Hindawi
IET Information Security
Volume 2024, Article ID 5559522, 16 pages
https://doi.org/10.1049/2024/5559522

Research Article

A Blockchain-Based Trustworthy Access Control Scheme for Medical Data Sharing

Canling Wang,[1] Wei Wu,[1] Fulong Chen,[1] Hong Shu,[2] Ji Zhang,[3] Yuxuan Zhang,[1] Taochun Wang,[1] Dong Xie,[1] and Chuanxin Zhao[1]

[1]*Anhui Provincial Key Laboratory of Network and Information Security, Anhui Normal University, Wuhu, China*
[2]*Tongling University, Tongling, China*
[3]*University of Southern Queensland, Toowoomba, Australia*

Correspondence should be addressed to Fulong Chen; long005@ahnu.edu.cn

Received 3 October 2023; Revised 6 January 2024; Accepted 13 January 2024; Published 31 January 2024

Academic Editor: Taimur Bakhshi

Blockchain is commonly employed in access control to provide safe medical data exchange because of the characteristics of decentralization, nontamperability, and traceability. Patients share personal health data by granting access rights to users or medical institutions. The major purpose of the existing access control techniques is to identify users who are permitted to access medical data. They hardly ever recognize internal assailants from legitimate entities. Medical data will involve multilayer access within the authorized organizations. Considering the cost of permissions management and the problem of insider malicious node attacks, users hope to implement authorization constraints within the authorized institutions. It can prevent their data from being maliciously disclosed by end-users from different authorized healthcare domains. For the purpose to achieve the fine-grained permissions propagation control of medical data in sharing institutions, a trust-based authorization access control mechanism is suggested in this study. Trust thresholds are assigned to different privileges based on their sensitivity and used to generate zero-knowledge proof to be broadcasted among blockchain nodes. This method evaluates the trust of each user through the dynamic trust calculation model. And meanwhile, smart contract is employed to verify whether the user's trust can activate some permissions and ensure the privacy of the user's trust in the process of authorization verification. In addition, the authorization transaction between users and institutions is recorded on the blockchain for patient traceability and accountability. The feasibility and effectiveness of the scheme are demonstrated through comprehensive comparisons and extensive experiments.

1. Introduction

In the era of big data, multisource heterogeneous, fast-growing, accurate, and massive healthcare data are widely used in various fields, such as disease research, clinical treatment, new drug development, epidemic prevention, and control. It can substantially improve the efficiency and accuracy of medical research and significantly reduce the burden of social medical costs [1]. However, due to the extensive collection and application of medical data, it also faces a series of security and privacy threats such as theft of personal data, phishing attacks, illegal user access, and ransomware attacks. All of these may lead to the leakage and loss of medical data, thus failing to ensure the privacy, integrity, and reliability of the data.

Patient-centered healthcare data exchange attempts to shift data ownership from the provider to the patient [2]. When a patient needs to treat across medical institutions or provide data sources for disease research centers [3], they can authorize organizations to share their medical data. As an emerging Internet database technology, blockchain [4] has the characteristics of decentralization, transparency, and data tamperability. It provides an innovative method for storing information, executing transactions, and building trust in an open environment [5]. It has become popular to employ blockchain technology to securely share patient data with different institutions. The existing methods [6–10] mainly focus on deploying access control policies on the blockchain to access medical data by identifying users authorized by patients. However, these schemes only

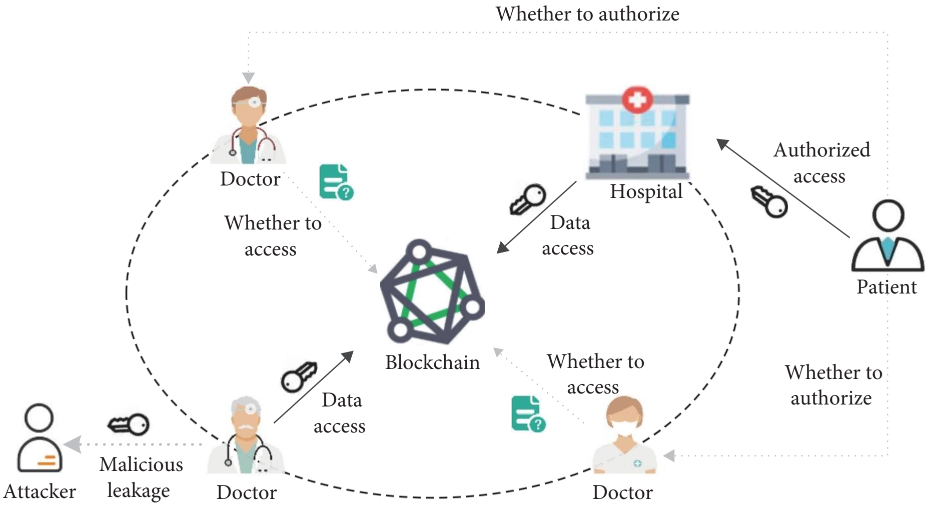


Figure 1: Internal access security model.

consider the single authorized access between patients and users and do not take into account the multilayer access of data between institutions. As shown in Figure 1, there is still a risk of internal leakage of data in the organizations that have been authorized by patients. Furthermore, it would be a huge cost if access control operations were carried out for each internal personnel. Considering the cost of authorization management and malicious attacks on internal nodes, patients want to restrict the secondary authorization of personal health data within the organization. Therefore, in the face of these challenging issues, we need to quickly find solutions to securely access patient data within authorized institutions in order to maximize patient privacy.

The lack of fine-grained security access control models for authorized institutions (*Ai*) is vulnerable to attacks by malicious nodes with legitimate identities and privileges within the institution [11]. Traditional access control models, such as role-based access control (RBAC) [12] and attribute-based encryption (ABE) [13], have been proven to be effective in detecting and preventing illegal access to data. However, the above methods cannot provide practical solutions to the security problems such as unknown user access, internal node attacks, and unaccountable data leakage that exist in *Ai*. To alleviate these troubles, trust-based access control (TBAC) [14] was then introduced to the authorization constraints within *Ai*. Patients delegate *Ai* to set trust thresholds for different permissions to identify sensitivity. For the staff from the delegated institution, only with a high trust level is granted authority to obtain the patient's medical information.

The trust of internal user (*Iu*) can be verified by designing smart contract on the blockchain [15]. However, there is still a challenge for *Iu* to prove to *Ai* that its trust can activate data access without divulging privacy. As a novel and effective variant of zero-knowledge proofs, zero-knowledge-succinct non-interactive arguments of knowledge (zk-SNARKs) [16, 17] can achieve verification without disclosing the content of the proof. Especially, zk-SNARKs are combined with smart contracts for authentication to ensure that the *Iu* trust matches the authority trust assigned by *Ai* and protect user privacy.

To overcome the abovementioned problems, a trust-based authorization access control (TAAC) scheme is proposed to solve the secure access of medical data within *Ai* though using blockchain technology and deploying smart contracts. Overall, our main contributions are summarized as follows.

(1) A trust-based authorization and verification scheme for medical data is designed, named TAAC. This approach can settle the difficulties of costly authorization constraitns and internal malicious node attacks caused by patient entities intervening in institutional entities. In our TAAC, *Ai* can send preassigned permission trust threshold ciphertexts to the blockchain through transactions for distributed access control.

(2) In the proposed scheme, a dynamic trust calculation model (TCM) is constructed to evaluate the trust of end users. Then, utilizing zk-SNARKs, the user creates a reliable zero-knowledge proof and sends it to the smart contract for validation, which can ensure that the user does not reveal any privacy. The smart contract determines whether it can activate the permission by comparing the user's trust with the permission trust threshold set by the organization. The permission transactions between *Iu* and *Ai* will be broadcast in the blockchain for patient accountability and traceability after the verification is successful [18].

(3) Through the comprehensive comparisons with the existing schemes, we find that the proposed scheme exhibits higher privacy and execution efficiency in terms of the metrics of data protection and resistance to attacks. The validation is carried out in terms of both theoretical security analysis and experimental performance evaluation, proving the superiority of TAAC.

The remainder of the paper is structured as follows. Related work is described in Section 2, and some early components are offered in Section 3. The proposed scheme is outlined in Section 4, and the specific implementation details of TAAC are introduced in Section 5. Section 6 summarizes the characteristics of the TAAC scheme and compares it with other schemes in terms of performance. Finally, Section 7 provides a summary of the work done in the paper.

2. Related Works

Conducting in-depth research on access control for medical data is an important tool for user privacy protection. In this section, the related work on trust-based access control and blockchain-based access control is presented.

2.1. Trust-Based Access Control Schemes. An access control model to evaluate the trust of interactive agents can improve the privacy and robustness of the medical system. Healthy-Broker [19] was a trust-building agent architecture specifically designed for eHealth services. It securely completes eHealth transactions by evaluating trust relationships and uses a distributed blockchain ledger for tracking to prevent potentially malicious behavior. Sahaana et al. [20] provided a trust assessment model for dynamically managing nodes, which uses trust to evaluate the behavior of each node, thus effectively avoiding the improper behavior of malicious nodes. Lewandowski et al. [21] improved the treatment effect and smoothed the overall function of the medical system by estimating patient trust. Jiang et al. [22] proposed a T-RBAC model utilizing role-attribute trust and physician-behavior trust for hierarchical authorization. Singh and Chatterjee [23] proposed an access control rule set, which can protect unauthorized access to medical data and dynamically control access views. Hu et al. [24] combined the entropy weight method with fuzzy theory to comprehensively evaluate the interactive trust value, and then used a two-way selection mechanism of roles and a third-party real-time monitoring mechanism to dynamically control access to the healthcare cloud system. The VARTE model [25] is based on vector auto-regressive (VAR) to effectively compute the trust value of user behavior in mobile health information systems. He et al. [26] discussed a distributed medical sensor network trust evaluation model, in which nodes can assess the credibility of each other to detect malicious nodes. Athanasiou et al. [27] introduced adaptive cloud inference system to fuzzy infer doctor credibility, which helps to ensure patient satisfaction in the universal medical environment. Xin et al. [28] used hierarchical analysis to determine trust evaluation index weights in medical big data and combined with whitening weight function to solve the problem of inaccurate trust evaluation results.

