Journal of Cloud Computing:
Advances, Systems and Applications

RESEARCH

Open Access

A secure and efficient electronic medical record data sharing scheme based on blockchain and proxy re-encryption

Guijiang Liu[1], Haibo Xie[1], Wenming Wang[1,3,4*] and Haiping Huang[2]

Abstract

With the rapid development of the Internet of Medical Things (IoMT) and the increasing concern for personal health, sharing Electronic Medical Record (EMR) data is widely recognized as a crucial method for enhancing the quality of care and reducing healthcare expenses. EMRs are often shared to ensure accurate diagnosis, predict prognosis, and provide health advice. However, the process of sharing EMRs always raises significant concerns about potential security issues and breaches of privacy. Previous research has demonstrated that centralized cloud-based EMR systems are at high risk, e.g., single points of failure, denial of service (DoS) attacks, and insider attacks. With this motivation, we propose an EMR sharing scheme based on a consortium blockchain that is designed to prioritize both security and privacy. The interplanetary file system (IPFS) is used to store the encrypted EMR while the returned hash addresses are recorded on the blockchain. Then, the user can authorize other users to decrypt the EMR ciphertext via the proxy re-encryption algorithm, ensuring that only authorized personnel may access the files. Moreover, the scheme attains personalized access control and guarantees privacy protection by employing attribute-based access control. The safety analysis shows that the designed scheme meets the expected design goals. Security analysis and performance evaluation show that the scheme outperforms the comparison schemes in terms of computation and communication costs.

Keywords Blockchain, Proxy re-encryption, EMR sharing, IPFS, Data security

Introduction

An electronic medical record (EMR) is a computerized version of a patient's previously paper-based medical records, encompassing medical history, lab results, and records of diagnosis and treatment [1]. The implementation of EMRs in medical institutions has enabled cross-regional accessibility of patient data, improved the quality of patient care, and reduced time costs. In addition, EMR sharing assists doctors in making more precise diagnoses and helps researchers develop new drugs or vaccines [2–4]. Therefore, EMRs are increasingly considered vital for the advancement of medical information [5].

Although electronic medical records have greatly improved healthcare, they still face many challenges when it comes to practical application. One inevitable challenge is the increased risk of medical data breaches in the EMR system when sharing or trading EMRs between medical organizations. Due to the inherently open nature of wireless channels, sensitive patient information such as addresses, ID numbers,

*Correspondence:
Wenming Wang
wy523@aqnu.edu.cn
[1] School of Computer and Information, Anqing Normal University, Anqing 246133, China
[2] School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China
[3] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
[4] Yunnan Key Laboratory of Service Computing, Yunnan University of Finance and Economics, Kunming 650221, China

and physiological data may be eavesdropped, tampered with, or disrupted by malicious attackers [6]. Furthermore, these exposed data can be traded for significant illegal gains. Additionally, many healthcare providers store EMRs on internal servers for quick access, which may be vulnerable to unexpected corruption or natural disasters. To address these complex problems, many cloud-based EMR systems [7, 8] have been introduced in recent years.

In general, cloud-based EMR systems encrypt patients' EMRs and establish relevant access control policies before outsourcing them to the cloud. Users with different identity attributes then submit encrypted search keywords to the cloud server. By searching for keywords, the cloud server retrieves the ciphertext of the corresponding EMR and sends it to the user. Finally, users who satisfy the attribute requirements can successfully decrypt the ciphertext to access the appropriate EMR. This process demonstrates that cloud-based EMR systems ensure enhanced security for EMR storage and allow legitimate users to access the required data from anywhere, thus avoiding misuse of data.

However, the existing cloud-based EMR system relies on a central server to handle EMR and process inquiry requests, which presents significant drawbacks. First, the centralized model is susceptible to a single point of failure that renders the entire system inoperable. Second, the reliability of cloud servers can be questionable. Fortunately, the concept of distribution is increasingly gaining attention from researchers. Blockchain, a decentralized architecture technology, offers a new approach to solving these problems [9].

Blockchain technology has the potential to resolve security concerns associated with EMR data due to its decentralized nature. Additionally, it can help protect individual privacy and data security when sharing data in the healthcare domain. Despite the promising potential of a blockchain-based EMR sharing system, it still faces the following challenges:

(1) How can patient privacy be protected on the blockchain while ensuring that they can verify and prove the authenticity of the shared EMR data?
(2) How does the system set access control policies based on user attributes so that only authorized users can access a patient's EMR?
(3) How can we share a patient's EMR using the Hyperledger Fabric system while ensuring data security?

To address the aforementioned challenges, we propose a secure EMR sharing scheme with privacy protection using a consortium blockchain and proxy re-encryption in this paper. Specifically,

(1) We design a scheme that combines attribute-based signature encryption (ABSE) with blockchain technology, which ensures the security of EMR data sharing. It allows for fine-grained access control and ensures that only authorized users can access and verify the authenticity of shared EMR data.
(2) We propose a framework that employs the interplanetary file system (IPFS) for distributed storage of ciphertexts of patients' EMRs. Additionally, we utilize proxy re-encryption (PRE) to effectively mitigate the risk of collusion attacks and reduce the possibility of unauthorized access.
(3) Comprehensive functionality and performance evaluation results show that the proposed scheme can achieve security requirements and outperform other existing schemes in terms of computation and communication costs.

The remainder of this paper is structured as follows. Related work is introduced in Related work section. Section 3 outlines the preliminary work relevant to the proposed scheme. Our proposed scheme is presented in Sect. 4. In Sect. 5, we conduct a security analysis and performance evaluation. Finally, Sect. 6 summarizes our work as a whole.

Related work

A Blockchain-based EMR data sharing schemes

Since the emergence of blockchain technology, many EMR data-sharing schemes [10–12] have been proposed in recent years, which utilize its promising potential for privacy protection. For instance, Azaria et al. [10] proposed the first scheme to implement a decentralized electronic medical record management system using blockchain, called MedRec. However, Akkaoui et al. [11] pointed out that MedRec does not provide access policies and relies heavily on hospital databases. Therefore, they proposed a new data management framework called "EdgeMedichain" to share medical data more securely and efficiently. Liu et al. [12] constructed a scheme to share medical data on a private blockchain. However, Wang et al. [13] pointed out that private blockchain is not effective when sharing patients' data among different healthcare organizations, so they proposed a patient-centered healthcare data-sharing system that implements querying a single keyword on the blockchain. However, the systems [13–15] are all implemented on Ethereum, whereas Rajput et al. [16] pointed out that the Ethereum system suffers from the weaknesses of inefficient transactions and higher energy consumption compared to Hyperledger Fabric. Thus, they utilized Hyperledger Fabric in their

scheme. However, Chi et al. [17] pointed out that the scheme in [16] was limited by the scalability of the blockchain. Mani et al. [18] proposed a novel approach, known as patient-centric healthcare data management (PCHDM), for storing certain data on IPFS to address the issue of data storage on the blockchain. To provide complete privacy protection and efficient ciphertext retrieval for EMR, Liu et al. [19] proposed an inner product searchable encryption scheme with multi-keyword search based on blockchain. To solve the inefficiency of the existing scheme, Lin et al. [20] proposed a pairing-free and blockchain-friendly universal designated verifier signature proof (UDVSP) scheme. It is worth noting that the scheme is the first system with anti-malicious propagation to date. Driven by the above work, we have designed a trusted data-sharing framework utilizing Hyperledger Fabric and IPFS that supports personal privacy protection and gets rid of the blockchain's scalability problem.

