IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 12, 2022, accepted February 23, 2022, date of publication March 3, 2022, date of current version March 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3156591

Blockchain Bridges Critical National Infrastructures: E-Healthcare Data Migration Perspective

YIYING LIU[1], GUANGXING SHAN[2], YUCHENG LIU[3], ABDULLAH ALGHAMDI[4], IQBAL ALAM[5], AND SUJIT BISWAS[6,7], (Member, IEEE)

[1]Computer Science Department, Kunming University of Science and Technology, Kunming 650500, China
[2]Wuhan Real Estate Information Centre, Management Department, Hubei 430015, China
[3]Huacheng Luzhou Phase, People's Public Security University of China, Beijing 050000, China
[4]Information Systems Department, College of Computer Science and Information Systems, Najran University, Najran 55461, Saudi Arabia
[5]Academic Department, Nanyang Academy of Sciences (NASS), Beijing 102400, China
[6]Computer Science and Engineering Department, Faridpur Engineering College, University of Dhaka, Dhaka 1000, Bangladesh
[7]CVSSP, University of Surrey, Guildford GU2 7XU, U.K.

Corresponding authors: Guangxing Shan (53050095@qq.com) and Sujit Biswas (sujitedu@gmail.com; sujitbiswas@ieee.org)

This work was supported in part by the Information and Communication Technology Division (ICTD), Bangladesh, under Grant Ref-004.20.362; and the authors are thankful to the Deanship of Scientific Research at Najran University for funding partially this work under the Research Collaboration Funding program grant code NU/RC/SERC/11/10.

ABSTRACT Secure management of Critical National Infrastructures (CNI) is a burning challenge to any state. As a CNI, Electronic Healthcare System (EHS) infrastructure records citizens' medical records, raising security and privacy concerns. Traditional EHS functions independently where patients' records are recorded and maintained in centralized systems that produce massive redundant data. Due to the non-coherence of these systems, data atomicity is not maintained; hence research results based on these data create questioning. Moreover, medical records are valuable for research but cannot be public due to security and privacy. Blockchain (BC) is currently considered a potential solution for the challenges. Blockchain can integrate every independent EHS as a bridging platform. The solution can ensure data uniqueness and overcome security issues. The prime difficulties for the integration are data synchronization of the traditional EHS and BC-based EHS. Furthermore, the autonomous interoperability between *SQL* and *NoSQL* database used in typical EHS and BC-based EHS, respectively, is a prime challenge. Therefore, this research proposes a Blockchain-based framework that bridges Traditional E-Health Systems(TEHS) and allows uninterruptible data exchanges between two systems, even for archive medical records. Beyond that, the framework shows an elevated way to overcome a single point of failure, data security, access control, etc., issues in a centralized system. Finally, the testbed implementation justifies the proposed architecture.

INDEX TERMS Critical infrastructure, blockchain, healthcare, IoT, IoMT, distributed ledger technology, interoperability, information security, security and privacy.

I. INTRODUCTION

The The critical national infrastructures are national assets, systems, technologies, networks, and services. CNI can be stand-alone or interconnected and interdependent within and across provinces, territories, and national borders. Destruction of any of these components can play a devastating role and catastrophic loss of life and create the ultimate effect on a state in various ways such as economy, social well-being, public safety, and the functioning of crucial government responsibilities. Pinpoint monitoring of structural CNI such as integrity of the structure (i.e., valuable landmark, bridges, national defense headquarters, etc.) is a visible responsibility to monitor by the physical workforce. Beyond such a physical CNI, recently, every state has complex logical structures in different application domains such as e-healthcare systems (EHS), Smart Grid, Data Center, Cyber security architectures, etc. Considering the importance of Healthcare on a national security system, different states consider these sectors as one of the vital CNI. For example, the UK government

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Zakarya.

considers second out of thirteen CNI [1], United States gives fourth out of sixteen [2]. Similarly, other countries also added Healthcare to the CNI list.

Due to the digitization effect, most CNI is somehow linked with Cyber-Physical Systems (CPS). For example, the advancement of ICT greatly impacted healthcare sectors due to Electronic Healthcare System (EHS), Mobile Healthcare System (MHS), Tele-medicine, etc. During natural disasters and emergencies, a nation's continuity or survival mostly depend on a solid healthcare system. While records are kept digitally, service interruptions in emergencies can play a devastating role for a nation. Moreover, it has a significant impact on the economy; for example, during COVID-19, world GDP reduces 2.4% by 2020 [3]. However, EHS is different from other CNI. The most challenging issues are local and remote public health service record management, spreader national, state, regional, local, tribal, territorial, etc. Although record management services are handled centrally, health services and relevant records are not adequately maintained. These services are running independently and healthcare-specific centralized service-based. Civil administration controls such organizations with pen-and-paper rules. In some cases, there is no specific policy or monitoring of how a private organization is heading, holding, or managing patients' records.

Healthcare organizations are the most insecure critical infrastructures that can potentially impact other sectors and the national security [4]. For example, straightforward scenarios in the health sector can cause national chaos. In industry 4.0, EHS data is being collected using wireless sensors, Internet of Medical Thing (IoMT), etc., which creates enormous challenges. Such smart devices monitor patients' real-time events and generate big data. In most cases, cyber-physical security problems arising from critical infrastructures cannot be neglected if they are not well managed. The existing EHS system is mostly managed independently using a centralized system that appears many security challenges such as single point failure, data security, the privacy of end-users, etc. The most complicated challenge is the interoperability of EHSs and data sharing with a different government organization. Data sharing with government organizations is an unavoidable condition for e-Governance. But it has very much potentiality to arise many security and privacy challenges. Therefore, most cyber security specialists recommend a smart agreement-dependent, decentralized, encrypted data management mechanism. Blockchain is such technology that can overcome the issues.