2.2. Blockchain-Based Access Control Schemes. Network attacks are prevalent in healthcare systems [29], and the application of blockchain technology to access control in medical data is also a hot topic of current studies. Siyal et al. [30] believed that the public verifiability of blockchain provides access control to electronic medical records without any third party but can not guarantee the reliability of data sources. Fan et al. [31] designed a blockchain-based electronic medical record framework named MedBlock for secure medical data sharing. Xia et al. [32] used smart contracts and access control mechanisms for data tracking to achieve auditable and secure sharing of medical data by revoking the access rights of illegal entities. However, it is impossible to avoid internal attacks by users with legal identities. Hussien et al. [33] proposed an access control scheme for outsourced encrypted medical data, which bridged the gap between personal health records and blockchain technology. Gan et al. [7] designed an incentive mechanism to encourage patients to share data and let patients act as supervisors to supervise unauthorized medical institutions to legally use their own medical data. Obviously, the data may have been maliciously compromised. Feng et al. [34] combined hierarchical attribute encryption with linear secret sharing, which avoided the security risk of submitting access policies to the blockchain network, and enabled authorized users to efficiently query the required data. Thwin and Vasupongayya [35] established a fine-grained access control model, which uses proxy re-encryption technology to protect medical data privacy and support access revocation. Sun et al. [36] stored the hash of medical data on the blockchain, while the specific data are stored in the IPFS. And only users who satisfy the attributes can decrypt the data. Saini et al. [37] proposed four types of smart contracts for user authentication and access control, and combined elliptic curve cryptography and Edwards curve digital signature algorithm technology to protect data privacy. However, these schemes mainly focus on single authorized access between patients and data users, while the risk of data leakage still exists in the institutions that have been authorized by patients.

Table 1: Notation setting.

Notations	Descriptions
H_0, H_1	Secure hash functions
P	Permission
$\mathrm{PL} = \{P_1, P_2, \ldots, P_n\}$	Permission list
Pt_i	Permission trust threshold
$T = \{\mathrm{Pt}_1, \mathrm{Pt}_2, \ldots, \mathrm{Pt}_n\}$	Permission trust threshold set
λ	Security parameter
h_i	Authorized hospital
d_i	Doctor
MPK, MSK	Master key pair
$\mathrm{PK}_h, \mathrm{SK}_h$	Key pair of a hospital
$\mathrm{PK}_d, \mathrm{SK}_d$	Key pair of a doctor
GK	Key for generating a proof
VK	Key for verifying a proof
CT	Ciphertext
σ	Digital signature
π	Zero-knowledge proof

3. Preliminaries

Theoretical knowledge and related techniques involving TAAC will be brought up in this section. For the sake of clear and concise presentation, the commonly used symbols are listed in Table 1.

3.1. Blockchain. Blockchain is considered to be a distributed storage database that combines technologies such as cryptographic principles, consensus mechanisms, and smart contracts. These technologies can ensure that the information in the blockchain network is traceable, nontamperable, and timely verifiable [38, 39].

3.1.1. Data Structure. Each block is composed of a block header and a block body. The block header contains three sets of metadata such as the previous block hash, Merkle tree root, and timestamp, which are used to ensure traceability and invariability. Transaction data are stored in the block body. Generally, cryptographic hashes and digital signatures are used to ensure the integrity and authenticity of transactions.

3.1.2. Consensus Protocol. It can effectively ensure that each node in the blockchain maintains the ledger according to the established rules, thus maintaining the consistency of transactions in the distributed network [40].

3.1.3. Smart Contract. The concept was first introduced by cryptographer Nick Szabo and applied to Ethereum by Vitalik Buterin. It is a computer protocol for trusted transactions without third-party supervision, which could additionally be self-executing and self-verifying [41].

3.2. Zero-Knowledge Proof. A zero-knowledge proof is one in which the prover is successful in persuading the verifier that a statement is true while withholding all relevant information from the verifier. It is also essentially an agreement involving two or more parties. Zero-knowledge proofs include both interactive and noninteractive types. Interactive proofs require multiple communications between the prover and verifier resulting in lower efficiency, while noninteractive proofs only require the prover to send a message to the verifier once according to the protocol, which makes them more efficient for blockchain applications [42].

3.3. zk-SNARKs. Zk-SNARKs are a special form of zero-knowledge proofs. A complete zk-SNARK scheme should consist of a key generation algorithm ZKKeyGen, a proof generation algorithm ZKProveGen and a verification algorithm ZKVerify. Specific definitions and explanations of these three algorithms are in Section 5. In addition to this, zk-SNARKs are distinguished from ordinary zero-knowledge proofs by the following properties.

(1) Noninteractive: One message from a prover can demonstrate to a verifier that they are aware of a certain piece of knowledge.

(2) Succinct: The time it takes to verify the proofs is minimal [40].

(3) No matter how sophisticated the program is that needs to be proved, zk-SNARKs always produce the same amount of information [41].

3.4. Bilinear Mapping. Suppose that G_1 and G_2 are two multiplicative cyclic groups with q as their common prime order. The following characteristics are met by the bilinear mapping:

(1) Bilinearity: $\forall U, V \in G_1$ and $\forall a, b \in Z_q$, the formula $e(U^a, V^b) = e(U, V)^{ab}$ is valid.

(2) Nondegeneracy: $\exists U, V \in G_1$ makes $e(U, V) \neq 1$.

(3) Computability: $\forall U, V \in G_1$, there exists an efficient algorithm to calculate $e(U, V)$.

4. System Overview

This section initially describes the TAAC system concept before going over three roughly similar scenarios. Finally, we describe the overview process of the scheme.

4.1. System Model of TAAC. The proposed TAAC model is shown in Figure 2 and mainly involves five entities: register authority (RA), hospital, doctor, TCM. and blockchain. Their respective functions can be described as follows:

(1) RA: In its capacity as a completely trusted entity, RA is in charge of configuring public parameters and keys required by the system. It is also responsible for assigning key pairs and individual identification identifiers to hospitals and doctors.

(2) Hospital: In this paper, hospitals represent typical *Ai* instance and are also incompletely trusted medical data consumers. Hospitals are responsible for making access policies, which are manifested by setting a trust threshold for each permission to identify the sensitivity of the permission. The trust set is then encrypted and the ciphertext address is sent to the blockchain via a stored transaction Tx_{storage}.

(3) Doctor: Doctors are representative roles of *Iu* in the hospital who need to access the patient medical data. If a doctor wants to access medical data, he needs to first conduct a trust assessment and generate a zero-knowledge proof π based on the trust. Then submit the π to the smart contract for authorization verification.

(4) TCM: It was designed to prevent malicious doctors from gaining permissions by forging trust. TCM calculates the doctor's trust by considering various quantitative factors, which are usually measured by those who have had direct or indirect interaction with the evaluated person.

(5) Blockchain: Transactions recorded in the blockchain will be preserved as evidence for patients to pursue accountability due to its tamper-resistance and traceable. The smart contract was deployed in advance, and automatically judges the validity of π without the participation of a third party.

4.2. The Overview Process of TAAC. Generally speaking, the intended TAAC's overview process maybe simply separated into the six steps, as shown in Figure 3, such as system initialization, broadcast in blockchain, trust evaluation, zero-knowledge proof generation, smart contract verification, and authorization recorded in blockchain.

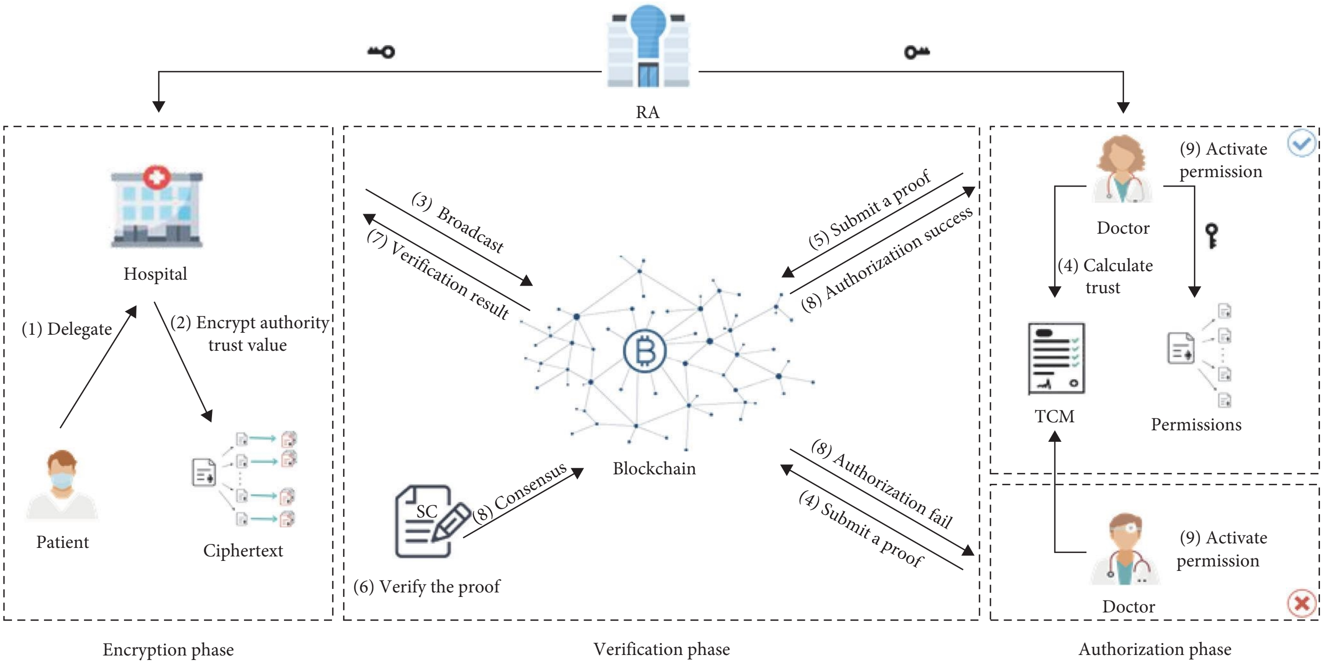


Figure 2: System model of TAAC.