B Applications of proxy re-encryption

PRE is an encryption method that securely converts ciphertext without revealing any corresponding plaintext information during the conversion process. In the course of related research, many researchers have improved and innovated this algorithm [21–23]. Chu et al. [23] proposed allowing the proxy to convert the ciphertext into a set of delegates. Alice can then grant decryption privileges to the user, resulting in significant cost savings. But Shabisha et al. [24] pointed out that [23] is not suitable for some dynamic data sharing. Subsequently, they proposed a scheme using pairing-free proxy re-encryption that can store data in the cloud. Unfortunately, they just proposed the idea and did not practice it. Kan et al. [25] proposed the chosen-ciphertext attack (CAA) scheme, which allows for selected ciphertext attacks and reduces the cost of keys, thus preventing collusion attacks and ensuring distributed storage. However, Wang et al. [26] pointed out that although their scheme could prevent the leakage of confidential information, it needed to replace the key regularly, which increased the operating cost. Therefore, they proposed an improvement scheme that combined proxy re-encryption and searchable encryption to achieve a better cost reduction of keys. However, Mamta et al. [27] pointed out that storage costs had increased. Chen et al. [28] propose a new EMR system that utilizes proxy re-encryption to secure data on the consortium blockchain, thereby addressing the issue of data security. In order to protect user privacy, Qi et al. [29] presented a point-of-interest (POI) category recommendation model based on group preferences, which can capture users' dynamic preferences to better recommend the POI categories. Liu et al. [30, 31] pointed out that the data in [29] is not encrypted at the root, and there is a risk of user privacy data leakage. Based on the above work, we also use the PRE algorithm to protect the data sharing authorization and consider combining it with IPFS and Fabric, achieving a secure and lightweight EMR sharing framework.

C Blockchain-based EMR sharing schemes with access control

Access control is widely recognized as a crucial method for ensuring secure and manageable sharing of EMRs. Many researchers have explored fruitful results [32–34] in this field. Attribute-based encryption (ABE) can be broadly categorized into two types, the first is key-policy attribute-based encryption (KP-ABE) [33] for biometric systems and the second is ciphertext-policy attribute-based encryption (CP-ABE) [34] for cryptographic storage systems. So many researchers have applied ABE to various scenarios based on blockchain. Sun et al. [35] proposed an attribute-based scheme that allows for cloud server data access through keyword searching. However, Guo et al. [36] pointed out that it is not feasible to verify the accuracy and completeness of the retrieved data. Consequently, they have devised an alternate scheme that encrypts the medical data employing CP-ABE technology and assigns distinct search privileges to different users. However, Xu et al. [37] pointed out that excessive searches can occasionally fail to validate all returned results, resulting in a waste of resources. Therefore, they proposed a scheme for decreasing the number of attribute encryptions and decryptions in a cloud environment, which permits effective data access control. However, Jiang et al. [38] pointed out that cloud servers are not entirely trustworthy. Egala et al. [39] proposed an efficient blockchain access system that employs a selection ring-based approach to attain data security. Wang et al. [40] designed a decentralized framework for secure EMR sharing. The scheme uses smart contract technology to build a trusted platform for medical centers to share encrypted EMRs. Driven by the above work, we combine blockchain and access control to achieve data traceability and integrity.

To address the shortcomings of previous related work, we propose to store patients' EMRs in IPFS and use proxy re-encryption to safeguard the data. To achieve controlled access to patients' data, access logs of users with different attributes will also be uploaded to the consortium blockchain. Through comparative analysis and experimental simulations, our scheme

can solve the aforementioned challenges in EMR systems and be practical in the real world.

Preliminary

Hyperledger fabric

Hyperledger Fabric is a platform based on blockchain technology, which can protect data through channels [16]. This platform allows participants to establish a subnet, and only relevant nodes can view the transactions of a specific set. In this way, smart contracts and processed data can only be accessed by authorized members, thus protecting the privacy and confidentiality of transactions.

Private data refers to data that can be aggregated among channel members, and they can be protected like channel data. This means that even if the data is aggregated, only authorized members can access them. This method can provide the same protection as the channel without the need to maintain and build a separate channel.

Interplanetary file system (IPFS)

IPFS replaces traditional domain-based addressing with content-based addressing, eliminating the need to worry about the location of servers or the storage path and name of files. Whenever a file is uploaded to an IPFS node, a unique encrypted hash value is generated based on the file's content. The hash value reflects the file's content, so even a slight change in a single bit will result in a different hash value. When IPFS receives a request for a file hash, it uses a distributed hash table to locate the corresponding file node and retrieve and verify its content data [18]. The most important feature of IPFS is its ability to retrieve content by completely transforming the lookup process.

Elliptic curve Digital signature algorithm (ECDSA)

ECDSA is mainly used to create digital signatures to verify the authenticity of data without affecting the security of data. It should be noted that ECDSA is not used to encrypt data or provide data access protection. Its purpose is to ensure that data is not tampered with during transmission.

The digital signature is a unique identification generated by applying mathematical algorithms to data, which is used to prove the integrity and identity of data. ECDSA uses elliptic curve cryptography to generate public key and private key pairs and uses the private key to sign the data. The signed data and the related public key can be publicly displayed without disclosing the private key. The receiver can use the public key to verify the authenticity of the signature, to determine whether the data has been tampered with.

In short, ECDSA ensures the integrity of data using digital signatures and can verify the authenticity of data even during transmission. This section provides an overview of the standard process of the ECDSA algorithm for generating key pairs and private key signatures.

Symbol Definition:

The parameter of elliptic curve E is defined as $params = (p, a, b, G, n)$, where a and b are parameters of the elliptic curve equation, p is a major prime number, the operation of coordinates x and y on the elliptic curve is uniform modulus p, G is the base point of the elliptic curve (G_x, G_y), $n$is the order of G on the elliptic curve, and $[k]P$ represents the k-fold point of P on the elliptic curve.

Signature process:

(1) Select an elliptic curve E_P, and a base point G.
(2) Generate a random private key SK_A and use G to compute the public key $PK_A = SK_A G$.
(3) Generate a random integer k ($k < n$, n is the order of G) and compute the point $(x_q, y_q) = kG$.
(4) Let $Q = x_q \% n$ and compute $T = \frac{H + Q * SK_A}{k} \% n$.
(5) Get signature (Q, T), if Q is 0 then re-select the random number k to compute again.

Verification process:

(1) After receiving the message m and the signature value (Q, T).
(2) Compute$u_1 = (Hmodn)/T, u_2 = (Qmodn)/T$.
(3) Calculate the point $(x_v, y_v) = u_1 * G + u_2 * PK_A$.
(4) Verify the equation: $Q = x_v modn$.
(5) If holds, (Q, T) is accepted, otherwise (Q, T) is invalid.