Blockchain is a Peer-to-Peer (P2P) Distributed Ledger Technology (DLT) where transactions are recorded in a block. Multiple (more than two) interconnected peers form a P2P network, and every peer holds a ledger. Ledger is the sequential connections of blocks which are the aggregation of the hash of transactions executed in a particular moment [5]. Any newly generated block only can be added with the ledger if it passes through the consensus of peers. The consensus process ensures proof-of-work and positive concern of maximum peers that conforms legality of transactions. For example, if block formation time is 60 sec, and all peers execute 200 transactions in 60sec time, then the block contains encrypted 200 transactions. Finally, the block is labeled as a hash of all transactions. It includes the key security features such as a) transactions are encrypted and stored distributedly that protects information leakage and data loss; 2) P2P network provides non-stop service networks that overcome a centralized system's common single point failure issues; 3) consensus process reconfirms the authenticity of transactions that protect intruders.

Several recent articles have discussed the various benefits of blockchain technology regarding EHS data management and the tools (i.e., DataX, Apache Sqoop, etc.) for decentralized management of big data. However, the technical solution for integrating blockchain systems with data management systems remains elusive. Moreover, existing systems that need to be upgraded, such as Centralized E-health System (CEHS) to Decentralized E-Health System (DEHS), are difficult to shut down and run using Blockchain Technology. To upgrade existing systems, smart migration, synchronization, and access-friendly system are crucial. Typical NoSQL or SQL data management tools are also incapable of handling the challenges. Instead, any hybrid full-duplex tool can solve the issue. In this paper, we have proposed such a migration mechanism named *Triple*. It can transform the NoSQL data to a triple format and incorporate these triples in the SQL database as a virtual relation. It leverages the query process to avoid a series of self-joins by reconstructing the NoSQL data from the triple association.

We have proposed a Blockchain-based architecture for synchronizing typical CEHS and DEHS data runtime and distributed ledger management. The core contribution of the article includes-

- a CEHS to DEHS data migration framework.
- an optimization policy for heavy the transaction that contains medical images.
- an optimistic smart way for SQL and NoSQL data migration.
- implementation result presents the effectiveness.

The consequent contents have been organized into four sections. Section II presents the summary of the state art of the contributions. The proposed architecture detailing every component has been described in Section III. Section IV illustrates the technical details of migration process. Implementation results, discussion including testbed details are discussed in Section V. Finally, Section VI conclude the overall contributions.

II. RELATED WORKS

Several research efforts have been made to integrate Blockchain into e-health systems. Targeting identity management of EHS users, a permissioned Blockchain-based security framework has been proposed in [6]. The authors propose an authentication scheme and use a mutual authentication key has been suggested. Likewise, authors [7]

present a compound key for every group of users (i.e., patient self, physicians, nurses, caregiver, etc.) who are related for treatment of the patients. They also suggested a specific channel for every patient who owns the channel where other service providers are members of that channel. The goal is patient privacy and authentication. Integration importance of IoT-based healthcare system (i.e., EHS) and Blockchain is described in [8]. The primary focuses on merging the technologies and promoting a secure system. Contrary, [9] discusses EHS and BC integration challenges and benefits. They focused on some generic challenges of Blockchain protocols, such as scalability issues generated by medical data. Some contributions proposed the lightweight consensus algorithm as a solution to scalability issues. In this regard, [10], [11] proposes a lightweight consensus algorithm for different application use cases as a solution to scalability challenges. Besides authentication policies, medical records are vital for security. E-Health Record (EHR) management and its sharing policy framework proposed in [12]. They use the InterPlanetary File System (IPFS) based distributed record storage and management system that maintains access control policies. Although it primarily contributed to record-keeping, it does not show how to connect traditional e-Healthcare. Similarly, A tree-based integrity management method was proposed for privacy and data storage security challenges in [13]. They also focused on scalability and efficient data processing mechanism. Many more articles focus on solving the security and privacy of EHR and scalability challenges. However, it is almost impossible to replace all centralized architecture with a Blockchain system. The optimum solution for the challenges is expecting an interoperability framework. Some contributions [14], [15] focuses interoperability of EHSs, but they consider only traditional EHS, which suffers many challenges, as we discussed in the introduction section. Many challenges might be solved if a Traditional EHS (TEHS) is replaced with a BC-based system. However, replacing the entire system with a BC-based system is next to impossible. Although we still didn't find any raw contributions that directly focused on the solutions, some contributions proposed possible solutions partially. For example, Catena [16] is such a research project that has done some credible work to support distributed immutable relational databases for Blockchain. The concept can also be utilized for the interoperability of EHS. Some available tools (i.e., Apache Sqoop [17] and DataX [18]) and work in [19] show generic conversion mechanisms. However, their integration with a BC-based EHS is an open challenge. Most of the related research contributions considered security and privacy issues of EHS, but none presented a comprehensive migration solution. More specifically, to the best of our knowledge, no work addresses the synchronization of RDB in conventional EHS with the file-based system of Blockchain technology. Contrary, this research shows an optimum solution to bridge every TEHS with the BC network. Ultimately it forms a bridge network that allows exchanging TEHS data. It also offers a real-time solution of exchanging archive data of TEHS with the BC system. The contribution focuses on novel solutions to overcome the scalability of block size to store heavy medical images.

Inter-blockchains interoperability is also a crucial research issue that has been focused on in [20]. The authors discussed cross-chain interoperability for heterogeneous blockchain network communication and proposed a decentralized application-based solution. The solution is suitable for interoperability among existing Blockchain solutions. Contrary, we aim to ensure interoperability between a BEHS and TEHS. Likewise, the authors [21] proposed a security and privacy-preserving scheme for EHS where their contribution goal was data transmission security. They considered an autonomous encryption-decryption mechanism and applied swarm exchange techniques to secure EHR transmission. However, they ensured secure communication using an encryption scheme. Our goal is smooth synchronization and secure communication for interoperability between two technically different EHS.

III. PROPOSED ARCHITECTURE

A. OVERVIEW

The proposed EHS comprises three subsystems, as shown in Figure 1. The *Blockchain Network* presents back-end data services using business Blockchain (i.e., private Blockchain). The prime objective is to integrate typical e-Healthcares and build solid collaborative data-sharing services among healthcare. This service is essential for e-governance, controlling epidemic disasters, insurance, etc. The *Typical e-Health System* (TEHS) depicts a centralized server-based independent health services provider. Such a centralized server may use a relational database to store health records. Many TEHSs are connected with the BC network using a gateway as Figure 1 *Gateway* used in every TEHS bridges the TEHS server and Blockchain peer.