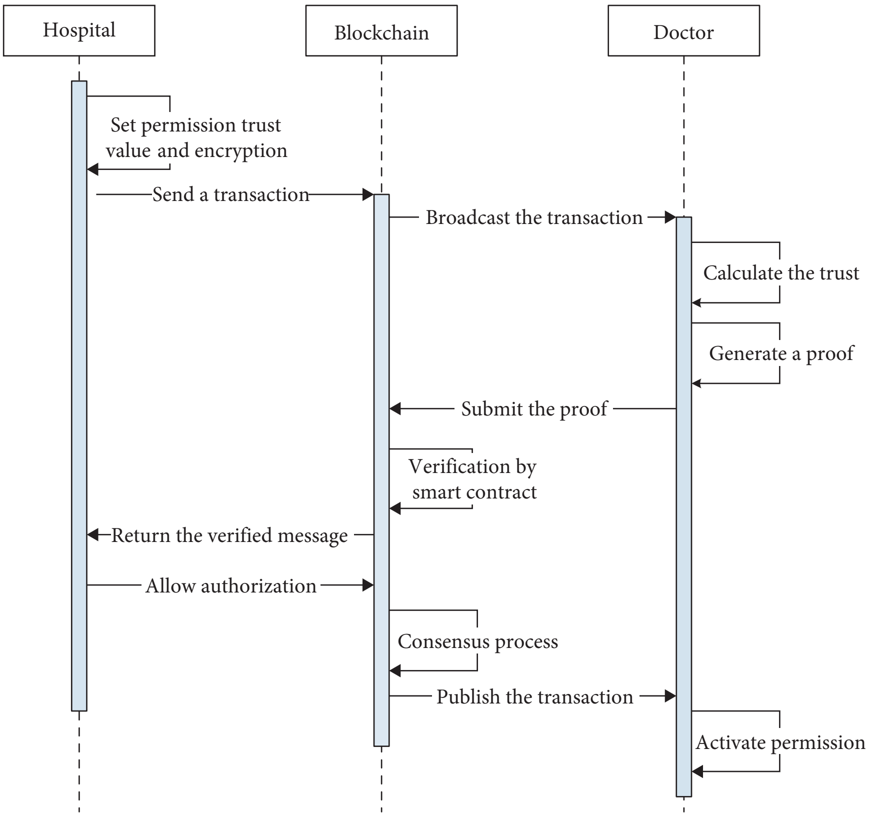


Figure 3: The overview process of TAAC.

(1) System initialization: This phase initializes the parameters in the system, such as generating key pairs and unique identifiers.

(2) Broadcast in blockchain: The patient delegates the authorized hospital to perform secondary access control. The hospital h_i is responsible for assigning the trust value Pt_i to each permission to identify the permission sensitivity, and then executes Enc algorithm to encrypt the set of trust values T by PK_h for locally secure storage. And meanwhile, the hospital h_i generates a zero-knowledge proof π' based on T and sends a storage transaction to the blockchain, and then broadcasts it in the blockchain network.

(3) Trust evaluation: The doctor d_i can provide the TCM with a unique identity identifier ID_d, and then the TCM executes the Eva algorithm to calculate the trust value of the doctor d_i.

(4) Zero-knowledge proof generation: Based on their mutual trust, the hospital h_i and the doctor d_i can each generate a reliable zero-knowledge proof π using zk-SNARKs and record the corresponding computation result R and hash value H. These are combined into a zero-knowledge proof information set *Proof* for protecting the privacy of patient medical data transactions.

(5) Smart contract verification: The doctor d_i transmits the zero-knowledge proof information set *Proof* to the blockchain. Then, the smart contract successively compares the zero-knowledge proof information set *Proof* with the corresponding information previously submitted by the hospital h_i through ZKVerify algorithm to obtain a verification result. Permission can only be activated if the personal trust exceeds the permission trust threshold.

(6) Authorization recorded in blockchain: The smart contract will inform the hospital h_i to encrypt and transfer the permission list (PL) to the doctor d_i as soon as the verification is successful. Finally, a transaction $Tx_{\mathrm{authorize}}$ records the authorization information between the doctor d_i and the hospital h_i and it will be published on the blockchain.

5. Specific Implementation of TAAC

Six phases comprise up the precise execution of our suggested strategy, each of which will be discussed in turn below.

5.1. System Initialization. Step 1: RA first takes a security parameter λ as input and selects the multiplicative cycle groups G_1 and G_2, which are produced by the same prime q, then defines $e: G_1 \times G_1 \rightarrow G_2$ as a cryptographic bilinear map. In addition, RA sets two secure hash functions $H_0: \{0, 1\}^* \rightarrow G_1$ and $H_1: G_2 \rightarrow Z_q$.

Step 2: RA randomly selects $x, y, z \in Z_q$, $g, h \in G_1$ are the different generators of G_1, then executes $\mathrm{Setup}(1^\lambda) \rightarrow (\mathrm{MPK}, \mathrm{MSK})$ to generate the public key and the master secret key, where $\mathrm{MPK} = (q, e, g, h, G_1, G_2, H_0, H_1)$, $\mathrm{MSK} = (x, y, z)$.

```
Input: The permission trust threshold set $T$;
       The private key $SK_h$ of the hospital $h_i$;
       The ciphertext storage address PTStoreAddress;
Output: The storage transaction $Tx_{\mathrm{storage}}$;
/*Generate zero-knowledge proof information*/
    $\pi, R, H \leftarrow ZKProveGen(T)$;
/*Calculate the message digest for $T$*/
    $MD = H_0(T)$;
/*Sign the message digest with the $SK_h$*/
    $\sigma_h = \mathrm{Sig}(SK_h, MD)$;
/*Generate the storage transaction*/
    $Tx_{\mathrm{storage}} = \mathrm{PTStoreAddress}, \pi', R', H', \sigma_h\}$;
return $Tx_{\mathrm{storage}}$;
```

Algorithm 1: Storage transaction generation.

Step 3: The hospital h_i provides RA with a special identification number ID_h so that RA can create the key pair $(\mathrm{PK}_h, \mathrm{SK}_h)$ for the hospital using algorithm $\mathrm{Reg}(\mathrm{MPK}, \mathrm{MSK}, \mathrm{ID}_h) \rightarrow (\mathrm{PK}_h, \mathrm{SK}_h)$.

Step 4: RA randomly chooses $a_i \in Z_q$ $(i = 1, 2, 3)$ and $A_i = \mathrm{ID}_h + g^{a_i}$ computes. Finally, the public key $\mathrm{PK}_h = (\{a_i\}_{1 \leq i \leq 3})$ and the private key $\mathrm{SK}_h = (\{A_i\}_{1 \leq i \leq 3})$ of the hospital h_i will be generated.

5.2. Broadcast in Blockchain. Step 1: The hospital h_i sets a trust threshold Pt_i for each permission to obtain the permission trust threshold set $T = \{\mathrm{Pt}_1, \mathrm{Pt}_2, \ldots, \mathrm{Pt}_n\}$. Then, the hospital h_i will encrypt the T by $\mathrm{Enc}(\mathrm{MPK}, \mathrm{PK}_h, T) \rightarrow \mathrm{CT}$ to get the ciphertext $\mathrm{CT} = \{\mathrm{ct}_1, \mathrm{ct}_2, \ldots, \mathrm{ct}_n\}$.

Step 2: The hospital h_i is going to generate a zero-knowledge proof information set $Proof_h = \{\pi', R', H'\}$ by zk-SNARKs depending on T when the permission trust set T has been established. The same method can be implemented by doctors to obtain personal private information relating to zero-knowledge proof.

Step 3: The hospital h_i computes the message digest of T and produces a digital signature by $\sigma_h = \mathrm{Sig}(\mathrm{SK}_h, H_0(T))$. The hospital h_i will utilize a storage transaction $Tx_{\mathrm{storage}} = \mathrm{PTStoreAddress}, \pi', R', H', \sigma_h\}$ to submit the ciphertext address to the blockchain in order to accomplish trust matching. PTStoreAddress represents the ciphertext storage location that is used to index the ciphertext CT.

Step 4: The other nodes in the blockchain will use the signature σ_h to determine whether the transaction is valid after the Tx_{storage} is generated. The transaction between the hospital h_i and the doctor d_i will be uploaded to the blockchain after the verification is completed.

A storage transaction's entire generating process is shown in Algorithm 1.

5.3. Trust Evaluation. In the medical background, the definition of trust for a specific user is the entire assessment of the credibility. Typically, the assessment is based on people who

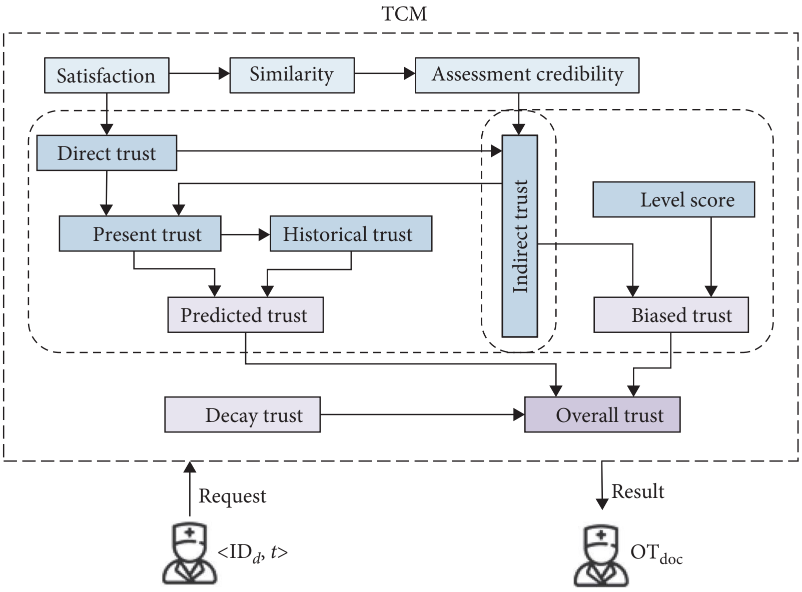


Figure 4: Trust calculation model.

have had direct or indirect interactions with the person being assessed and uses a trust model to calculate the user's trust value based on user feedback, prior behavior, and daily observations. In the proposed scheme, TCM offers an efficient method for calculating user trust value.