Proxy Re-encryption (PRE)

In proxy re-encryption based on elliptic curves, we designate E as an elliptic curve over a finite field F_q, where q is a large prime number, and G is a point on the elliptic curve E of order n [41]. Let G_1 and G_2 be two cyclic groups of multiplication with the prime modulo n. We can describe the bilinear map $e : G_1 \times G_1 \rightarrow G_2$, such that $z = e(G_1, G_1) \in G_2$ [42]. The following properties are met:

(1) Bilinear: For any $a, b \in Z_p^*$ and $x, y \in G_1$, $e(x^a, y^b) = e(x, y)^{ab}$ holds;
(2) Non-degenerate: There exists $x, y \in G_1$ such that $e(x, y) \neq 1$;
(3) Computable: For any $x, y \in G_1$, there exists an effective algorithm to compute$e(x, y)$.

Proxy re-encryption is a secure encryption technology, which can help users to achieve more flexible operations in the process of transforming ciphertext while maintaining data confidentiality. Specifically, PRE allows user A to encrypt and upload the ciphertext using the public key, and then convert the ciphertext to another format. In this way, user B can decrypt the new ciphertext with its private key, while ensuring the confidentiality of any corresponding plaintext during the whole conversion process.

In short, PRE provides a way to encrypt and decrypt data, so that the owner of the data can operate without directly exposing the plaintext. By using PRE, users can choose to convert the ciphertext to different formats, so that other users can decrypt and obtain plaintext using their private key. This method provides higher flexibility and security because the ciphertext can be decrypted by multiple users without disclosing the plaintext content:

(1) Key generation algorithm $KeyGen(G) \rightarrow (PK_A, Sk_A, PK_B, Sk_B)$: When the system public parameter G is inputted, the algorithm produces a public-private key pair (PK, SK) for the user.
(2) Encryption algorithm $Enc(G, M, PK_A) \rightarrow C_A$: When "$G$", the plaintext message "M" in the information space and "PK_A" are entered into the algorithm, the algorithm generates the ciphertext "C_A" encrypted by "PK_A".
(3) Rekey generation algorithm $ReKeyGen(Sk_A, PK_B) \rightarrow rk_{A\rightarrow B}$: A transformation key $rk_{A\rightarrow B}$ for one-way re-encryption between user A and B is generated by the algorithm using Sk_A and PK_B.
(4) Re-encryption algorithm $ReEnc(C_A, rk_{A\rightarrow B}) \rightarrow C_B$: The operation converts C_A to C_B and sends it to user B. Then, C_B can be decrypted by user B with their private key Sk_B.
(5) Decryption algorithm $Dec(C_B, Sk_B) \rightarrow M$: When Sk_B and C_B are inputted, the algorithm produces the corresponding plaintext M.

Attribute-based signature encryption (ABSE)

ABSE technology is a method for information encryption, which allows the encrypting party to specify the access policy and express it as an access structure. This access structure describes the set of attributes required to understand the secret party. Only when the decrypting party has a set of attributes that meet the requirements of the access structure can the information be decrypted successfully.

In short, ABSE technology is an encryption method that can restrict the decryption permission according to the decryption Party's attributes. For example, suppose a file is encrypted, and decryption of the file needs to meet certain conditions, such as age over 18, position as a doctor, etc. Only those who meet these conditions can decrypt the file.

By using the ABSE technology, the encrypting party can more accurately control the access rights of information and ensure that only qualified personnel can decrypt sensitive information. This is important for protecting confidential data and privacy:

(1) *Setup*: This algorithm is executed by system and is mainly used to generate public keyPK and private key SK.
(2) *Encrypt*: This algorithm is executed by the data owner and uses access structures to encrypt plaintext, generating ciphertext.
(3) *KeyGen*: This algorithm is executed by system and generates the key SK based on the attribute set S provided by the data user.
(4) *Decrypt*: This algorithm is executed by the data user, using SK to decrypt the ciphertext and obtain the plaintext.

System model

Figure 1 shows the EMR sharing system model of our proposed scheme, which is based on Hyperledger Fabric and IPFS. There are four entities in the system, i.e., Hospital blockchain system (HB), Doctor (D), Patient (P), and IPFS.

Hospital blockchain system (HB)

Hospital blockchain system consists of multiple medical institutions, e.g., the general hospital, and specialized hospital. Its functions include distributed storage capabilities, digital identity certification, user identity management, and signature verification. It is built on Hyperledger Fabric and pre-deployed chain codes such as signature verification. Any user (e.g., a doctor, or a patient) who needs services of the consortium system must register with it first.

Doctor (D)

When the doctor requests a patient's medical records for further diagnosis and treatment, he sends the patient a request for access through HB. After the reception of the patient's authorization, the doctor can use the obtained hash address to query the patient's EMR in IPFS to diagnose the patient.

Patient (P)

The patient is the owner of the personal electronic medical record. They upload encrypted EMRs to IPFS for storage. In addition, the patient is responsible for giving the

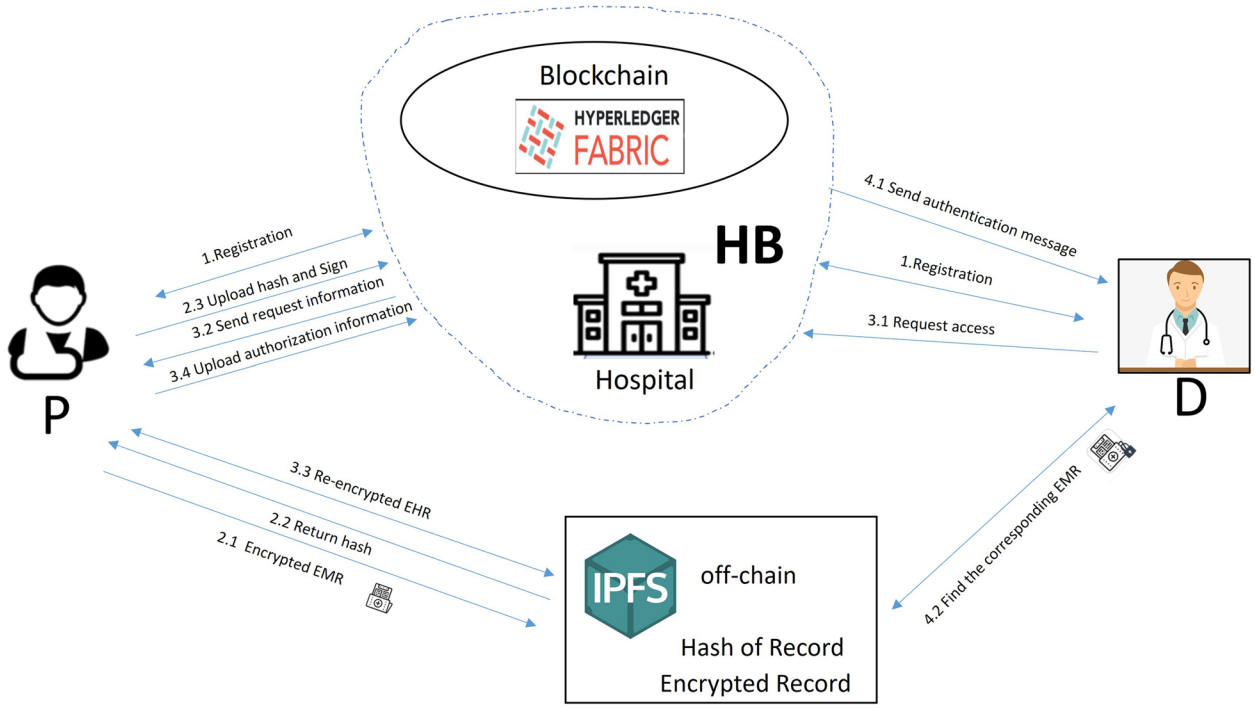

Fig. 1 System model

doctor permission to access his EMR according to configurable access policies.