B. BLOCKCHAIN NETWORK

The Blockchain Network(BN) forms with more than three peers with a distributed immutable ledger. The network represents a private business Blockchain architecture that will interlink each existing TEHS. Beyond TEHS, BN also allows access to other enterprises (i.e., insurance companies, government agencies, etc.) and remote users (i.e., physicians, patient attendants, etc.). The network details are presented below.

1) CERTIFICATE AUTHORITY (CA^{BN})

Certificate Authority (CA) provides unique credentials for every element utilized in the ecosystem. All devices are necessarily registered with CA, where CA is the sole authority to generate different certificates and signatures for components and users. These components include users (i.e., physicians, patients, patient attendants, medical staff, administrators, or anyone who requires access), orderers, devices, channels, and peer nodes. In this contribution, we have used two CA, such as one CA for Blockchain users

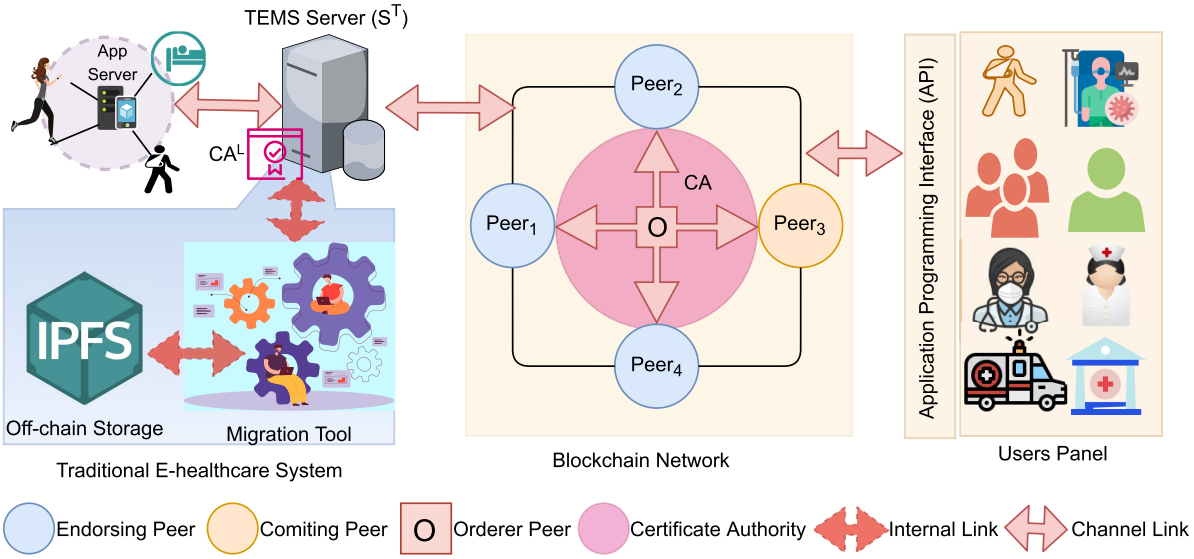


FIGURE 1. Blockchain based national healthcare Infrastructure.

CA^{BN} and CA^{L} for local TEHS. CA^{BN} is responsible for creating credentials for only BC users or components. It is used to ensure proof of identity for secure transactions. The application allows signup transactions that execute in CA^{BC} and are finally approved by the admin.

2) PEER NODES(P)

At least three peer nodes are used to form the Blockchain network. Usually odd number of nodes ensures the consensus of the network. Any registered user (u_i) execute the transaction (Tx_i) to a peer ($P_i \in P$). A leader peer (randomly selected within the active peers) conducts a consensus process and endorse the transaction by signing it. Here, $P^e \subseteq P$ are the endorsing peers who verify the terms and conditions properly justify chain codes pre-installed in every P^e.

3) CHAINCODE (C)

Chaincode is the installed smart contract code in the peer node, a programming script for defining agreement terms between two parties. In this project, the chaincode defines access control and data sharing policies between different participants, including devices. For example, a patient restricts their particular medical record for insurance purposes; no peers will allow the transaction requested by insurance companies. As a result, the transaction denies due to the restriction of the chaincode agreement. Whether chaincode writing and maintenance are costly, it is challenging to change them frequently. So, a generic chaincode policy has been used for such frequent changes cases—for example, chaincode for each patient and physician, which is changeable for every physician change.

4) ORDERER NODES

Every executed, peer nodes endorsed transactions are collected by orderer nodes which provides ordering services in the BN. It is mainly responsible for forming blocks, including all completed transactions for a particular moment. Transactions Tx^D are approved by P^e through a consensus process that is collected by the orderer to be committed in a block B_i. Orderer verifies all valid signatures of endorsing peers and sign itself to finalize the block. Consequently, the ledger is being updated through adding B_i with previously generated chain B_{i-1}, and the process continues in all peers, including committing peers.

5) LEDGER(L)

The ledger comprises all transaction records as a form of block. The blocks are stored as a sequential chain recognized as a block of chain or blockchain. It is a tamper-proof, immutable, and un-forkable encrypted record that is ensured through the chaincode invocations.

C. TRADITIONAL E-HEALTH SYSTEM (TEHS)

Traditional EHSs are typically controlled through a centralized server S^T. Every user and device (i.e., Pt, Ph, SD, etc.) generated data is stored in the server. In addition, TEHS consists of some integrated components, a Migration tool, and a newly added local Certificate Authority (CA^L). Figure 1 presents the typical components of TEHS.

1) INTEGRATED COMPONENTS

- **End-users:** Every participant who generates input for S^T is considered as a user for the particular TEHS. It is assumed that all existing users are already registered with S^T. For bridging with the BC system, users are required to be registered through CA^{BC}. The system administrator also can migrate all existing registration credentials with CA^{BC} using other logical processes for uninterruptible services of users.
- **Application Server (S^a):** Wearable IoT devices are used for continuous record monitoring are resource constraints mostly. These device collected data are primarily processed through a gateway device using an application. For example, smartwatches are monitored

through a smartphone application. These applications are controlled through a remote server (S^a). Every application server is restricted for the specific service, which might be shared or interact with other servers based upon service agreement; for example, S^T can receive data from S^a if compatible APIs are available.