The procedure employed by TCM to determine each doctor's trust by using a variety of quantitative indicators is shown in Figure 4. The arrows in Figure 4 illustrate how one factor depends on another. A doctor can provide a set of information $<\mathrm{ID}_d, t>$ to TCM to calculate the personal trust value, where ID_d is used to identify the doctor and t represents the local timestamp. Supposed the hospital assessor h needs to calculate the reliability of the doctor d, the specific process is made up of the following components.

Step 1: The satisfaction $\mathrm{Sat}^{t+1}(h,d)$ denotes the job appraisal of the assessor h on the doctor d in the recent period, and the trust is positively related to the satisfaction. It can be computed as Formula (1):

$$\mathrm{Sat}^{t+1}(h,d) = \beta \times \mathrm{Sat}_{\mathrm{cur}} + (1-\beta) \times \mathrm{Sat}^{t-1}(h,d), \tag{1}$$

where if $\beta = 0$, then $\mathrm{Sat}^{t+1}(h,d) = \mathrm{Sat}^{t-1}(h,d)$; if $\beta = 1$, then $\mathrm{Sat}^{t+1}(h,d) = \mathrm{Sat}_{\mathrm{cur}}$.

Let $\mathrm{Sat}^{t-1}(h,d)$ represents the satisfactory value of h to d within $t-1$ time interval, which is determined by the exponential average of the prior satisfaction in Formula (1). For the recent satisfaction $\mathrm{Sat}_{\mathrm{cur}} \in [0,1]$, $\mathrm{Sat}_{\mathrm{cur}} = 1$ indicates that h is completely satisfied with d, while $\mathrm{Sat}_{\mathrm{cur}} = 0$ indicates that h is completely dissatisfied with d. In addition, β is a carefully selected relative weight value to ensure that $\mathrm{Sat}_{\mathrm{cur}}$ has a higher weight value than $\mathrm{Sat}^{t-1}(h,d)$.

Step 2: As shown in Formula (2), the similarity $\mathrm{Sim}^{t+1}(h_i,h_j)$ measures the degree of similarity between the feedback given by two different assessors h_i and h_j to the same doctor d. The higher similarity leads to the higher precision of trust calculation. The similarity is calculated based on the feedback from people who have contact with the doctor d:

$$\mathrm{Sim}^{t+1}(h_i,h_j) = \begin{cases} \mathrm{Sim}^t(h_i,h_j) + \varphi \times \left(1 - \mathrm{Sim}^t(h_i,h_j)\right), & \text{if } \mathrm{ED}^{t+1}(h_i,h_j) \le \Delta \\ \mathrm{Sim}^t(h_i,h_j) - \vartheta \times \mathrm{Sim}^t(h_i,h_j), & \text{if } \mathrm{ED}^{t+1}(h_i,h_j) > \Delta \end{cases}, \tag{2}$$

where $\mathrm{ED}^{t+1}(h_i,h_j)$ is defined as the evaluation difference and Δ is a similarity deviation constant indicating the upper limit of the permissible variation, as shown in Formula (3). The difference value no greater than the fluctuation constant Δ would indicate that the feedback from the two assessors was closer. SA is the evaluator set of d except h_i. What's more, the reward and punishment coefficients φ and ϑ for updating the similarity are respectively set to reward evaluators for their work and prevent evaluators from providing fake feedback on d. Since it is more difficult to establish trust than to lose it, the punishment coefficient is assigned a greater weight value than the reward coefficient, that is $0 < \varphi < \vartheta < 1$.

$$\mathrm{ED}^{t+1}\left(h_i, h_j\right)=\begin{cases}\sqrt{\sum_{h_j \in \mathrm{SA}} \dfrac{\left[\mathrm{Sat}^{t+1}\left(h_i, d\right)-\mathrm{Sat}^{t+1}\left(h_j, d\right)\right]^2}{|\mathrm{SA}|}}, & \text{if } |\mathrm{SA}|>0 \\ 0, & \text{if } |\mathrm{SA}|=0\end{cases}. \tag{3}$$

Step 3: As shown in Formula (4), the assessment credibility $\mathrm{AC}^{t+1}(h, d)$ indicates the accuracy of the feedback provided by the evaluator. The more similar result of two assessors implies a higher assessment credibility. It is calculated according to the direct logarithm function of similarity, and $\theta=0.01$ represents the minimum allowable value of similarity.

$$\mathrm{AC}^{t+1}(h, d)=\begin{cases}1-\dfrac{\ln\left(\mathrm{Sim}^{t+1}\left(h, h_i\right)\right)}{\ln \theta}, & \text{if } \mathrm{Sim}^{t+1}\left(h, h_i\right)>\theta \\ 0, & \text{otherwise}\end{cases}. \tag{4}$$

Step 4: The direct trust $\mathrm{Dir}^{t+1}(h, d)$ is defined as the assessor h calculating the trust value for doctor d from personal experience, which is obtained in accordance with Formula (5):

$$\mathrm{Dir}^{t+1}(h, d)=\mathrm{Sat}^{t+1}(h, d). \tag{5}$$

Step 5: When the assessor h has insufficient direct interactions with the doctor d, h can request others to provide their evaluations of d. Then, the assessor h will calculate the indirect trust value by combining the direct trust of other evaluators and the assessment credibility, where W denotes the set of evaluators f who have had contact with d. The indirect trust $\mathrm{Ind}^{t+1}(h, d)$ is computed as Formula (6):

$$\mathrm{Ind}^{t+1}(h, d)=\begin{cases}\dfrac{\sum_{f \in W-\{h\}} \mathrm{AC}^{t+1}(h, f) \times \mathrm{Dir}^{t+1}(f, d)}{\sum_{f \in W-\{h\}} \mathrm{AC}^{t+1}(h, f)} & \text{if } |W-\{h\}|>0 \\ 0, & \text{if } |W-\{h\}|=0\end{cases}. \tag{6}$$

Step 6: The present trust $\mathrm{Pre}^{t+1}(h, d)$ means the trust of the assessor h to the doctor d at the most recent time, which is calculated based on the number of interactions between h and d, according to the direct trust and the indirect trust, as shown in Formulas (7)–(9):

$$I^{t+1}(h, d)=\frac{\sum_{f \in W-\{h\}} \mathrm{AC}^{t+1}(h, f) \times D^{t+1}(f, d)}{|W-\{h\}|}, \tag{7}$$

$$\alpha=\frac{D^{t+1}(h, d)}{D^{t+1}(h, d)+I^{t+1}(h, d)}, \tag{8}$$

$$\mathrm{Pre}^{t+1}(h, d)=\alpha \times \mathrm{Dir}^{t+1}(h, d)+(1-\alpha) \times \mathrm{Ind}^{t+1}(h, d). \tag{9}$$

Assuming that $D^{t+1}(h, d)$ is the number of direct interactions between h and d, $I^{t+1}(h, d)$ is the average number of interactions between evaluators other than h and d, and α is the weight of direct trust.

Step 7: The historical trust $\mathrm{His}^{t+1}(h, d)$ is the trust calculated from what happened in the past, as shown in Formula (10). Over time, the present trust has become the historical trust. $\partial \in[0,1]$ is defined as a neglected factor that prevents the appraiser h from attempting to forget the past malicious behavior with the current behavior if the appraiser had malicious behavior in the past:

$$\mathrm{His}^{t+1}(h, d)=\frac{\partial \times \mathrm{His}^t(h, d)+\mathrm{Pre}^{t+1}(h, d)}{2}. \tag{10}$$

Step 8: The predicted trust $\mathrm{PT}_{\mathrm{doc}}$ reflects the future expectation of the assessor h on the doctor d, which is calculated from present trust and historical trust, as shown in Formula (11):

$$\mathrm{PT}_{\mathrm{doc}}=\begin{cases}\gamma \mathrm{Pre}^{t+1}(h, d)+(1-\gamma) \mathrm{His}^{t+1}(h, d), & \text{if either Pre or His} \\ 0, & \text{if neither Pre nor His}\end{cases}. \tag{11}$$

The relative weight $\gamma \in(0.5,1.0)$ is dynamically adjusted based on present trust and historical trust to ensure a higher weight is given to present trust $\mathrm{Pre}^{t+1}(h, d)$.

Step 9: The trust level $L^{t+1} \in\{0,1,2,3,4\}$ is defined as the evaluation of patient to the work of doctor, where the number represents the respective ratings of terrible, poor, average, good, and best. Then, the trust level score LS^{t+1} can be computing, as shown in Formulas (12) and (13):

$$\mathrm{LS}^{t+1} = \frac{2 + nL^{t+1}}{n\left(2 + \sum_{i=1}^{n} L_i^{t+1}\right)}, \tag{12}$$

$$\mathrm{LS}^{t+1} = \frac{1}{2}\left(\sum_{i=1}^{n} v(i)\mathrm{LS}^{t+1}\right) + \frac{1}{2}, \text{ where } v(i) = \frac{2(i-1)}{n-1} - 1. \tag{13}$$

Formula (12) indicates a single value result, while Formula (13) calculating the normalized result.