Interplanetary File System (IPFS)

IPFS can store a patient's EMR and return hashes when the patient uploads an encrypted EMR. Moreover, the patient submits these hashes to the chain, a process that enables decentralized data storage. Once the doctor's access is approved, the doctor can retrieve the patient's corresponding EMR from the IPFS by getting the hash value from the patient's authorization information.

The proposed scheme

The main notations and corresponding definitions are listed in Table 1.

System overview

According to the system model in Fig. 1, the workflow of the proposed scheme is as follows.

Step 1: Doctors and patients are required to register through HB. When a registration request is received, HB creates public-private key pairs and digital certificates for every user and sends them to the corresponding recipients. It is worth noting that every certificate contains a specific set of predetermined characteristics, which includes role.

Step 2: The EMR is encrypted by the patient and then uploaded to IPFS storage. Following that, patients sign the information returned by IPFS to upload it to the blockchain.

Step 3: The doctor desires to access the patient's EMR and initiates an access request to HB. Subsequently, the HB assesses compliance with the access policy before granting the request. If the HB grants approval, the patient receives the request message from the doctor. Subsequently, the patient utilizes the doctor's public key to execute a proxy re-encryption algorithm and sends the resulting data back to the HB. The doctor receives the patient's authorization information through HB and gets the ciphertext for the patient's corresponding EMR on IPFS using the hash value. Finally, the doctor can decrypt the EMR with his private key.

Construction of the proposed scheme

Registration phase

During the registration phase, all users must register via HB. The registration phase is presented in Fig. 2. The specific steps are described as follows:

Step 1. To register, User X registers through the client and sends the registration information $Info_X$ to HB.

Table 1 Notations

Notation	Description
$Info_X$	The registration information of all participants in the system
SK_X, PK_X	public-private key pairs of X
G	A generator for the elliptic group
$Cert_X$	X's digital certificate
$role_X$	X's attribute
$f(m)$	The function $f(\cdot)$ for the elliptic curve that embeds the message m
P_m	m Embedding data into $f(\cdot)$
$f^{-1}(\bullet)$	The inverse function of $f(\cdot)$
J_{Pi}	Patient P's transaction information in the chain
$timestamp$	Current timestamp
M_{Pi}	Patient information and corresponding encrypted EMR
(C_A, C_B)	User ciphertext encrypted with encryption key
(C'_A, C'_B)	User ciphertext encrypted with re-encryption key
(Q_{Xj}, T_{Xj})	The jth signature generated by user X
(x_i, y_i)	A random point on an elliptic curve
$\bar{x_i}$	Point (x_i, y_i) convert x_i to an integer
H_{Xj}	The jth hash generated by user X
$rK_{A\to B}$	Proxy re-encryption key generated from A to B
n	the elliptic curve of order
e	Bilinear mapping

Step 2. Upon validation of user X's registration information, HB returns the key pair (SK_X, PK_X) and the user's certificate $Cert_X$ to user X, where $PK_X = SK_X G$.

Step 3. The user X saves $(Info_X, SK_X, PK_X, Cert_X)$.

EMR storage phase

During the EMR storage phase, the patient first encrypts the EMR source files and stores them in IPFS. Subsequently, the returned message is signed and uploaded to the blockchain. Figure 3 illustrates the flowchart of this phase.

First, patient P is required to encrypt his EMR and upload its ciphertext to IPFS. The specific process is as follows:

Step 1: The Patient P first constructs a function on the medical record data m_{P_i}.

$$Pm_{P_i} = f\left(m_{P_i}\right) \tag{1}$$

Then choose a random number k_1 to encrypt m_{P_i}, obtaining the ciphertext.

$$(C_A, C_B) = (k_1 PK_{P_i}, z^{k_1} G + Pm_{P_i}) \tag{2}$$

Step 2: P packages his EMR-related information.

$$M_{P_i} = (Info_{P_i}, (C_A, C_B)) \tag{3}$$

Then, P randomly selects a random number k_2 and invokes the general signature algorithm of ECDSA to generate a signature $(Q_{P_{i1}}, T_{P_{i1}})$ on M_{P_i}:

$$H_{P_{i1}} = hash\left(M_{P_i}\right) \tag{4}$$

$$Q_{P_{i1}} = x_q \% n \tag{5}$$

$$if\left(Q_{P_{i1}} ! = 0\right) \tag{6}$$

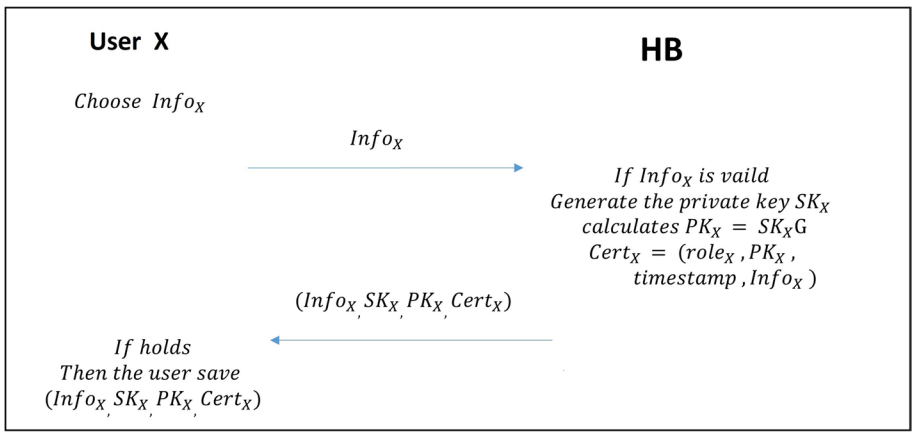

Fig. 2 Registration phase

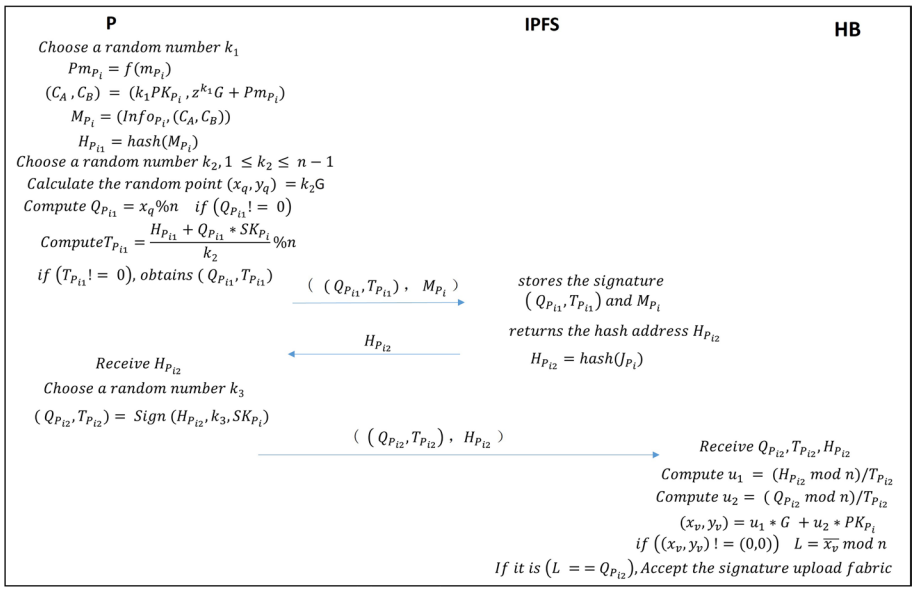

Fig. 3 EMR storage phase

$$T_{P_{i1}} = \frac{H_{P_{i1}} + Q_{P_{i1}} * SK_{P_i}}{k2} \% n \tag{7}$$

Step 3: P sends $(Q_{P_{i1}}, T_{P_{i1}})$ and M_{P_i} to IPFS for storage. Once received by IPFS, it will calculate $H_{P_{i2}}$ and return $H_{P_{i2}}$ to P.

$$H_{P_{i2}} = hash\left(J_{P_i}\right) \tag{8}$$

Step 4: Once P receives $H_{P_{i2}}$ from IPFS, P will select a random number k_3 and call the ECDSA signature algorithm to generate a signature $(Q_{P_{i2}}, T_{P_{i2}})$.

$$(Q_{P_{i2}}, T_{P_{i2}}) = Sign(H_{P_{i2}}, k_3, SK_{P_i}) \tag{9}$$

After that, P will upload the signed $H_{P_{i2}}$ to HB for storage.

Step 5: Once the HB system receives $H_{P_{i2}}$ and $(Q_{P_{i2}}, T_{P_{i2}})$ sent by the patient, the nodes participating in the consensus in the HB system will calculate the hash value of the transaction $H_{P_{i2}}$ and call the verification algorithm to verify the validity of the signature $(Q_{P_{i2}}, T_{P_{i2}})$ sent by the patient.

If the signature is valid, the consistency node puts it into the data transaction pool. After some time, the sorting node packs valid transactions into a block and submits them to the network.

Request for data access phase

To conduct further diagnosis, the doctor submits an access request to the HB to obtain permission to access the patient's EMRs. Subsequently, the HB processes the access request according to the access policy. The specific steps are shown in Fig. 4.

Step 1: Doctor D generates the request message $Req_{D_i}(Info_{D_i}, Cert_{D_i}, operation, object, timetamp)$ Then, he selects a random number k_4 and calculates $H_{D_{i1}}$ and $(Q_{D_{i1}}, T_{D_{i1}})$.

$$H_{D_{i1}} = hash\left(Req_{D_i}\right) \tag{10}$$

$$(Q_{D_{i1}}, T_{D_{i1}}) = Sign(H_{D_{i1}}, k_4, SK_{D_i}) \tag{11}$$

Then doctor D sends the request information to the patient p through the HB system.

Step 2: Once HB receives the message $H_{D_{i1}}$ and $(Q_{D_{i1}}, T_{D_{i1}})$ sent by the doctor, it will immediately verify the signature $(Q_{D_{i1}}, T_{D_{i1}})$.

The values of the "role", "object", "operation", and "time" fields are read by HB, depending on the access policy.

$$Policy\left(RoleobjectoperationTime\right) \rightarrow allow \tag{12}$$

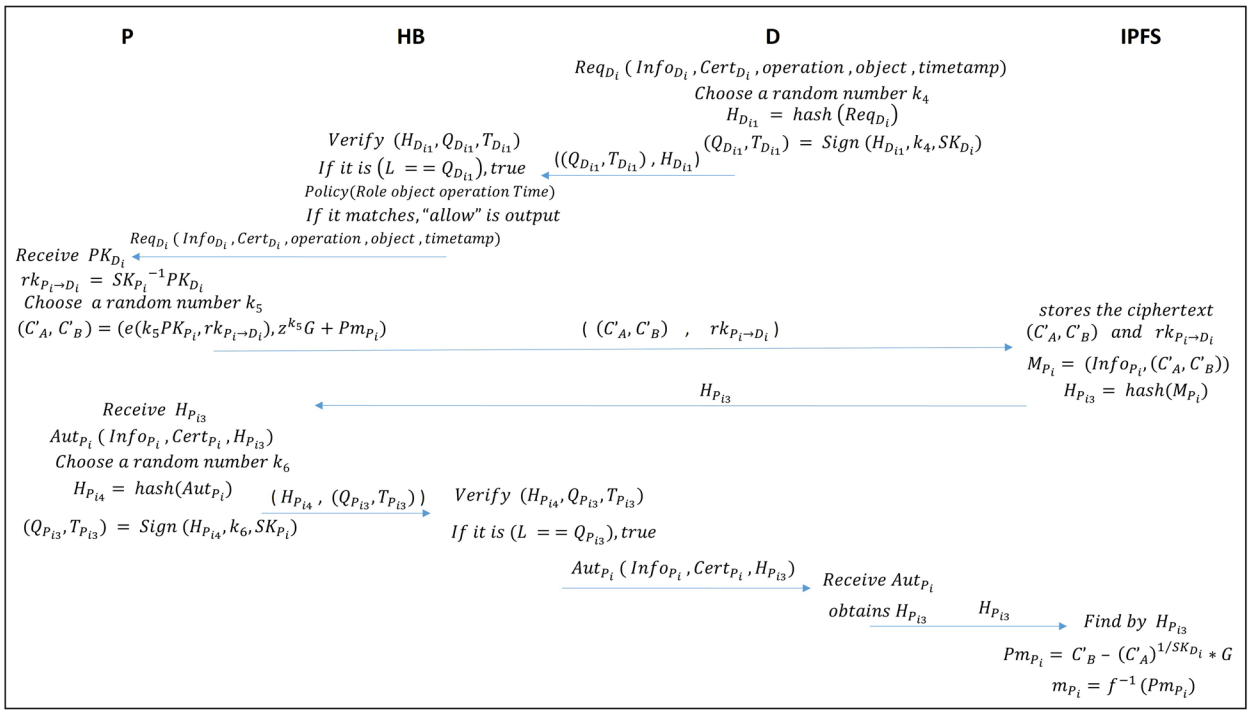

Fig. 4 Request for data access phase

If the output allows, it indicates that access is possible; otherwise, the access request is denied.