- **TEHS Server (S^T):** Every typical server provides complete control of TEHS, including data storage. Typically, such a server uses Relational Database Management System (RDBMS) services such as Oracle, Mysql, SQL Server, etc. However, in this research, the server does not limit to RDB. Beyond RDB, it provides desktop back-end services that migrate SQL response data to NoSQL format. This back-end service works as SQL-to/from NoSQL Data Converter (SNDC). The S^T also works as off-chain storage of heavy-medical images.

2) MIGRATION TOOL (SNDC)

Bridging TEHS and BC peers is the only way to bring all TEHS into a common umbrella. This can be implemented in two ways such as 1) direct conversion of existing all SQL data to NoSQL ledger at a time, and 2) run-time conversion. The first one is almost impossible due to the real-time data access property of BC technology. Secondly, a logical tool, SNDC can transform SQL-to/from-NoSQL Data Conversion. The SNDC tool works in cooperation with S^T and executes any kind of query from/to *NoSQL* supported BC network. It parses SQL responses to JASON text and consecutively forwards them to BC peers for execution (details in section IV).

3) LOCAL CERTIFICATE AUTHORITY (CA^L)

Typically TEHS registers users (i.e., patients, physicians, IoMT, etc.) through a signup process and stores the users' details in an RDB. In this research, a CA^L is a newly added component with the existing healthcare system. The CA^L is responsible for creating every credential (i.e., certificates, keys, etc.) for TEHS. It is assumed all registered users' credentials in a TEHS are adopted with CA^L at deployment. Any newly added user can be part of the TEHS being a member of CA^L. It should be cleared that $\forall U^T \nexists U^{BC}$, hence, $\forall U^{CA^L} \nexists U^{CA}$.

IV. TECHNICAL DETAILS OF MIGRATION

This section presents transaction processing and technical details of transactions exchanges between centralized TEHS and BC networks. It is assumed that a Blockchain network ensures system security, such as transaction security access control as its nature. Whether transactions are originated in an individual TEHS but executed, processed, and finally stored in a Blockchain system, the in-built security mechanism of the BC system ensures the security of overall EHS. The overall architecture faces two kinds of transactions such as 1) Transaction for Data (Tx^D), used for sending/retrieving the actual payload; 2) Transaction for Query (Tx^Q), responsible for retrieving data from BC ledger/ S^T/Off-chain storage.

TABLE 1. List of symbols.

Symbols	Description
$u_i \in U$	Any kind of User ($SD_i, Pt_i, Ph_i \in U$)
$P_i \in P$	A Peer in BC network
P^e	Set of endorsing peers
SD_i	IoMT service device
Pt_i	Patient
Ph_i	Physician
CA^{BC}	Certificate Auth. of BC network
CA^T	Certificate Auth. of TEHS
S^a	Application Server
S^T	Traditional e-health server
S^s	Off-chain storage server
C_i	A smart contract
Tx_i	A single transaction
$Tx_i^{U_i}$	A transaction from user i
$\acute{Tx}$	Transaction with heavy file
B_i	A block in ledger
sk	secret key
pk	public key
sck	secret compound key
σ	Select operation
π	Attributes
ρ	Rename (table name)
$\bowtie$	Join operation

A. DATA STRUCTURE

Overall system uses a hybrid data structure such as table-based RDB for typical EHS and Blocks of the chain (ledger-based file structure). We elaborate on the features of both for clarity of understanding.

1) BLOCKCHAIN

It is like a typical blockchain structure with minor changes that can be viewed as a linked list where a block B_i is linked with B_{i-1} through a hash value. It can store the pointer of TEHS file storage or actual payload. The core structure and elements of a transaction in a block are explained below.

Block Header: Comprises block sequence in integer Nonce (N^{B_i}), Hash (Tx) whereTx are all existing transactions in the block, and immediate previous block Hash ($N^{B_{i-1}}$).

Transactions/Block Data: Contains a list of all transactions (i.e. Tx_1, Tx_2, ...), where each contains the following:

- *Header* contains metadata such as chaincode name, version, etc.
- *Signature* is a cryptographic signature of issuer.
- *Transaction Proposal* is the payload deployed by issuer.
- *Response* is a Read-Write set (RW-set) that caries approval of proposals by the endorsers.
- *Endorsement* is the collection of signatures of endorsing peers.

Meta Data: It contains all information about the whole block.

2) RELATIONAL DATABASE

It is a typical tabular structure where a transaction is collaboratively interlinked with Blockchain. The records are usually inserted, extracted, or updated using SQL.