Step 10: As shown in Formula (14), the biased trust $\mathrm{BT}_{\mathrm{doc}}$ is used to handle malicious trust fluctuation of the evaluator h, so as to prevent h from wavering in the evaluation of doctor d and thus affecting the network performance:

$$\mathrm{BT}_{\mathrm{doc}} = \mathrm{Ind}^{t+1}(h, d) \times \mathrm{LS}^{t+1}. \tag{14}$$

Step 11: The medical malpractice decay trust $\mathrm{DT}_{\mathrm{doc}}$ is expressed as a trust deduction mechanism. In this process, each doctor was initially assigned to K trust points, with fixed trust points reduced for each medical incident. In this mechanism, the trust decay coefficient ω is introduced, and the doctor's i^{th} medical malpractice is recorded as $\mathrm{MR}_{\mathrm{doc}}(i)$, then the decay trust value of the doctor is calculated, as shown in Formula (15):

$$\mathrm{DT}_{\mathrm{doc}} = 1 - \frac{\sum_{i=1}^{n} \omega(i)\mathrm{MR}_{\mathrm{doc}}(i)}{K}. \tag{15}$$

Step 12: The overall trust $\mathrm{OT}_{\mathrm{doc}}$ is calculated by combining predicted trust $\mathrm{PT}_{\mathrm{doc}}$, biased trust $\mathrm{BT}_{\mathrm{doc}}$, and medical malpractice decay trust $\mathrm{DT}_{\mathrm{doc}}$, as shown in Formula (16):

$$\mathrm{OT}_{\mathrm{doc}} = \mu_1(\gamma_1 \mathrm{PT}_{\mathrm{doc}} + \gamma_2 \mathrm{BT}_{\mathrm{doc}}) + \mu_2 \mathrm{DT}_{\mathrm{doc}}. \tag{16}$$

For the relative weights $\mu_1 + \mu_2 = 1, \gamma_1 + \gamma_2 = 1$, the trust threshold $\mathrm{OT}_{\mathrm{doc}} \in [0.0, 1.0]$, where $\mathrm{OT}_{\mathrm{doc}} = 1.0$ indicates that the doctor d is completely trusted, and $\mathrm{OT}_{\mathrm{doc}} = 0.0$ indicates that the doctor d is completely untrusted.

5.4. Zero-Knowledge Proof Generation.

5.4. Zero-Knowledge Proof Generation. After obtaining the personal trust value from the TCM, the doctor d_i completes an initial determination of whether the trust level meets the trust thresholds set by the hospital h_i for medical data access privileges. To create zero-knowledge proof π, the doctor must affix his digital signature σ_d to the trust. The digital signature σ_d and zero-knowledge proof information set $Proof_d = \{\pi, R, H\}$ of the doctor will be generated according to the following processes and elaborated by Algorithm 2.

Step 1: The additional parameter $\delta = (\mathrm{ID}_d, t, \mathrm{OT}_{\mathrm{doc}})$ can be calculated according to the unique identifier ID_d of the doctor d_i, the timestamp t, and the assessed trust $\mathrm{OT}_{\mathrm{doc}}$.

```
Input: The identifier $ID_d$ of the doctor $d_i$;
       The private key $SK_d$ of the doctor $d_i$;
       The trust value $OT_{doc}$ of the doctor $d_i$;
       The security parameter $\lambda$;
       The random number $r$;
Output: The zero-knowledge proof $\pi$;
/*Get the current time*/
    $t = \mathrm{TimeNow}()$;
/*Compute the extended information*/
    $\delta = (ID_d, t, OT_{doc})$;
/*Generate the digital signature with the $SK_d$*/
    $\sigma_d = Sig(SK_d, H_0(\delta, r))$;
/*Produce the calculation result*/
    $C(<PK_1, PK_2, \cdots, PK_n>,$
    $<Ot_1, Ot_2, \cdots, Ot_n, r>) \rightarrow R$;
/*Produce the hash value*/
    $C(<Ot_1, Ot_2, \cdots, Ot_n, r>, <ID_d, t>) \rightarrow H$;
/*Calculate the proof key pair*/
    $ZKKeyGen(1^\lambda, C) \rightarrow (GK, VK)$;
/*Obtain the zero-knowledge proof*/
    $ZKProveGen(GK, OT_{doc}, R, H, \sigma_d) \rightarrow \pi$;
return $\pi$;
```

Algorithm 2: Zero-knowledge proof generation.

Step 2: The additional parameter $\delta = (\mathrm{ID}_d, t, \mathrm{OT}_{\mathrm{doc}})$ and a random number r are used as the input conditions for the hash operation $H_0(\delta, r)$. Then the digital signature σ_d by executing $\sigma_d = \mathrm{Sig}(\mathrm{SK}_d, H_0(\delta, r))$ is generated.

Step 3: Assuming $\vec{c_1} \in C_1$ is the proposition and $\vec{c_2} \in C_2$ is the proof, let $C: C_1 \times C_2 \rightarrow C_3$ be a mathematical operation and $R_C = (\vec{c_1}, \vec{c_2}) \subseteq C_1 \times C_2$ be the associated logic computational relationship. The doctor d_i constructs the circuit C, as shown in Figure 5. It takes the public key set $<\mathrm{PK}_1, \mathrm{PK}_2, \ldots, \mathrm{PK}_n>$, the doctor trust value set $\mathrm{OT}_{\mathrm{doc}} = <\mathrm{Ot}_1, \mathrm{Ot}_2, \ldots, \mathrm{Ot}_n, r>$ and the extended data $<\mathrm{ID}_d, t>$ as inputs. In order to confirm the accuracy and accessibility of the data, a circuit computation result R and a hash value H are respectively output.

Step 4: The security parameter λ and the circuit C will be taken as input parameters to execute $ZK\mathrm{KeyGen}(1^\lambda, C) \rightarrow (\mathrm{GK}, \mathrm{VK})$ to compute the key pair. The zero-knowledge proof is created with the help of the key GK, and it is verified with the help of the key VK.

Step 5: The key GK, the doctor trust value $\mathrm{OT}_{\mathrm{doc}}$, the digital signature σ_d, the circuit result R, and the hash value H are utilized as inputs to generate a reliable zero-knowledge proof π by $ZK\mathrm{ProveGen}(\mathrm{GK}, \mathrm{OT}_{\mathrm{doc}}, R, H, \sigma_d) \rightarrow \pi$.

5.5. Smart Contract Verification. The zero-knowledge proof information set $Proof_d$ is provided to the blockchain by the doctor d_i. Without the involvement of a third-party, the smart contract will immediately confirm whether the relevant attestation information of the doctor d_i meets the

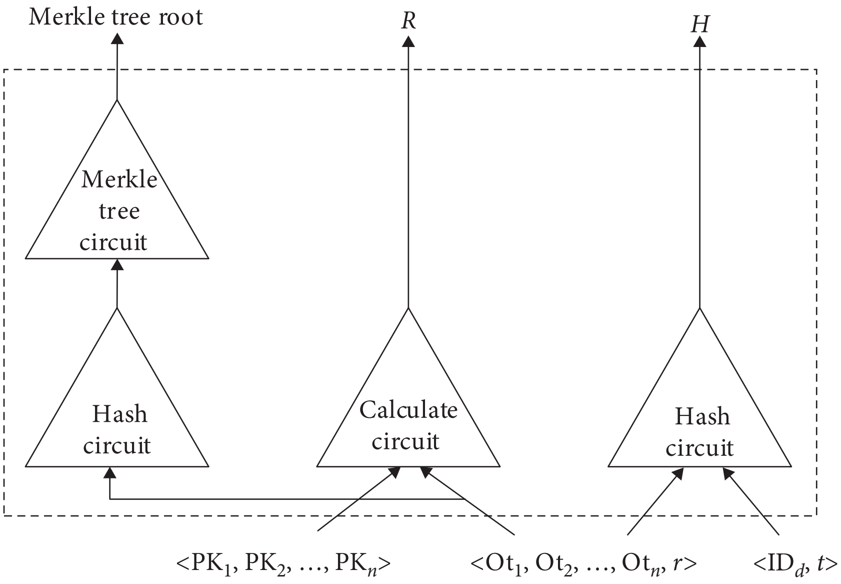


FIGURE 5: Circuit structure diagram.

permission trust threshold set by the hospital h_i. The verification steps are shown as follows.

Step 1: The smart contract utilizes the public key PK_d of the doctor d_i to verify the electronic signature σ_d and return a result SigResult.

Step 2: Then, the verification key VK generated by zk-SNARKs is adopted to contrast the zero-knowledge proof π and return a result ZKResult.

Step 3: Assuming that both validations have been successfully passed, the smart contract further compares the proof information set $Proof_h = \{\pi', R', H'\}$ generated by the hospital h_i based on T with the proof information set $Proof_d = \{\pi, R, H\}$ generated by the doctor d_i based on $\mathrm{OT}_{\mathrm{doc}}$ in correspondence. A result such as True or False will be output if all of the verifications have been successful.

The verification procedure is illustrated in Algorithm 3.

5.6. Authorization Recorded in Blockchain. Step 1: The smart contact in blockchain returns the verification result to the hospital h_i. If the returned result is *True*, the hospital will encrypt and store the list of grantable permissions $\mathrm{PL} = \{P_1, P_2, \cdots, P_n\}$ and the ciphertext address is PLStoreAddress.

Step 2: After receiving the authorization notification from the blockchain, the doctor d_i retrieves the encrypted permission list $\mathrm{CPL} = \{\mathrm{CP}_1, \mathrm{CP}_2, \ldots, \mathrm{CP}_n\}$ through PLStoreAddress.

Step 3: Then the doctor d_i gets the private key SK_h of the hospital h_i through the secure key channel and decrypts the ciphertext by $\mathrm{Dec}(\mathrm{SK}_h, \mathrm{CPL}) \rightarrow \mathrm{PL}$ to obtain the permission list PL.

Step 4: Finally, the hospital h_i generates a transaction to record the authorization information between the doctor d_i and the hospital h_i, which is described by Algorithm 4. The authorization transaction $Tx_{\mathrm{authorize}} = \{\mathrm{ID}_A, \mathrm{ID}_d, \mathrm{PK}_h, \mathrm{PLStoreAddress}, t, \sigma_h\}$ will be published on the blockchain, where ID_A is used to identify authorization transaction and t represents the $Tx_{\mathrm{authorize}}$ generated time.

```
Input: The Verification key $VK$;
       The public key $PK_d$ of the doctor $d_i$;
       The digital signature $\sigma_d$ of the doctor $d_i$;
       The proof information $Proof_h = \{\pi', R', H'\}$ of the
       hospital $h_i$;
       The proof information $Proof_d = \{\pi, R, H\}$ of the
       doctor $d_i$;
Output: The verified information of proof;
/* Verify the digital signature of the doctor */
    $\mathrm{SigResult} = \mathrm{SigVerify}(PK_d, \sigma_d)$;
/* Verify the zero-knowledge proof of the doctor */
    $ZK\mathrm{Result} = ZK\mathrm{Verify}(VK, \pi)$;
/* Compare relevant information */
    if (SigResult = 1 and $ZK$Result = 1) then
      result = $Verify(\pi, \pi', R, R', H, H')$;
      /* Return validation result */
      if result = 1 then
        return True;
    return False;
```

ALGORITHM 3: Verify the zero-knowledge proof.