Step 3: Once patient P receives the requested information from doctor D through the HB system, patient P will be able to obtain the public key PK_{D_i} of doctor D from the information Req_{D_i}, to set the re-encryption key $rk_{P_i \to D_i}$ in combination with its private key SK_{P_i}.

$$rk_{P_i \to D_i} = SK_{P_i}{}^{-1} PK_{D_i} \tag{13}$$

Then patient P selects a random number k_5 to re-encrypt the ciphertext to get (C'_A, C'_B), and patient P stores (C'_A, C'_B) in IPFS.

$$(C'_A, C'_B) = (e(k_5 PK_{P_i}, rk_{P_i \to D_i}), z^{k_5} G + Pm_{P_i}) \tag{14}$$

Step 4: Once doctor D receives the patient P's authorization message Aut_{P_i} through the HB system, doctor D can obtain the hash address of patient P's encrypted EMR and find the corresponding EMR on IPFS. Because patient P has set the key for proxy re-encryption, doctor D can decrypt the EMR using the private key SK_{D_i}.

$$Pm_{P_i} = C'_B - \left(C'_A\right)^{1/SK_{D_i}} * G \tag{15}$$

$$m_{P_i} = f^{-1}\left(Pm_{P_i}\right) \tag{16}$$

Analysis and performance evaluation

Functional analysis

This subsection presents an informal functional analysis of the proposed scheme and compares it with previous schemes [18, 28, 39] in terms of several common features. Table 2 presents the comparison results. The symbol √ indicates that the scheme supports that function, and the symbol indicates that it does not. It can be seen from Table 2 that the proposed scheme is superior to other protocols in terms of functional features.

Data integrity

Ensuring data integrity, the proposed scheme utilizes ECDSA for signing and verifying the information. In addition, proxy re-encryption can convert encrypted data from one key to another while ensuring that user A's private key is not leaked, and authorize user B to use his own private key to decrypt the ciphertext. This can protect the privacy of both the sender and receiver, ensuring the security of the data.

Table 2 Comparison of functionality

Scheme	Data Integrity	Access control	Scalability	Traceability	Resistance to Collusion Attack
Mani et al. [18]	√	√	√	√	×
Chen et al. [28]	√	√	×	√	√
Egala et al. [39]	√	√	×	√	×
Ours	√	√	√	√	√

We take the transaction J_{Pi} to be stored in blockchain as an example. After patient P signs $H_{P_{i2}}$ using ECDSA, a signature $(Q_{P_{i2}}, T_{P_{i2}})$ is generated and sent to HB. HB can verify whether $(Q_{P_{i2}}, T_{P_{i2}})$ is legal through ECDSA's verification algorithm.

$$(Q_{P_{i2}}, T_{P_{i2}}) = Sign(H_{P_{i2}}, k_3, SK_{P_i}) \tag{17}$$

$$u_1 = (H_{P_{i2}} modn)/T_{P_{i2}} \tag{18}$$

$$u_2 = (Q_{P_{i2}} modn)/T_{P_{i2}} \tag{19}$$

$$(x_v, y_v) = u_1 * G + u_2 * PK_{P_i} \tag{20}$$

Further, tapering any data in the blockchain at this point requires extremely expensive computing power, which is impractical in the real world. Based on the above analysis, the proposed scheme can ensure the integrity of the data.

Access control

In the proposed scheme, if the doctor needs to access the patient's electronic medical records, he needs to submit an access request to HB first. Only after HB has passed the verification according to the visitor's attributes can the doctor have access to the patient's electronic medical records. Unauthorized users can't access electronic medical records. Therefore, this scheme not only realizes access control but also protects patients' privacy and data security.

Traceability

The traceability of the proposed scheme is achieved by using the blockchain's distributed ledger and encryption algorithm. Specifically, each block contains the hash value of the previous block, thus forming a tamper-proof chain. This mechanism ensures that previous transactions cannot be tampered with. Subsequent blocks rely on the information of the previous block, and tampering is detected and rejected by other nodes.

After the doctor has checked the patient's EMR, because the doctor has previously sent a request to the patient, an interactive process occurs. If the patient's condition suddenly deteriorates, due to the traceability of the system, the doctor who has previously treated the patient can be found faster through signature verification, so that the patient can receive treatment in a shorter time.

$$H_{P_{i4}} = hash(Aut_{P_i}) \tag{21}$$

$$(Q_{P_{i3}}, T_{P_{i3}}) = Sign(H_{P_{i4}}, k_6, SK_{P_i}) \tag{22}$$

$$Verify(H_{P_{i4}}, Q_{P_{i3}}, T_{P_{i3}}) \tag{23}$$

Through the above analysis, the proposed scheme can ensure the traceability of data.

Scalability

In the on-chain database, hash values of EMRs instead of operation logs are recorded in Hyperledger Fabric. The proposed scheme uploads $H_{P_{i2}}$of the patient's data address stored in IPFS to HB, and the doctor can obtain $H_{P_{i3}}$ from HB after the patient's authorization, so that the patient's EMR can be viewed. In the off-chain solution, the actual EMRs are encrypted and stored securely through IPFS, which ensures the scalability of the HB system. In the proposed scheme, the patient stores the encrypted M_{P_i} in IPFS. After re-encryption by proxy, the doctor can use his private key to obtain the patient's M_{P_i} in IPFS.

Based on the above analysis, the proposed scheme can ensure scalability.

Security analysis

In this subsection, the proposed scheme is proved to be insusceptible to some widely known attacks with an informal security analysis.

- *Resistance to Replay Attack.* In our scheme, random numbers and timestamps are used for each round of interaction. Due to the randomness of the random number and the freshness of the timestamp, the replay behavior will be accurately judged. Therefore, the proposed protocol withstands the replay attack.
- *Resistance to Man-in-the-Middle Attack.* Because of the open nature of wireless channels, adversary can intercept messages in transit. If the adversary wanted to tamper with the intercepted message, it would need random numbers and associated private keys, which is impossible to achieve. Therefore, the proposed protocol withstands the man-in-the-middle attack.
- *Resistance to Stolen Verifier Table Attack.* The proposed scheme adopts the blockchain technology of distributed architecture, no entity needs to maintain the verifier table, which avoids the risk of the verification table being stolen. Therefore, the proposed protocol withstands the stolen verifier table attack.
- *Resistance to Collusion Attack.* The proposed scheme computes the re-encryption key $rk_{P_i \to D_i}$ by utilizing SK_{P_i} of patient and PK_{D_i} of doctor. Furthermore, patient's key is well protected by the PRE algorithm. Therefore, our proposed scheme is well protected against collusion attacks.

Computation cost

In this subsection, we evaluate the performance of the proposed scheme by comparing the computation cost. To facilitate the comparison of the computation costs between the proposed scheme and other related solutions, we first define the execution time of various cryptographic operations involved in the scheme. Let T_{eo}, T_{so}, T_{ho},T_{vo}, T_{do}, T_{rk},T_{reo} respectively represent the time to execute an encryption operation, signature operation, hash operation, verification operation, decryption operation, re-encryption key operation, and re-encryption operation. Table 3 shows the comparison results of computation costs between the proposed scheme and related schemes in terms of data storage and data access phases.

