Chen et al.'s scheme, while being competitive with the other three schemes.

In summary, our scheme strikes a favorable balance between practicality and communication costs, making it an efficient solution for privacy-preserving data sharing and access control.

7.3. Performance evaluation

We evaluated our proposed scheme by implementing it on a consortium chain. These evaluations provide valuable insights into the efficiency and scalability of our solution, allowing us to assess its practical viability.

To carry out the evaluations, we developed a prototype system and deployed it on Hyperledger Fabric 1.4.8 using our customized smart contracts. Hyperledger Caliper, a blockchain performance benchmarking tool, was employed to test the performance of our prototype system. The evaluations were conducted on an Ubuntu 20.04 system.

During the testing process, we conducted multiple rounds of testing with varying trading volumes: 100, 250, 500, 750, and 1000. To ensure accuracy, we removed the maximum and minimum values and calculated the average of each result. The test results are presented in Figs. 3–6, providing a comprehensive overview of the performance of our proposed scheme under different trading volumes.

As depicted in Fig. 3, the sending rate and throughput of query operations exhibit a relatively stable trend with a small fluctuation amplitude, maintaining an average of around 100 transactions per second (TPS). This indicates that our system can handle query operations efficiently, even with varying transaction volumes.

Furthermore, as shown in Fig. 4, the delay of query operations slightly increases as the transaction volume increases. However, the delay remains within 0.05 s, which is negligible and does not significantly impact overall platform operations. This suggests that our system

Table 3
Comparison of communication costs according to various operations and parameters.

Schemes	$Enc()$	$ReEnc()$	$Dec_2()$	$Dec_1()$	$Trapdoor()$	$Test()$
He et al. [26]	$t_p + 3t_{e1}$	$2t_p + t_{e1}$	-	$2t_p$	-	-
Isshiki et al. [27]	$t_p + 7t_s + ENC$	$4t_p + 5t_s + ENC$	$8t_p + 3t_s + t_{e2} + DEC$	$5t_p + t_s + t_{e2}$	-	-
Yao et al. [28]	$8t_{e1} + t_p$	$6t_{e1}$	-	$t_{e1} + t_p$	-	-
Chen et al. [29]	$(2l+2)t_{e1} + 2t_{e2}$	$2t_p + l(2t_p + t_{e2})$	-	-	$(s+4)t_{e2}$	$(2l+2)t_p + kt_{e2}$
This work	$t_p + 6t_{e1} + t_{e2}$	$4t_p + 2t_{me}$	$2t_p + 2t_{me} + 2t_{e1}$	$2t_{e1}$	t_{e1}	$4t_p$

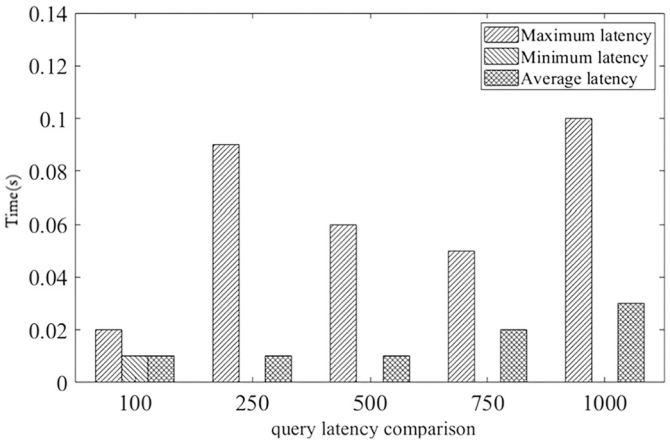

Fig. 4. Comparison of the latency of query operations among schemes.