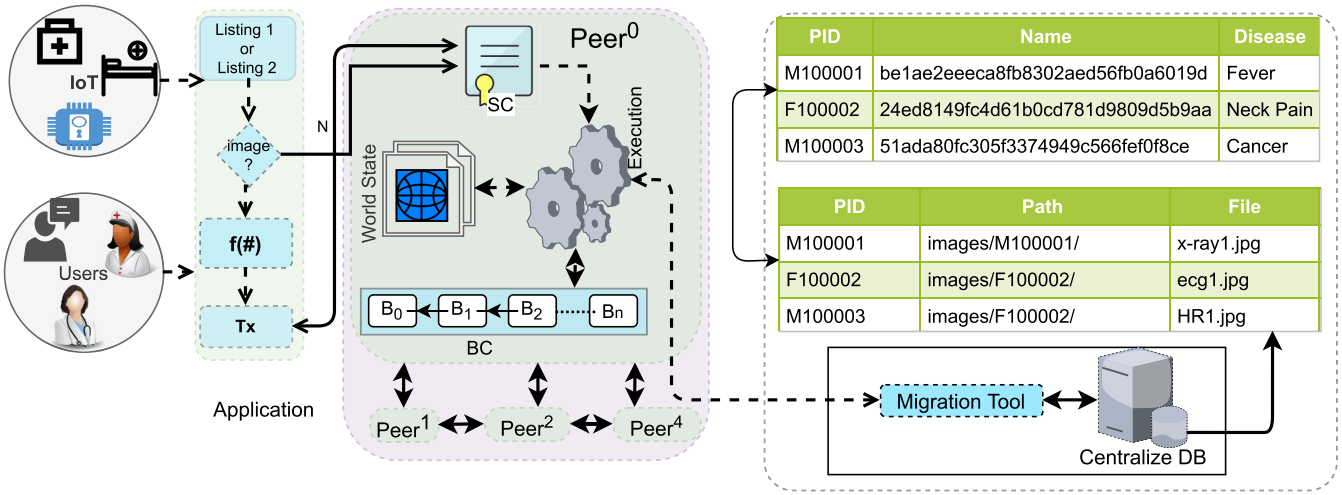


FIGURE 2. Work flow of transactions.

B. DATA TRANSACTION (Tx^D)

Generally, patients are registered members of any TEHS. A patient (Pt^i) initiates transactions under respective TEHS using the application. The application-generated transactions are initially executed in S^T where it is categorized in two types such 1) transaction data contains only string and is executable to BC peer (Tx^D); 2) transaction data contains medical images ($\acute{Tx}^D$).

1) STRING TRANSACTION (Tx_i^D)

Blockchain transactions are executed through the channel where channel members are defined with a smart contract agreement. The application prepares the transaction proposal and executes `invoke( )` function through a channel with all parameters, as given in Listing 1. It is forwarded to endorsing peers P^e, where $P^e \subseteq P$. Transaction proposal forms with users' credentials, chaincode, source, destination addresses, etc. The actual data payload is part of `arg[ ]`, as shown on line 6. P^e verifies the chaincode conditions signatures and signs them with positive consent. Signed transactions are forwarded to the application; consecutively, the proposal is forwarded to the orderer for fitting it into a block. The newly generated block is added with the last block linked with Blockchain as a ledger of peers.

2) TRANSACTION WITH HEAVY DATA ($\acute{Tx}^D$)

It is a widespread phenomenon that transactions contain medical reports such as (x-ray, CT-Scan, ECG, etc.). The payload for such a transaction is too heavy and not executable in the Blockchain network due to block size constraints of BC (e.g., the maximum transaction size of Bitcoin is 1MB [22]). The challenging issues have been resolved using off-chain storage [23]. The client application receives $\acute{Tx}^D$ and generates a hash of heavy files containing reports. The file hashes and file pointer that has a real-time location of

```
1invoke() {
2 let tx_id = this.connection.newTx_id();
3 let TxData = {
4  chaincodeId: 'PatientVsServiceProviders',
5  fcn: 'upload_msg()',
6  args: [],
7  txId: tx_id,
8  chainId: 'chainid',
9 };
10 return this.connection.submitTx(TxData);
11}
```

LISTING 1. Transaction payload function.labelinvoke

```
1query() {
2let QryData = {
3chaincodeId: 'PatientVsServiceProviders',
4fcn: 'Search',
5args: ['']
6};
7return this.connection.query(QryData);
8
```

LISTING 2. Transaction query function.

medical images are executed with other text as a regular transaction, and images are stored in off-chain storage [24].

C. QUERY TRANSACTION

In a TEHS integrated BC system, a query is a more complex task than say. Generally, medical history is stored in a S^T, but the query is executed in the BC system. Therefore previously stored data can be retrieved after the successful migration of RDB data to NoSQL. Likewise, if the query response is related to medical images, it should be retrieved from off-chain storage. A query Tx_i^Q execute Listing 2 in P_i. Application parse the query Tx_i^Q responses, if it contains hash and pointer of files, then rerun the nested query to retrieve the images from off-chain storage.

D. DATA MIGRATION

Data migration performs a real-time synchronization, smooth and seamless exchange of transactions between Blockchain and TEHS. Autonomous transformation is extremely

TABLE 2. Transaction $\acute{Tx}_i^D$ in RDB.

id	*pointer*	*path*	*file*
23ad	123fec	a54ced4c	img1.jpeg, img2.jpeg

complex as the structure of RDB, and BC ledger is quite different. Therefore, it is impossible to exchange/store data directly, but it is essential for synchronization. Therefore, a proper conversion methodology is required for adopting both data structures. The overall conversion execution flow is depicted in Figure 2. The following sections present the technical details of conversion processes.

1) MIGRATION PRINCIPLE

As discussed in earlier sections, the client application categorized the transactions where they will be executed based upon transaction nature; for example, BC ledger, off-chain storage, RDB. Although BC ledger is a document file, it contains nested sets of key-value pairs. On the other hand, RDB is a relational key-based structure. Although both structures are very different, RDB and BC have a commonality of a *key*. Based on this 'key' triples mechanism [25] is used to solve the migration challenges.

Basically, *Triple* forms with three string as word such as S, P, O which express the Subject, Predicate, and Object respectively. Here, *Subject* links the multiple triples, *Object* carries a constant of subject, and *Predicate* establish a relationship between subject and object. It can be written as -

$$\{\mathtt{S}, \mathtt{P}, \mathtt{O}\} \leftrightarrow \{\mathtt{id}, \mathtt{key}, \mathtt{value}\}.$$

We can present an example for a better understanding of the principle. We can assume a heartbeat rate measuring IoMT device measures the data 72 per minute. In the triple format it can be written

$$\{\mathtt{id}, \mathtt{key}, \mathtt{value}\} \rightarrow \{\mathtt{IoMT1}, \mathtt{HR}, 72\}.$$

A real-life example can be presented for more clarity. A transaction proposal with two medical images in JSON format:

$$\begin{aligned}\acute{Tx}_i^D = \{&\text{id:}\mathtt{23ad},\ \text{path:}\mathtt{a54ced4c},\ \text{pointer:}\mathtt{123fec},\\ &\text{file:}[\mathtt{img1.jpg}, \mathtt{img2.jpg}]\}\end{aligned} \tag{1}$$

BC supported transaction proposal (as shown in Eq. 1) can be written in RDB format as shown in Table 2.