6. Scheme Analysis

We primarily assess the TAAC method from three angles in this section. First, we analyze whether the scheme meets the basic security and privacy requirements of blockchain operations. Furthermore, the TAAC scheme is comprehensively compared with some existing methods in related work. Finally, extensive experiments are conducted to demonstrate the efficiency.

6.1. Security Analysis. The security of the TAAC scheme is analyzed in five aspects, including trustworthiness, traceability, privacy, integrity, and against DDoS attack.

```
Input: The permission list PL;
       The identifier $ID_A$ of the authorization transaction;
       The identifier $ID_d$ of the doctor $d_i$;
       The key pair $(PK_h, SK_h)$ of the hospital $h_i$;
       The storage address PLStoreAddress of the
       permission list ciphertext $CPL$;
Output: The authorization transaction $Tx_{\text{authorize}}$;
if result is valid then
    /* Get the current time */
    $t = \text{TimeNow}()$;
    /* Compute the message digest for $PL$ */
    $MD = H_0(ID_A, PL)$;
    /* Sign the message digest with the $SK_h$ */
    $\sigma_h = \text{Sig}(SK_h, MD)$;
    /* Generate the authorization transaction */
    $Tx_{\text{authorize}} = \{ID_A, ID_d, PK_h, \text{PLStoreAssress}, t, \sigma_h\}$;
    return $Tx_{\text{authorize}}$;
return $\perp$;
```

Algorithm 4: Authorization transaction generation.

6.1.1. Trustworthiness. In this paper, our TAAC transmits the ciphertext address with digital signature to the blockchain by a storage transaction, and broadcasts it throughout the network nodes. From beginning to end, the entire encrypted broadcast process only involves the hospital and the blockchain, thus guaranteeing the trustworthiness of authorization management with avoiding the possibility of any other intermediate entities trying to steal permission data. In addition, in the verification phase, the blockchain platform automatically verifies whether the doctor′s trust conforms to the authority trust level set by the hospital through a predeployed smart contract, further realizing a trustworthy authorization delegation without manual intervention.

6.1.2. Traceability. Our system can monitor and validate the access data in the blockchain. Any operations related to the blockchain are recorded as an immutable storage transaction and authorization transaction, respectively. Once the storage transactions and authorization transactions are validated by the smart contract, the doctors will be notified of the categories of medical data access rights open to the hospital, and the hospital will query what authorizations have been granted to each doctor. Therefore, if it is detected that the medical data have been maliciously disclosed or changed, neither the hospital nor the doctor can deny it, and can be held accountable retroactively based on the authorization transactions recorded on the blockchain.

6.1.3. Privacy. Most of the existing trust evaluation models directly output the trust value, which poses a great threat to privacy leakage. Our scheme ensures that doctors′s identity information will not be disclosed during trust evaluation by generating a unique identity for each doctor. To solve the problem of trust privacy leakage, the smart contract can only obtain a zero-knowledge proof generated based on trust but not directly get individual trust during the verification process. In terms of authority information privacy, storing permission data in encrypted form can prevent malicious users from stealing sensitive data.

6.1.4. Integrity. In order to increase the scalability of the TAAC, the hospital may sporadically change the authorization data, and the ciphertext connected to the ciphertext address will also be altered. The additional digital signature in the storage transaction and proof generation can verify the ciphertext at any time and ensure the authenticity of the zero-knowledge proof. In addition, the information of authorized transactions between doctors and hospital is recorded on the chain through the consensus algorithm. The blockchain can ensure data are not tampered with by using the Merkle tree feature.

6.1.5. Against DDoS Attack. It is fairly straightforward that our scheme can resist DDoS attack as the blockchain-based architecture is decentralized. Even if the blockchain nodes are attacked maliciously, users can access the network normally as long as one node exists. In this scheme, both storage transactions and authorization transactions must be verified before recording. They need to provide valid digital signatures when interacting with the blockchain network, which is an effective method to thwart DDoS attack.

6.2. Comprehensive Comparisons of TAAC. In Table 2, five related trust-based access control methods are contrasted with the TAAC. This analysis mainly focuses on five aspects such as the dynamics of the trust evaluation model, data encrypted storage, trust privacy, blockchain technology and whether it can resist malicious attacks. The access control scheme designed by Jiang et al. [22] and Hu et al. [24] sets influence factors to dynamically adjust the accuracy rate of the trust evaluation model, but they cannot guarantee the trust privacy. The HealthyBroker trust-building agent architecture designed by Kurdi et al. [19] conducts audit and tracking through the blockchain ledger to prevent potential malicious behavior, but this scheme does not consider the security of patient data. He et al. [26] used cryptographic technology to encrypt and store information without causing data leakage. However, this scheme relies too much on the cooperation and reliability of distributed nodes, which is prone to creating system crash. Lin et al. [14] introduced the Vickrey–Clark–Groves (VCG)-based adaptive reputation mechanism (VARM) into the access control scheme, which can effectively identify malicious users and resist internal attacks, but it does not take into account the security issues arising from transparency.

Overall, these trust-based access control mechanisms are unable to simultaneously protect trust privacy and impede malicious user behavior. The TAAC scheme designed in this paper enables data privacy protection and avoids access initiated by malicious nodes. By storing the encrypted data locally and sending the ciphertext address to the blockchain, the credibility of access authorization with blockchain characteristics can be guaranteed. In terms of privacy, trust verification through zero-knowledge proof can effectively prevent

Table 2: Performance comparisons of different schemes.

Scheme	Dynamics	Encryption	Trust privacy	Blockchain	Resist attack
[19]	×	×	×	✓	✓
[22, 24]	✓	✓	×	×	×
[26]	✓	✓	×	×	✓
[14]	✓	✓	×	×	✓
Our	✓	✓	✓	✓	✓

the privacy leakage of the permission trust and user trust. In addition, TCM uses trust decay coefficient, reward and punishment coefficients, and neglect factor to enhance the sensitivity and dynamics of the model, which can resist the access initiated by internal malicious nodes.

6.3. Performance Evaluation. Since TCM and zk-SNARK constitute two crucial components of the proposed system TAAC, the performance of trust evaluation and zero-knowledge proof are tested separately. For the performance analysis, the experiments are implemented on a computer with Intel(R) Core(TM) i7-4790 U CPU @3.60 GHz, 12 GB of RAM, and Ubuntu Linux 18.04 LTS. We publish the computed trust data and shared information on Hyperledger Fabric and write smart contracts in the Go language. Zero-knowledge proofs are implemented using libSNARK-based zk-SNARK provided by Electric Coin Company.

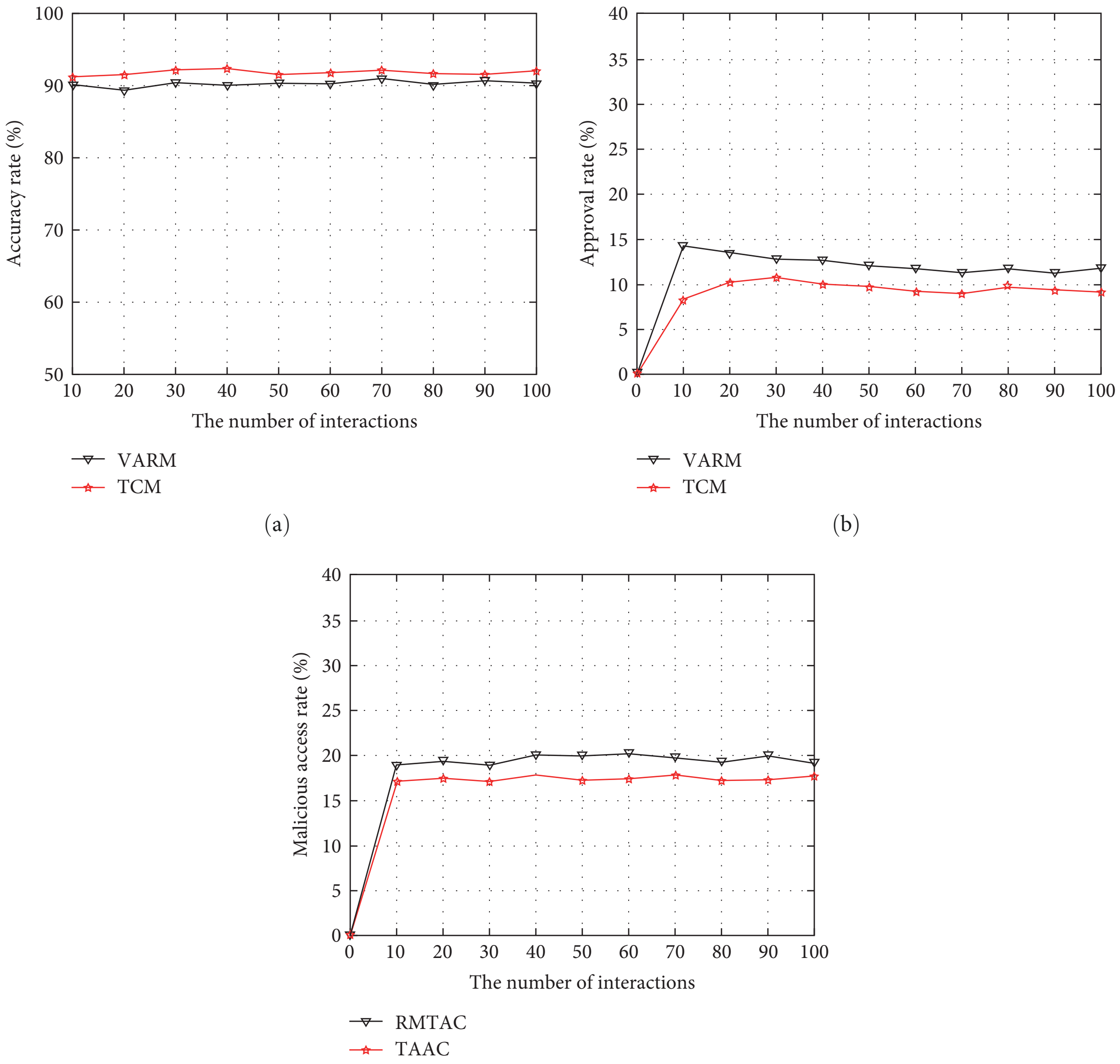
For the experimental evaluation of the proposed system, we used data from a medical database [43]. One hundred virtual users are given random trust values, which represent their actual trust values. These assigned parameters are utilized to calculate the proposed system accuracy rate. The experiments initially began with 10 interactions, after which 10 interactions are added each time. Each experiment is repeated for 20 times and the average is calculated. To measure the performance of this scheme, we will compare it with RMTAC based on VARM [14] in terms of the accuracy rate, approval rate, and system malicious access rate. The theoretical analyses and performance comparisons between the schemes are shown in Figure 6.