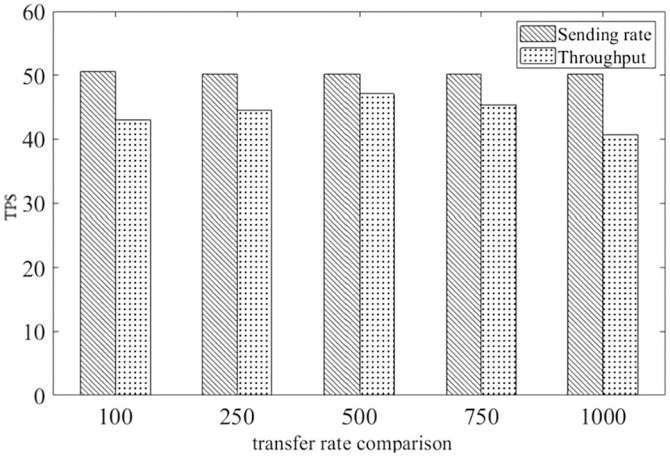

Fig. 5. Comparison of sending rate and throughput of transfer operations among schemes.

is capable of processing queries promptly, ensuring a satisfactory user experience.

Fig. 5 demonstrates the sending rate and throughput of transmission operations as the transaction volume increases from 100 to 1000. The sending rate and throughput exhibit a stable pattern, averaging around 40–50 TPS. This indicates that our system maintains a consistent performance in handling transmission operations, regardless of the transaction volume.

Additionally, Fig. 6 presents the delay of transmission operations. The maximum delay remains between 0.2 s and 0.3 s, while the minimum delay remains below 0.05 s. The average delay ranges from 0.1 s to 0.15 s. These results suggest that our system can efficiently process transmission operations, ensuring timely delivery of data with acceptable delays.

Based on the presented results, we can confidently conclude that our platform demonstrates stable performance and is well-equipped to meet the requirements of various scenarios. The sending rate, throughput,

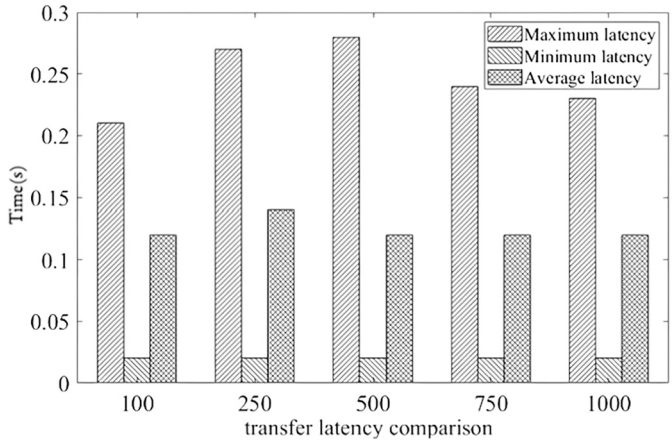

Fig. 6. Comparison of the latency of transfer operations among schemes.

and delay of both query and transmission operations remain within acceptable ranges as the transaction volume increases. This indicates that our solution can effectively handle a significant number of transactions without compromising performance. The stability and efficiency of our platform make it a reliable choice for a wide range of applications and scenarios.

8. Conclusion

Network-based sharing is considered an ideal solution to facilitate broad utilization of pathological data. Given the privacy-sensitive nature of pathological data and its potential use in remote diagnosis, the confidentiality of the sharing scheme and the reliability of the sharing nodes are critical requirements.

In this work, the reliability challenge is addressed by leveraging the capabilities of blockchain technology. Meanwhile, by designing a sharing scheme based on C-PRE, the transmission and sharing process ensures that plaintext data remain confidential. Thus, the privacy of patient information is preserved, and pathological data can be securely shared among authorized data owners and requesters, with access controlled based on specified user-permissions and data-attributes. Additionally, the efficiency of our system is enhanced through the use of ciphertext equality tests.

To evaluate the practicality of the solution, we implemented a prototype using Hyperledger Fabric. The evaluation results confirm that this solution shows both efficiency and security, demonstrating its viability for real-world applications.

CRediT authorship contribution statement

Wei Wu is the lead author of the article, completing the collection and analysis of relevant literature and the writing of the first draft of the paper; Fulong Chen is the author and director of the article, directing thesis writing; Chao Wang, Detao Tang and Jingtao Li participated in the analysis and collation of literature; Pinghai Yuan and Ji Zhang performed writing, review, and validation. Taochun Wang, Xie Dong, Chuanxin Zhao participated in the proofreading and finalization of the article. All authors read and agree with the final text.