The challenges is autonomous transformation. It can be achieved using Triples which has been presented in Table 3.

Table 1 shows the generic symbols for syntax of relational algebra. The relational algebra shown in (2) depicts the migration of Table 2 into triple (Table 3).

$$\begin{aligned}\acute{Tx}_i^D = {} & \rho_{T^s(\text{id, pointer, path, file})}(\pi_{\upsilon_i,\upsilon_p,\upsilon_{pa},\upsilon_f} \\ & \times (\sigma_{k_i=\text{id}\wedge k_p=\text{pointer}\wedge k_{pa}=\text{path}\wedge k_f=\text{file}} \\ & \times (\rho_{T^s_{\text{id}}(i,k_i,\upsilon_i)}(T^{tr}) \bowtie \rho_{T^s_{\text{pointer}}(i,k_p,\upsilon_p)}\end{aligned}$$

TABLE 3. Triple representation of transaction $\acute{Tx}_i^D$.

id	*key*	*value*
i_1	id	23ad
i_1	path	a54ced4c
i_1	pointer	4afc32bc
i_1	file	i_2
i_2	0	img1.jpeg
i_2	1	img2.jpeg

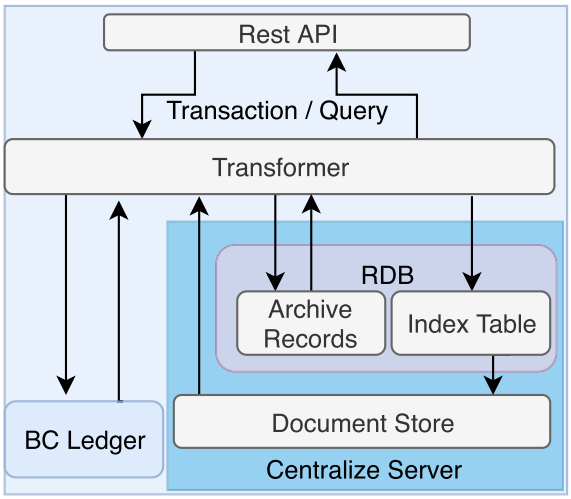


FIGURE 3. Transaction/query transformation flow.

$$(T^{tr}) \bowtie \rho_{T^s_{\text{path}}(i,k_{pa},\upsilon_{pa})}(T^{tr}) \bowtie \rho_{T^s_{\text{file}}(i,k_f,\upsilon_f)}(T^{tr}))))\} \tag{2}$$

The equation presents a queries optimization process for searching a triple value from a relational model using relational algebra where the same id offers multiple key values. Overall optimization modules work based upon a single *key* which is also used in the BC ledger.

Overall synchronization steps are shown in Figure 3. The users forward SQL/NoSQL syntax through API to the migration tool. Consecutively, it forwards to the BC ledger or the centralized server after completing the transformation process. It may require access to the off-chain storage (S^T) for retrieving the archive medical files of patients.

2) MIGRATION OF TRANSACTION BC ↔ RDB

Migration means data transformation from RDB to/from BC ledger, discussed in previous sections—challenging task to retrieve medical images in response to a query from a BC system. Heavy medical images are stored in off-chain storage, but the images location pointer is maintained in an RDB which holds S^T. Both RDB and ledger data can be presented in a triple format as an intermediate process. It is almost impossible to use triple data directly in a production environment. For example, the Eq. 3 retrieves all medical images where $id = i_2$ and create a new table named T^p from triples shown in Table 3.

$$\rho_{T^p(\text{image})}(\pi_{\text{value}}(\sigma_{\text{key}=i_2}(T^{tr}))) \tag{3}$$

It can be written as a set of key-value pairs, such as $\{k_1 : v_1,\ k_2 : v_2, \ldots,\ k_n : v_n\}$. Likewise, every value might have a nested key-value pair in the BC ledger as shown in Eq. 4.

$$\{k_i : [v_i^1, v_i^2, \ldots, v_i^n]\} \equiv \{k_i : \{0 : v_i^1,\ 1 : v_i^2, \ldots,\ n-1 : v_i^n\}\} \quad (4)$$

Here, each value v is represented as an *integer Key*: *value* within the value part of k_i.

It can be more complex if nested key-value pairs are required to be transformed into triples. Explicit presentation example presented in Eq. (5) nested structure of (1) can make clear about the process. Equation 6 shows a function δ_i for transforming a NoSQL transaction (Tx_i^{ns}) to a key-value pair where Γ_i shown in (7) for a complete set of key-value pairs. Finally, (9) shows the complete conversion steps.

$$Tx^{ns} = \{\text{id:}\texttt{23ad},\ \text{path:}\texttt{a54ced4c},\ \text{pointer:}\texttt{123fec},\ \text{file:}\{0 : \texttt{img1.jpg},\ 1 : \texttt{img2.jpg}\}\} \quad (5)$$

$$\delta_i(Tx_i^{ns}) = \{(i, Tx_k^{ns}, T_v^{ns}), (i, Tx_k^{ns}, j) \cup \Gamma_j(Tx_v^{ns})\} \quad (6)$$