It can be seen from Table 3 that the computational overhead of the proposed scheme in the data storage phase and data access phase is lower than that of several other related schemes. This is because, in the traditional way of data storage and access, data usually needs to be stored locally or on the server, and read and accessed according to the demand. This method has some problems. For example, when the amount of data is large, the capacity of local storage or server may be insufficient. At the same time, in the process of data interaction, a large number of computing operations are required, which increases the computer overhead of the user. This paper uses IPFS for data storage and sharing. In the process of interaction, some computing operations are transferred to IPFS, which reduces the computer overhead of the user. In contrast, IPFS adopts a distributed storage method, which stores data blocks on each node and uses a hash pointer for data access, which can effectively solve the problems of capacity and access speed in traditional storage methods. In IPFS, the storage and access operations of data are carried out between nodes, not on the client side, so it can reduce the computational burden on the client side.

To sum up, the scheme proposed in this paper uses IPFS distributed storage and sharing technology to reduce the computational overhead of the user in the process of data storage and access, to improve the performance and efficiency of the system.

Communication cost

In evaluating the performance of a scheme, communication overhead is also another important factor. In this section, we will compare the communication overhead of the proposed scheme with other existing schemes. We assume that the sizes of ECDSA signatures, private/public keys, hash values, transactions, and requests are 256 bits, 256 bits, 160 bits, 1024 bits, and 1024 bits respectively, while other information is 80 bits. The comparison results of communication costs are shown in Table 4.

By utilizing the IPFS mechanism for storage and access, the proposed scheme effectively reduces communication overhead. Let's take the communication costs of the proposed scheme in the data storage phase and data access

Table 3 Computation cost

Scheme Phase	Mani et al. [18]	Chen et al. [28]	Egala et al. [39]	Ours
Data storage	$2T_{eo}+2T_{so}+$ $3T_{ho}+2T_{vo}$	$2T_{eo}+4T_{so}+$ $5T_{vo}$	$T_{eo}+2T_{so}+$ $2T_{ho}+2T_{vo}$	$T_{eo}+2T_{so}+3T_{ho}+T_{vo}$
Data access	$4T_{so}+2T_{ho}+$ $3T_{vo}+T_{do}$	$5T_{so}+6T_{vo}+$ $T_{rk}+T_{ho}+$ $2T_{do}+2T_{reo}$	$3T_{so}+4T_{ho}+$ $3T_{vo}+2T_{do}$	$T_{rk}+T_{reo}+2T_{so}+2T_{ho}+T_{vo}+T_{do}$

Table 4 Communication cost

Item Scheme	Data storage phase Message Length	Data access phase Message Length
Mani et al. [18]	3056 bits	4640 bits
Chen et al. [28]	3216 bits	6096 bits
Egala et al. [39]	2720 bits	3888 bits
Ours	2880 bits	3296 bits

phase as examples. Firstly, in the data storage phase, the patient needs to send encrypted medical record information to IPFS for storage, including encryption, signature, and request verification, with a size of 80 bits + 256 bits + 1024 bits = 1360 bits. Then IPFS returns the hash value to the patient and uploads it to HB for storage, including hash value, signature, verification, and other information, with a size of 160 bits + 256 bits + 1024 bits + 80 bits = 1520 bits. The message length in the data storage phase is 1360 bits + 1520 bits = 2880 bits.

Next, in the data access phase, the doctor needs to send a request to the patient, including signature, hash value, request message, and other messages, with a total size of 256 bits + 160 bits + 1024 bits + 80 bits = 1520 bits. The patient sends signature information, transaction information, key information, hash value, and other information, totaling 256 bits + 160 bits + 1024 bits + 80 bits = 1520 bits. Then the doctor accesses IPFS to retrieve the EMR information using the hash address and decrypts it into 256 bits using the private key. The message length in the data access phase is 1520 bits + 1520 bits + 256 bits = 3296 bits.

Conclusions

To ensure the secure storage and sharing of EMRs, a secure and efficient sharing scheme based on blockchain and proxy re-encryption was proposed. Our scheme combines IPFS and proxy re-encryption. In addition, the scheme uses attribute-based personalized access control on the blockchain to enhance security. Security analysis and performance evaluation show that the proposed scheme can satisfy security requirements and outperforms the existing schemes in terms of computation and communication overhead. In future work, we will develop a prototype system to apply this scheme to real smart medical scenarios.

Acknowledgements
The authors are very grateful to the anonymous referees for their detailed comments and suggestions regarding this paper.

Authors' contributions
In this work, Xaibo Xie conceived and designed the system model and concrete algorithms; the idea is proposed by Guijiang Liu and Wenming Wang, and they critically reviewed the paper and contributed to the improvement on paper writing; Haiping Huang critically reviewed the method used and contributed to structuring the paper; the experiments are performed by Xaibo Xie. The author(s) read and approved the final manuscript.

Funding
This work is supported by Anhui Provincial Natural Science Foundation under Grant 2308085MF223; in part by the Program for Excellent Young Talents in University of Anhui Province under Grant gxyq2021192; in part by the Open Fund of State Key Laboratory for Novel Software Technology under Grant KFKT2022B33; in part by the Foundation of Yunnan Key Laboratory of Service Computing under Grant YNSC23106; and in part by the Key Project on Anhui Provincial Natural Science Study by Colleges and Universities under Grant KJ2020A0513 and Grant KJ2020A0514.

Availability of data and materials
No datasets were generated or analysed during the current study.

Declarations

Competing interests
The authors declare no competing interests.

Received: 20 December 2023 Accepted: 6 February 2024
Published online: 15 February 2024