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.

Funding

This research is partially supported by National Natural Science Foundation of China under Grant 61972438, Wuhu Science and Technology Plan Project under Grant 2022yf50, Key Research and Development Projects in Anhui Province under Grant 202004a05020002 and 2022a05020049.

References

[1] F. Chen, Y. Tang, X. Cheng, et al., Blockchain-based efficient device authentication protocol for medical cyber-physical systems, Secur. Commun. Netw. 2021 (2021), https://doi.org/10.1155/2021/5580939.

[2] F. Chen, Y. Tang, C. Wang, et al., Medical cyber-physical systems: a solution to smart health and the state of the art, IEEE Trans. Comput. Soc. Syst. 9 (5) (2022) 1359–1386, https://doi.org/10.1109/TCSS.2021.3122807.

[3] A. Kumari, V. Kumar, M.Y. Abbasi, EAAF: ECC-based anonymous authentication framework for cloud-medical system, Int. J. Comput. Appl. 44 (5) (2022) 491–500, https://doi.org/10.1080/1206212x.2020.1815334.

[4] T. Adnan, C. Fei, U. Habib, et al., A systematic review on cloud storage mechanisms concerning e-healthcare systems, Sensors 20 (18) (2020) 5392, https://doi.org/10.3390/s20185392.

[5] M. Mbarek, K. Ali, O. Hassan, A framework to secure medical image storage in cloud computing environment, J. Electron. Commer. Organ. 16 (1) (2018) 1–16, https://doi.org/10.4018/jeco.2018010101.

[6] D. Sharad, P.J. Karuna, G.C. Seung, Multi authority access control in a cloud ehr system with ma-abe, in: Proceedings of the 2019 IEEE International Conference on Edge Computing (EDGE), IEEE, 2019, pp. 107–109, https://doi.org/10.1109/EDGE.2019.00032.

[7] R. Zhang, R. Xue, L. Liu, Searchable encryption for healthcare clouds: a survey, IEEE Trans. Serv. Comput. 11 (6) (2017) 978–996, https://doi.org/10.1109/TSC.2017.2762296.

[8] G. Peng, A. Zhang, X. Lin, Patient-centric fine-grained access control for electronic medical record sharing with security via dual-blockchain, IEEE Trans. Netw. Sci. Eng. 10 (6) (2023) 3908–3921, https://doi.org/10.1109/TNSE.2023.3276166.

[9] S. Xu, J. Ning, X. Li, et al., A privacy-preserving and redactable healthcare blockchain system, IEEE Trans. Serv. Comput. 17 (2) (2024) 364–377, https://doi.org/10.1109/TSC.2024.3356595.

[10] M. Blaze, B. Gerrit, S. Martin, Divertible protocols and atomic proxy cryptography, in: K. Nyberg (Ed.), Advances in Cryptology – EUROCRYPT'98, Springer, Berlin, Heidelberg, 1998, pp. 127–144, https://doi.org/10.1007/BFb0054122.

[11] Y. Yang, M. Ma, Conjunctive keyword search with designated tester and timing enabled proxy re-encryption function for e-health clouds, IEEE Trans. Inf. Forensics Secur. 11 (4) (2015) 746–759, https://doi.org/10.1109/TIFS.2015.2509912.

[12] W. Li, C. Jin, S. Kumari, et al., Proxy re-encryption with equality test for secure data sharing in Internet of things-based healthcare systems, Trans. Emerg. Telecom. Technol. 33 (10) (2022) e3986, https://doi.org/10.1002/ett.3986.

[13] Q. Tang, Type-based proxy re-encryption and its construction, in: D.R. Chowdhury, V. Rijimen, A. Das (Eds.), Progress in Cryptology - INDOCRYPT 2008, Springer, Berlin, Heidelberg, 2008, pp. 130–144, https://doi.org/10.1007/978-3-540-89754-5_11.

[14] J. Weng, H.D. Robert, X. Ding, et al., Conditional proxy re-encryption secure against chosen-ciphertext attack, in: Proceedings of the 4th International Symposium on Information, Computer, and Communications Security, ACM, 2008, pp. 322–332, https://doi.org/10.1145/1533057.1533100.