$$\Gamma_i(S) = \bigcup_{Tx^{ns} \in S} \delta_i(Tx^{ns}) \quad (7)$$

Hence, we can write

$$\begin{aligned}
\Gamma_{i_1} &= \bigcup_{Tx^{ns} \in S} \delta_{i_1}(Tx^{ns}) \\
&= \{\text{id:}\texttt{23ad},\ \text{path:}\texttt{a54ced4c},\ \text{pointer:}\texttt{123fec},\ \text{file:}\{0 : \texttt{img1.jpg},\ 1 : \texttt{img2.jpg}\}\} \quad (8) \\
\Gamma_{i_1} &= \bigcup_{Tx^{ns} \in S} \delta_{i_1}(Tx^{ns}) \\
&= \{(i_1, \text{id}, \texttt{23ad}), (i_1, \text{path}, \texttt{a54ced4c}), (i_1, \text{pointer}, \texttt{123fec}) \cup \{(i_1, \text{file}, i_2)\} \\
&\quad \cup \delta_{i_2}(0 : \texttt{img1.jpg}, 1 : \texttt{img2.jpg})\} \\
&= \{(i_1, \text{id}, \texttt{23ad}), (i_1, \text{path}, \texttt{a54ced4c}), (i_1, \text{pointer}, \texttt{123fec}), (i_1, \text{file}, i_2)\} \bigcup_{\acute{T}^{ns} \in \{0:\texttt{img1.jpg}, 1:\texttt{img2.jpg}\}} \delta_{i_2}(\acute{T}^{ns})\} \\
&= \{(i_1, \text{id}, \texttt{23ad}), (i_1, \text{path}, \texttt{a54ced4c}), (i_1, \text{pointer}, \texttt{123fec}), (i_1, \text{file}, i_2)\} \cup \delta_{i_2}(0 : \texttt{img1.jpg}) \cup \delta_{i_2}(1 : \texttt{img2.jpg}) \\
&= \{(i_1, \text{id}, \texttt{23ad}), (i_1, \text{path}, \texttt{a54ced4c}), (i_1, \text{pointer}, \texttt{123fec}), (i_1, \text{file}, i_2), (i_2, 0, \texttt{img1.jpg}), (i_2, 1, \texttt{img2.jpg})\} \quad (9)
\end{aligned}$$

3) QUERY RESPONSES

As discussed earlier, BC ledger maintains world state and state databases. Hence, queries are executed from NoSQL supported world-state (e.g., CouchDB). For query response, user defined conditions are used as a slectors DB, which also is used in JSON object. Both purposes the selector plays the role of *key*, consicuently the key retrieves the associated values from RDB or file DB. For example, according to Table 3 and (9),

$$\{\text{selector} : \{\text{“id}'' : \text{“}\texttt{23ad}'', \text{“file}'' : \text{“}i_2''\}\}$$

an *id* field containing *23ad*, and file i_2 that matches in whole database document. The key matching works as a key to retrieve the respective images as values. Overall, the query process follows the Algorithm 3, which executes at the peer in response to the query from API. In the algorithm, users' credentials are verified in line 1, and Line 2 separate the main key. Searching is done through Line 3–8, while lines 5–8 retrieve images from off-chain storage. If the key doesn't exist in the ledger, it is considered to be available in RDB, which is executed by lines 10–14.

Algorithm 1: Query Processing

Input : Query (Tx^Q, U_{id}, U_{sign}, C^{ver})
Output: `Arg[]`

1 **if** U_{id}, U_{sign}, C^{ver} **then**
2 set $key^{Tx^Q} \leftarrow$ `parse`(Tx^Q)
3 **if** $key^{Tx^Q} \exists\ L$ **then**
4 $R^{Tx^Q} \leftarrow$ `Query`($key \exists L$)
5 **if** ($img.Hash \leftarrow$ `parse`(R^Q)) **then**
6 set
7 $img \leftarrow$ `Query`(`Off − chain`, $\forall_{key^{Tx^Q}}$)
8 set $R^{Tx^Q} + \leftarrow$ `merge`(R^{Tx^Q}, img)
9 **end**
10 **else**
11 **while** U_{id} **do**
12 set $R^{Tx^Q} \leftarrow$ `Query`($RDB(key^{Tx^Q})$)
13 **end**
14 Set $R^{Tx^Q} \leftarrow$ `JSON.Conv`($Pt_{id} + value$)
15 **end**
16 **else**
17 $R^{Tx^Q} \leftarrow$ `null`
18 **end**
19 return R^{Tx^Q}

E. MEMORY CONSUMPTION DISCUSSION

While many TEMS are integrated with the BC system, it produces a big-data. The data are also varied in nature and have to migrate in BC-ledger. This section estimates the memory consumption and how the solution mitigates excessive memory usage. BC stores payload data along with metadata, and every block is copied to all peers. As every peer maintains the same copy of the ledger, the memory occupies by peers increases geometrically. Heavy medical images can not be part of BC transactions due to block size limitations and huge bandwidth to process the consensus. Moreover, it will affect the TPS (scalability) because the consensus mechanism must approve every transaction. We have introduced the following equations to

measure the block weights in this solution. For standard formulation, we can assume that a number of users U^n invoke T^n transactions to create a block B_x in a block session (e.g., 1sec) where P_x^e endorsing peers are responsible. Equation 10 estimates the block weight in real-time where ω denotes weight.

$$\omega^{B_x} = \sum_{i=1}^{n} \omega^{T_i} + \omega(B_j^{\text{header}} + B_j^{\text{metadata}}) \tag{10}$$

where,

$$\omega^{B_j^{\text{header}}} = \omega(B_{j-1}^{\text{Hash}} + Hash(\forall_{T_n})), \quad \text{and}$$
$$\omega^{T_i} = \omega(T_i^{\text{header}} + U_i^{\text{sign}} + T_i^{\text{data}} + \sum_{j=1}^{n}(P_j^e(\text{resp}) + P_j^e(\text{sign}))).$$

The above equations calculate only BC transaction weight, excluding medical images. As medical images are stored in off-chain storage, it doesn't affect ledger size.

V. EVALUATION AND ANALYSIS

The proposed system has been implemented using Hyperledger Fabric (v2.0) platform in a docker container platform. For implementation, two physical systems have been used, such as *i*) Intel i5 3 GHz processor with 8 GB of 1600 MHz DDR3 RAM, and *ii*) Intel i7 2.7 GHz processor with 16 GB of 1600 MHz DDR3 RAM. Overall the prototype is implemented using four peers where node-red based application is used to generate transactions. In addition, TEMS has been implemented in MySQL Database, and N1QL [26] is used as a JSON document model for developing queries or transactions.