6.3.1. Accuracy Rate. It is described as the proportion between the calculated trust value and the real trust value. According to Table 3 and Figure 6(a), TCM has a marginally greater accuracy rate than VARM. The accuracy rate of VARM basically remains around 90%, while that of TCM fluctuates slightly around 91.5%. The reason is that VARM only considers the reward factor to motivate users to provide more accurate recommendations but does not set a penalty factor to avoid users′ dishonest recommendations. On the contrary, TCM establishes a tight system of rewards and penalties that can efficiently reward users who provide trustworthy suggestions and penalize harmful users who produce untruthful comments. Therefore, the accuracy rate of TCM is marginally higher than that of VARM.

6.3.2. Approval Rate. It is defined as the ratio of the number of successful access that does not meet the security requirements to the total number of access requests. The results in Table 4 and Figure 6(b) illustrate that TCM has a lower approval rate compared to VARM. It is assumed that there are a fixed number of malevolent users at the beginning, and their trust does not satisfy the security requirements. As the quantity of user interactions grows, the approval rates of both VARM and TCM tend to a stable range. The approval rate of VARM is basically stable around 12%, while that of TCM fluctuates around 10%. This means that TCM exhibits better attack prevention ability than VARM under the same number of malicious users. There are two main reasons for the discrepancy in result. False comments cannot be avoided due to the existence of malicious users in the initial stage. VARM only adaptively adjusts the initial trust of unknown users according to the actual situation of the network, without considering the punishment of malicious behavior. In contrast, TCM adds the historical trust and introduces a neglect factor in calculating doctor trust, which can prevent the doctor from trying to cover up past irregularities with present compliance behaviors. In addition, TCM designs a trust deduction mechanism to reduce fixed trust points according to the trust attenuation coefficient if the doctor commit misconduct, which will curb the occurrence of malicious access to some extent.

6.3.3. Malicious Access Rate. It is used to evaluate the privacy protection capability of the proposed TAAC system. The malicious access rate is defined as the percentage of successful access to illegal information in the whole interaction, supposing that some access requests from malicious users have been granted in the experiment. The comparison result is demonstrated in Table 5 and Figure 6(c). Since RMTAC does not implement fine-grained division of permissions, that means, users can use ungranted access rights when legitimate identity users cheat. This will cause authorized users to tamper with or disclose information, thereby increasing the malicious access rate. TAAC verifies the trust of each permission by introducing zero-knowledge proof protocol. Only the doctor who meets the authority trust level can obtain the corresponding authority, and the encrypted authorization list can also prevent attackers from stealing access rights. Therefore, the average malicious access rate of TAAC is lower than that of RMTAC.

Next, the performance of zero-knowledge proof is evaluated. The time required to generate evidence is the main bottleneck of this technique since the noninteractive zero-knowledge proof model is utilized. By simulating the process of the model, this experiment focuses on evaluating the zk-SNARK key pair generation time, proof generation time, and proof verification time. The experiment starts with 100 permission trust values as the circuit inputs, and then the test is



(a)

(b)

(c)

Figure 6: Performance analyses: (a) accuracy rate, (b) approval rate, and (c) malicious access rate.

Table 3: Accuracy rates of different interactions.

Scheme	10	20	30	40	50	60	70	80	90	100
[14]	90.014	89.217	90.890	90.005	90.898	90.692	91.566	90.013	90.981	90.221
Our	91.513	91.556	92.394	92.890	91.533	91.724	92.393	91.688	91.509	92.006

Table 4: Approval rates of different interactions.

Scheme	10	20	30	40	50	60	70	80	90	100
[14]	14.010	13.532	13.014	12.993	12.102	11.820	11.271	11.807	11.116	11.915
Our	8.160	10.004	10.992	10.001	9.700	9.005	8.899	9.603	9.315	9.002

Table 5: Malicious access rates of different interactions.

Scheme	10	20	30	40	50	60	70	80	90	100
[14]	18.878	19.186	18.872	20.001	19.997	20.102	19.578	19.013	20.001	18.997
Our	17.466	17.508	17.165	17.897	17.388	17.484	17.889	17.002	17.191	17.566

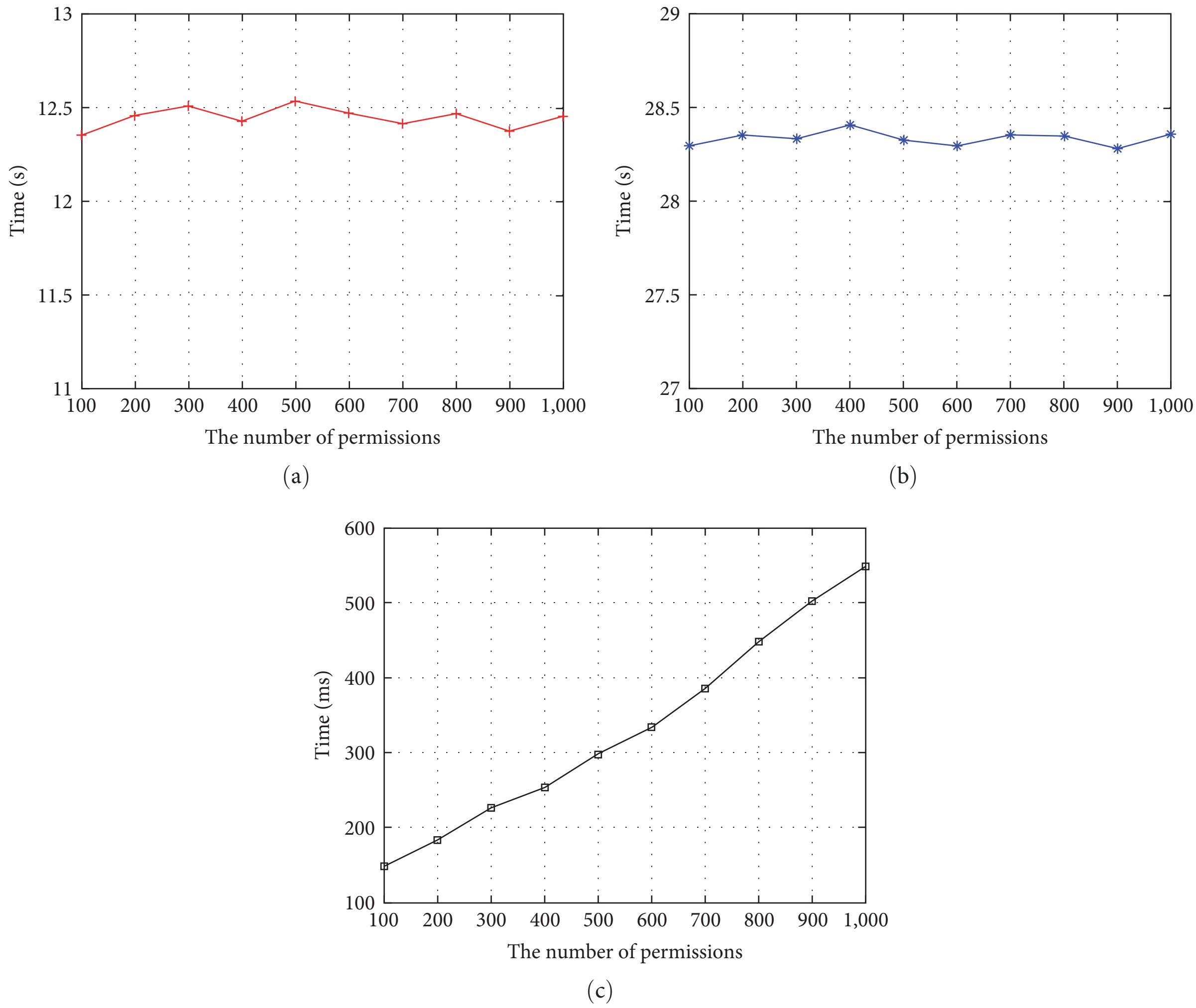


Figure 7: Zero-knowledge proof results: (a) key pair, (b) proof generation, and (c) proof verification.

repeated for 20 times. Finally, the average value of these indicators is then determined.

As observed in Figures 7(a) and 7(b), the proposed scheme takes approximately 12.5 s to generate a zero-knowledge proof key pair and 28.3 s to produce a proof. This indicates that the time required to construct a zero-knowledge proof key pair and a proof will not vary much even when the authorization data used as circuit inputs increases, dramatically boosting the scalability of the scheme. Furthermore, Figure 7(c) illustrates when the quantity of input parameters expands, the verification time for zero-knowledge proofs similarly increases. The result indicates that the validation time does not exceed 0.6 s when the number of input parameters reaches 1,000, which is still an acceptable time limit and does not affect concurrency.

7. Conclusion

In this paper, we propose a TAAC scheme, which can resolve the issues of high-cost authorization management and internal malicious node attacks caused by the intervention of patient entities into the institutional entities. We first design an authorization verification model and describe in detail the specific processes of the entities and schemes involved in the model. Then, we construct a dynamic trust computation model TCM to evaluate the trust of users and elaborate on the zero-knowledge proof generation based on the trust and verification authorization process. Finally, the security analysis and performance comparisons between the proposed scheme and the existing schemes are carried out. The experimental results show that the scheme is more secure, reliable, and efficient in terms of privacy protection, trust evaluation, and resisting malicious access. Future research efforts will focus on further optimizing the trust evaluation model to improve the accuracy of trust calculation while reducing the malicious access rate. Meanwhile, the efficiency of zero-knowledge proof can be also improved by meliorating the implementation process.