[15] Fimiani, Gianluca, Supporting privacy in a cloud-based health information system by means of fuzzy conditional identity-based proxy re-encryption (fci-pre), in: Proceedings of the 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), IEEE, 2018, pp. 569–572, https://doi.org/10.1109/WAINA.2018.00146.

[16] G. Yang, C.H. Tan, Q. Huang, et al., Probabilistic public key encryption with equality test, in: J. Pieprzyk (Ed.), Topics in Cryptology - CT-RSA 2010, Springer, Berlin, Heidelberg, 2010, pp. 119–131, https://doi.org/10.1007/978-3-642-11925-5_9.

[17] T. Qiang, Public key encryption schemes supporting equality test with authorisation of different granularity, Int. J. Appl. Cryptogr. 2 (4) (2012) 304–321, https://doi.org/10.1504/ijact.2012.048079.

[18] Y. Ling, S. Ma, Q. Huang, et al., Group public key encryption with equality test against offline message recovery attack, Inf. Sci. 510 (2020) 16–32, https://doi.org/10.1016/j.ins.2019.09.025.

[19] S. Alornyo, Y. Zhao, G. Zhu, et al., Identity based key-insulated encryption with outsourced equality, Int. J. Netw. Secur. 22 (2) (2020) 257–264, https://doi.org/10.6633/IJNS.202003_22(2).09.

[20] P.S. Roy, D.H. Duong, W. Susilo, et al., Lattice-based public key encryption with equality test supporting flexible authorization in standard model, Theor. Comput. Sci. 929 (2020) 124–139, https://doi.org/10.1016/j.tcs.2022.06.034.

[21] F. Li, K. Liu, L. Zhang, et al., EHRChain: a blockchain-based ehr system using attribute-based and homomorphic cryptosystem, IEEE Trans. Serv. Comput. 15 (5) (2021) 2755–2765, https://doi.org/10.1109/TSC.2021.3078119.

[22] G. Wu, S. Wang, Z. Ning, et al., Blockchain-enabled privacy-preserving access control for data publishing and sharing in the Internet of medical things, IEEE Int. Things J. 9 (11) (2021) 8091–8194, https://doi.org/10.1109/JIOT.2021.3138104.

[23] W. Chen, S. Zhu, J. Li, et al., Authorized shared electronic medical record system with proxy re-encryption and blockchain technology, Sensors 21 (22) (2021) 7765, https://doi.org/10.3390/s21227765.

[24] D. Boneh, M. Franklin, Identity-based encryption from the Weil pairing, SIAM J. Comput. 32 (3) (2003) 586–615, https://doi.org/10.1137/s0097539701398521.

[25] B. Libert, D. Vergnaud, Unidirectional chosen-ciphertext secure proxy re-encryption, IEEE Trans. Inf. Theory 57 (3) (2011) 1786–1802, https://doi.org/10.1109/TIT.2011.2104470.

[26] J. He, D. Zheng, R. Guo, et al., Efficient identity-based proxy re-encryption scheme in blockchain-assisted decentralized storage system, Int. J. Netw. Secur. 23 (5) (2021) 776–790, https://doi.org/10.6633/IJNS.202109_23(5).05.

[27] T. Isshiki, H.N. Manh, T. Keisuke, Proxy re-encryption in a stronger security model extended from ct-rsa2012, in: E. Dawson (Ed.), Topics in Cryptology – CT-RSA 2013, Springer, Berlin, Heidelberg, 2013, pp. 277–292, https://doi.org/10.1007/978-3-642-36095-4_18.

[28] S. Yao, R.V.J. Dayot, H.-J. Kim, et al., A novel revocable and identity-based conditional proxy re-encryption scheme with ciphertext evolution for secure cloud data sharing, IEEE Access 9 (2021) 42801–42816, https://doi.org/10.1109/ACCESS.2021.3064863.

[29] Y. Chen, Y. Hu, M. Zhu, et al., Attribute-based keyword search with proxy re-encryption in the cloud, IEICE Trans. Commun. 101 (8) (2018) 1798–1808, https://doi.org/10.1587/transcom.2017ebp3274.