References

1. Zhu H, Hou M (2018) Research on an electronic medical record system based on the internet, in *Proc. ICDSBA*, Changsha, China, pp. 537–540
2. Peng G, Zhang A, Lin X (2023) Patient-centric fine-grained access control for electronic medical record sharing with security via dual-blockchain. IEEE Trans Netw Sci Eng 10(6):3908–3921
3. Wu G, Wang S, Ning Z, Records JL (2022) Blockchain-enabled privacy-preserving access control for data publishing and sharing in the internet of medical things. IEEE Internet Things J 9(11):8091–8104
4. Rezaee K et al (2023) IoMT-assisted medical vehicle routing based on UAV-Borne human crowd sensing and deep learning in smart cities. IEEE Internet Things J 10(21):18529–18536
5. Li C, Dong M, Li J, Xu G, Chen X, Ota K (2021) Healthchain: secure EMRs management and trading in distributed healthcare service system. IEEE Internet Things J 8(9):7192–7202
6. Wu G, Wang S, Ning Z, Zhu B (2022) Privacy-preserved electronic medical record exchanging and sharing: a blockchain-based smart healthcare system. IEEE J Biomedical Health Inf 26(5):1917–1927
7. Hu W, Chai Y, Chen X, Zheng C (2022) Lattice based ring signature scheme for secure cloud-based EMR sharing, in *Proc. ICCCS*, Wuhan, China, pp. 789–794
8. Ge X, Yu J, Hao R, Lv H (2022) Verifiable keyword search supporting sensitive information hiding for the cloud-based healthcare sharing system. IEEE Trans Industr Inf 18(8):5573–5583
9. Zhang L, Zhang T, Wu Q, Mu Y, Rezaeibagha F (2022) Secure decentralized attribute-based sharing of personal health records with blockchain. IEEE Internet Things J 9(14):12482–12496
10. Azaria A, Ekblaw A, Vieira T, Lippman A (2016) MedRec: Using blockchain for medical data access and permission management, in *Proc. OBD*, Vienna, Austria, pp. 25–30
11. Akkaoui R, Hei X, Cheng W (2020) EdgeMediChain: a hybrid edge blockchain-based framework for health data exchange. IEEE Access 8:113467–113486
12. Liu X, Wang Z, Jin C, Li F, Li G (2019) A blockchain-based medical data sharing and protection scheme. IEEE Access 7:118943–118953
13. Wang S, Zhang D, Zhang Y (2019) Blockchain-based personal health records sharing scheme with data integrity verifiable. IEEE Access 7:102887–102901
14. Nguyen DC, Pathirana PN, Ding M, Seneviratne A (2019) Blockchain for secure EHRs sharing of mobile cloud based E-health systems. IEEE Access 7:66792–66806
15. Al Omar A, Bhuiyan MZA, Basu A, Kiyomoto S, Rahman MS (2019) Privacy-friendly platform for healthcare data in cloud based on blockchain environment. Future Generation Comput Syst 95:511–521
16. Rajput AR, Li Q, Taleby Ahvanooey M, Masood I (2019) EACMS: Emergency access control management system for personal health record based on blockchain. IEEE Access 7:84304–84317

17. Chi J, Li Y, Huang J, Liu J, Jin Y, Chen C, Qiu T (2020) 'A secure and efficient data sharing scheme based on blockchain in industrial internet of things.' J Netw Comput Appl 167:102710–102720
18. Mani V, Manickam P, Alotaibi Y, Alghamdi S, Khalaf OI (2021) 'Hyperledger healthchain: Patient-centric IPFS-based storage of health records.' Electronics 10(23):3003
19. Liu J, Fan Y, Sun R, Liu L, Wu C, Mumtaz S (2023) Blockchain-aided privacy-preserving medical data sharing scheme for E-healthcare system. IEEE Internet Things J 10(24):21377–21388
20. Lin C, Huang X, He D (2023) Efficient blockchain-based electronic medical record sharing with anti-malicious propagation. IEEE Trans Serv Comput 16(5):3294–3304
21. Blaze M, Bleumer G, Strauss M (1998) Divertible protocols and atomic proxy cryptography, in *Proc. EUROCRYPT*, Espoo, Finland, pp. 127–144
22. Sun M, Ge C, Fang L, Wang J (2018) A proxy broadcast re-encryption for cloud data sharing. Multimedia Tools Appl 77(9):10455–10469
23. Chu CK, Weng J, Chow SSM, Zhou J, Deng RH (2009) Conditional proxy broadcast re-encryption, in *Proc. ACISP*, Brisbane, QLD, Australia, pp. 327–342
24. Shabisha P, Braeken A, Touhafi A, Steenhaut K (2017) Elliptic curve qu-vanstone based signcryption schemes with proxy re-encryption for secure cloud data storage. Proc. CloudTech, Rabat, pp 1–18
25. Kan J, Zhang J, Liu D, Huang X (2022) Proxy re-encryption scheme for decentralized storage networks. Appl Sci 12(9):1–20
26. Wang Y, Zhang A, Zhang P, Wang H (2019) Cloud-assisted EHR sharing with security and privacy preservation via consortium blockchain. IEEE Access 7:136704–136719
27. Gupta BB, Li KC, Leung VC, Psannis KE, Yamaguchi S (2021) Blockchain-assisted secure fine-grained searchable encryption for a cloud-based healthcare cyber-physical system. IEEE/CAA J Automatica Sinica 8(12):1877–1890
28. Chen W, Zhu S, Li J, Wu J, Chen C-L, Deng Y-Y (2021) Authorized shared electronic medical record system with proxy re-encryption and blockchain technology. Sensors 21(22):7765
29. Qi L, Liu Y, Zhang Y, Xu X, Bilal M, Song H (2022) Privacy-aware point-of-interest category recommendation in internet of things. IEEE Internet Things J 9(21):21398–21408
30. Liu Y, Zhou X, Kou H, Zhao Y, Xu X, Zhang X et al (2023) Privacy-preserving point-of-interest recommendation based on simplified graph convolutional network for geological traveling. ACM Trans Intell Syst Technol
31. Liu Y et al (2023) Interaction-enhanced and time-aware graph convolutional network for successive point-of-interest recommendation in traveling enterprises. IEEE Trans Industr Inf 19(1):635–643
32. Sahai A, Waters B (2005) Fuzzy identity-based encryption, in *Proc. EUROCRYPT*, Aarhus, Denmark, pp. 457–473
33. Qi F, Li Y, Tang Z (2018) 'Revocable and traceable key-policy attribute-based encryption scheme.' J Commun 39(11):63–69
34. Bethencourt J, Sahai A, Waters B (2007) Ciphertext-policy attribute-based encryption, in *Proc. IEEE S&P*, Berkeley, CA, USA, pp. 321–334
35. Sun W, Yu S, Lou W, Hou YT, Li H (2014) Protecting your right: Attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud, in *Proc. INFOCOM*, Toronto, ON, Canada pp. 226–234
36. Guo C, Zhuang R, Jie Y, Ren Y, Wu T, Choo K-K-R (2016) Fine-grained database field search using attribute-based encryption for E-healthcare clouds. J Med Syst 40(11):235
37. Xu L, Xu C, Liu JK, Zuo C, Zhang P (2020) Building a dynamic searchable encrypted medical database for multi-client. 527:394–405Information Sciences
38. Jiang P, Guo F, Liang K, Lai J, Wen Q (2020) Searchain: Blockchain-based private keyword search in decentralized storage. Future Generation Comput Syst 107:781–792
39. Egala BS, Pradhan AK, Badarla V, Mohanty SP (2021) Fortified-chain: a blockchain-based framework for security and privacy-assured internet of medical things with effective access control. IEEE Internet Things J 8(14):11717–11731
40. Wang M, Guo Y, Zhang C, Wang C, Huang H, Jia X (2023) MedShare: a privacy-preserving medical data sharing system by using blockchain. IEEE Trans Serv Comput 16(1):438–451
41. Thangam V, Chandrasekaran K (2016) 'Elliptic curve based proxy re-encryption.' Proc. ICTCS. Udaipur, India, pp 1–6
42. Zhang F, Safavinaini R, Susilo W (2004) An efficient signature scheme from bilinear pairings and its applications, in *Proc. PKC*, Singapore, Mar. pp. 277–290

Publisher's Note