Data Availability

The data used to support the findings of this study are openly available in the UCI Machine Learning Repository database at https://archive.ics.uci.edu/dataset/296/diabetes+130-us+hospitals+for+years+1999-2008.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research is partially supported by the National Natural Science Foundation of China under Grant 61972438, the Anhui Provincial Department of Education University Scientific Research Project under Grant KJ2020A0694, Key Research and Development Projects in Anhui Province under Grant 2022a05020049, and the Natural Science Foundation of Anhui Province of China under Grant 2108085MF219.

References

[1] A. A. Abdellatif, A. Z. Al-Marridi, A. Mohamed, A. Erbad, C. F. Chiasserini, and A. Refaey, "sshealth: toward secure, blockchain-enabled healthcare systems," *IEEE Network*, vol. 34, no. 4, pp. 312–319, 2020.

[2] Y. Zhuang, L. R. Sheets, Y.-W. Chen, Z.-Y. Shae, J. J.-P. Tsai, and C.-R. Shyu, "A patient-centric health information exchange framework using blockchain technology," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 8, pp. 2169–2176, 2020.

[3] H. Huang, P. Zhu, F. Xiao, X. Sun, and Q. Huang, "A blockchain-based scheme for privacy-preserving and secure sharing of medical data," *Computers & Security*, vol. 99, Article ID 102010, 2020.

[4] S. Nakamoto, "Bitcoin: a peer-to-peer electronic cash system," Decentralized Business Review, p. 21260, 2008.

[5] R. Zhang, R. Xue, and L. Liu, "Security and privacy on blockchain," *ACM Computing Surveys (CSUR)*, vol. 52, no. 3, pp. 1–34, 2019.

[6] Y. Zhao, M. Cui, L. Zheng et al., "Research on electronic medical record access control based on blockchain," *International Journal of Distributed Sensor Networks*, vol. 15, no. 11, pp. 1–11, 2019.

[7] C. Gan, A. Saini, Q. Zhu, Y. Xiang, and Z. Zhang, "Blockchain-based access control scheme with incentive mechanism for eHealth systems: patient as supervisor," *Multimedia Tools and Applications*, vol. 80, no. 20, pp. 30605–30621, 2021.

[8] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: using blockchain for medical data access and permission management," in *2016 2nd International Conference on Open and Big Data (OBD)*, pp. 25–30, IEEE, Vienna, Austria, 2016.

[9] S. Shamshad, Minahil, K. Mahmood, S. Kumari, and C.-M. Chen, "A secure blockchain-based e-health records storage and sharing scheme," *Journal of Information Security and Applications*, vol. 55, Article ID 102590, 2020.

[10] X. Cheng, F. Chen, D. Xie, H. Sun, and C. Huang, "Design of a secure medical data sharing scheme based on blockchain," *Journal of Medical Systems*, vol. 44, no. 2, pp. 1–11, 2020.

[11] H. Wang, S. Wu, M. Chen, and W. Wang, "Security protection between users and the mobile media cloud," *IEEE Communications Magazine*, vol. 52, no. 3, pp. 73–79, 2014.

[12] G. Nyame, Z. Qin, K. O.-B. O. Agyekum, and E. B. Sifah, "An ECDSA approach to access control in knowledge management systems using blockchain," *Information*, vol. 11, no. 2, Article ID 111, 2020.

[13] H. Wang and Y. Song, "Secure cloud-based EHR system using attribute-based cryptosystem and blockchain," *Journal of Medical Systems*, vol. 42, no. 8, pp. 1–9, 2018.

[14] H. Lin, L. Xu, X. Huang, W. Wu, and Y. Huang, "A trustworthy access control model for mobile cloud computing based on reputation and mechanism design," *Ad Hoc Networks*, vol. 35, pp. 51–64, 2015.

[15] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *Journal of Medical Systems*, vol. 42, no. 7, pp. 1–7, 2018.

[16] O. Goldreich, S. Micali, and A. Wigderson, "Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems," *Journal of the ACM*, vol. 38, no. 3, pp. 690–728, 1991.

[17] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Scalable zero knowledge via cycles of elliptic curves," *Algorithmica*, vol. 79, no. 4, pp. 1102–1160, 2017.

[18] H. Sukhwani, J. M. Martínez, X. Chang, K. S. Trivedi, and A. Rindos, "Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric)," in *2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS)*, pp. 253–255, IEEE, Hong Kong, China, 2017.

[19] H. Kurdi, S. Alsalamah, A. Alatawi, S. Alfaraj, L. Altoaimy, and S. H. Ahmed, "HealthyBroker: a trustworthy blockchain-based multi-cloud broker for patient-centered eHealth services," *Electronics*, vol. 8, no. 6, Article ID 602, 2019.

[20] V. Sahaana, A. S. Preetha, and R. Sukanesh, "A novel decentralized trust evaluation model for secure mobile healthcare systems," in *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pp. 1–5, IEEE, Chennai, India, 2014.

[21] R. Lewandowski, A. G. Goncharuk, and G. T. Cirella, "Restoring patient trust in healthcare: medical information impact case study in Poland," *BMC Health Services Research*, vol. 21, no. 1, pp. 865–875, 2021.

[22] R. Jiang, Y. Xin, H. Cheng, and W. Wu, "T-RBAC model based on two-dimensional dynamic trust evaluation under medical big data," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 9957214, 17 pages, 2021.

[23] A. Singh and K. Chatterjee, "ITrust: identity and trust based access control model for healthcare system security," *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 28309–28330, 2019.

[24] X. Hu, R. Jiang, M. Shi, and J. Shang, "A privacy protection model for health care big data based on trust evaluation access control in cloud service environment," *Journal of Intelligent & Fuzzy Systems*, vol. 38, no. 3, pp. 3167–3178, 2020.

[25] J. Guo and X. Zhang, "VARTE: trust evaluation model based on VAR for mobile medical information system," in *2020 6th International Conference on Big Data Computing and Communications (BIGCOM)*, pp. 96–102, IEEE, Deqing, China, 2020.

[26] D. He, C. Chen, S. Chan, J. Bu, and A. V. Vasilakos, "A distributed trust evaluation model and its application scenarios for medical sensor networks," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 6, pp. 1164–1175, 2012.

[27] G. Athanasiou, G. C. Anastassopoulos, E. Tiritidou, and D. Lymberopoulos, "A trust model for ubiquitous healthcare environment on the basis of adaptable fuzzy-probabilistic inference system," *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 4, pp. 1288–1298, 2018.

[28] Y. Xin, Z. Qian, R. Jiang, and Y. Song, "Trust evaluation strategy based on grey system theory for medical big data," in *2019 IEEE International Conference on Computer Science and*

Educational Informatization (CSEI), pp. 157–160, IEEE, Kunming, China, 2019.

[29] Z. Rahman, I. Khalil, X. Yi, and M. Atiquzzaman, "Blockchain-based security framework for a critical industry 4.0 cyber-physical system," *IEEE Communications Magazine*, vol. 59, no. 5, pp. 128–134, 2021.

[30] A. A. Siyal, A. Z. Junejo, M. Zawish, K. Ahmed, A. Khalil, and G. Soursou, "Applications of blockchain technology in medicine and healthcare: challenges and future perspectives," *Cryptography*, vol. 3, no. 1, Article ID 3, 2019.

[31] K. Fan, S. Wang, Y. Ren, H. Li, and Y. Yang, "MedBlock: efficient and secure medical data sharing via blockchain," *Journal of Medical Systems*, vol. 42, no. 8, pp. 1–11, 2018.

[32] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani, "MeDShare: trust-less medical data sharing among cloud service providers via blockchain," *IEEE Access*, vol. 5, pp. 14757–14767, 2017.

[33] H. M. Hussien, S. M. Yasin, N. I. Udzir, and M. I. H. Ninggal, "Blockchain-based access control scheme for secure shared personal health records over decentralised storage," *Sensors*, vol. 21, no. 7, Article ID 2462, 2021.

[34] T. Feng, H. Pei, R. Ma, Y. Tian, and X. Feng, "Blockchain data privacy access control based on searchable attribute encryption," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 871–890, 2021.

[35] T. T. Thwin and S. Vasupongayya, "Blockchain-based access control model to preserve privacy for personal health record systems," *Security and Communication Networks*, vol. 2019, Article ID 8315614, 15 pages, 2019.

[36] J. Sun, L. Ren, S. Wang, X. Yao, and H. Debiao, "A blockchain-based framework for electronic medical records sharing with fine-grained access control," *PLOS ONE*, vol. 15, no. 10, Article ID e0239946, 2020.

[37] A. Saini, Q. Zhu, N. Singh, Y. Xiang, L. Gao, and Y. Zhang, "A smart-contract-based access control framework for cloud smart healthcare system," *IEEE Internet of Things Journal*, vol. 8, no. 7, pp. 5914–5925, 2021.

[38] S. N. G. Gourisetti, M. Mylrea, and H. Patangia, "Evaluation and demonstration of blockchain applicability framework," *IEEE Transactions on Engineering Management*, vol. 67, no. 4, pp. 1142–1156, 2019.

[39] Z. Zheng, S. Xie, H. N. Dai, X. Chen, and H. Wang, "Blockchain challenges and opportunities: a survey," *International Journal of Web and Grid Services*, vol. 14, no. 4, pp. 352–375, 2018.

[40] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A survey of distributed consensus protocols for blockchain networks," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1432–1465, 2020.

[41] N. B. Truong, K. Sun, G. M. Lee, and Y. Guo, "GDPR-compliant personal data management: a blockchain-based solution," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1746–1761, 2019.

[42] P. Alikhani, N. Brunner, C. Crépeau et al., "Experimental relativistic zero-knowledge proofs," *Nature*, vol. 599, no. 7883, pp. 47–50, 2021.

[43] S. Beata, "UCI machine learning repository: diabetes 130-US hospitals for years 1999-2008 data set," 2014, [Online]. Available: https://archive.ics.uci.edu/dataset/296/diabetes+130-us+hospitals+for+years+1999-2008.