A. GENERAL OBSERVATIONS

The bridging platform implementation reflects some general observation which has been received using docker log analysis. For user registration in CA^{BC} requires between 3–20ms, which is not too high. Moreover, one user registered once with the system. Transaction completion time is a significant issue in the business blockchain, affecting scalability. In the proposed system, the average transaction completion time is $\approx 3s - 180s$. It takes more time than usual because of transaction preparation in the application compared to consensus formation or creation of the block. Hence, the performance of the blockchain network is not affected by it.

B. TRANSACTION DATA PROCESSING

To evaluate the system performance, we have compared complete transaction Tx^D time with centralized, traditional EHS where every technical decision comes from a single server. It is calculated from transaction origination to the final commit of a block. Figure 4 presents the transaction execution times in a four endorsing peers P^e network. A single transaction in a BC integrated EHS takes $\approx 20ms$, while a centralized system requires $\approx 12ms$. In a centralized system, 5 concurrent transactions require $\approx 65ms$ while BC-based system requires $\approx 85ms$. Finally, for 50 transactions, the centralized model requires ≈ 500 *ms* while the BC-based system requires close to 900*ms*.

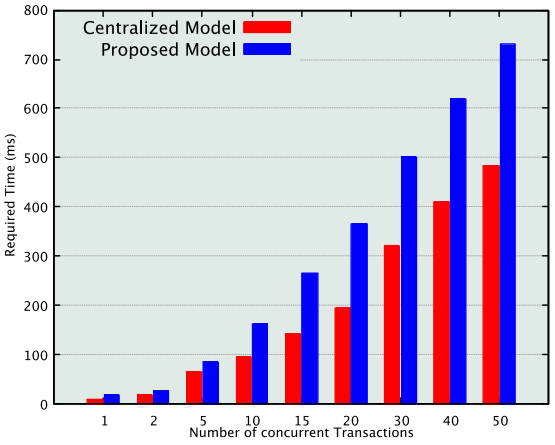


FIGURE 4. Transaction time for TEHS and BC-EHS.

The overall graph summarizes that the number of transactions is proportional to the execution time. Although the BC-based system takes a little more time than the conventional system, the increase is not significant, acceptable. More time can be overlooked in terms of security, interoperability, and openness of the systems. Moreover, transaction execution time in a blockchain system depends on many parameter settings, and it can be overcome with an efficient and lightweight consensus algorithm.

C. QUERY PROCESSING EVALUATION

This experiment evaluates the time of information retrieval from the system. Figure 5 shows the overall impact on query processing. It presents the query response time is significantly less than the traditional system. It is noticed that the processing time is increasing for a single peer with concurrent queries. Although BC and the centralized system's times are very close ($\approx 4ms$), the difference significantly increases with increment in queries. The centralized model requires 25*ms*, while a single peer BC requires $\approx 70ms$ for 50 concurrent queries. Contrary, required times sharply fall in the multi-peer network. For example, approximately 8ms to 6ms is required for 15 to 20 concurrent queries. As all peers hold ledger, queries are distributed in peers.

D. LEDGER SCALABILITY

As discussed in previous sections, transactions containing medical images/documents processing are complex. EHS transaction weight mostly depends on how many images carry each transaction and how much weight each. For a better estimate, if a single medical image's weight varies 10KB to 5MB and one transaction carries 10 such images, then total weight can be estimated easily. Moreover, it will be increased more due to BC metadata. We have used images not more than 1MB in size. Figure 6 presents required memory

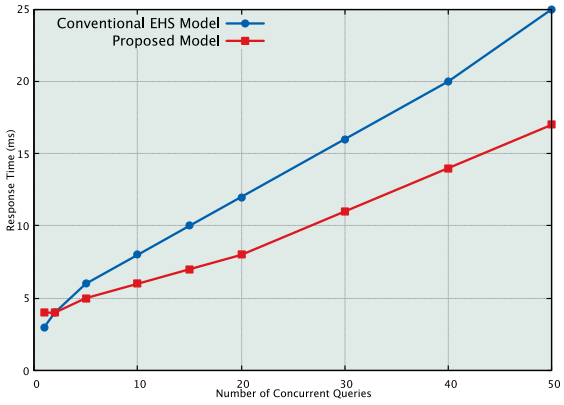


FIGURE 5. Required time for queries.

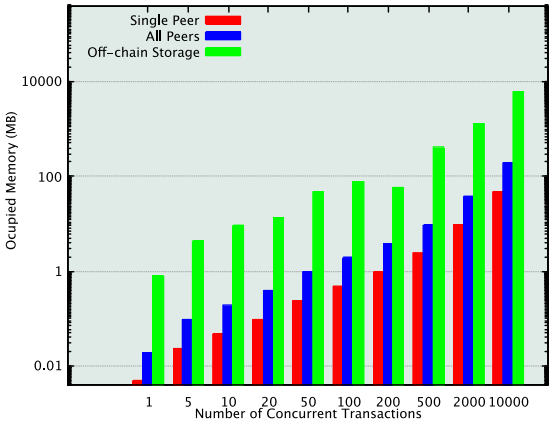


FIGURE 6. Blockchain ledger and off-chain memory expansion.

in different scenarios such as for a single ledger, all four ledgers, and the off-chain storage. At the same time, the bulk of the heavy data is separated from the transaction and stored in off-chain storage. It creates a significant impact on memory that shows the increment of memory occupying is almost linear and constant for a single peer. As four peers maintain identical ledgers, their total memory requirement is also linear. The solution proposed in this work does not affect the efficiency of the blockchain itself. The ledger remains as scalable as a non-heavy data environment such as crypto-currency transactions.

E. SCALABILITY: BANDWIDTH CONSERVATION

The scalability of a Blockchain network is an open question. It is more critical for the E-healthcare sectors. Processing heavy transactions in the consensus stage requires massive bandwidth, affecting TPS. Figure 7 presents a bar graph for required bandwidth using the proposed solution against a generic BC solution for heavy data trades $\acute{T}x_D$. It shows that if the heavy data is part of 1K transactions, then the available bandwidth should be approximately 10 GBps, to achieve the same efficiency (TPS) as our solution, which can work within 60MBps. Hence, the proposed solution reduces the memory requirements and limits the network bandwidth required to create a unified blockchain-based e-health system.

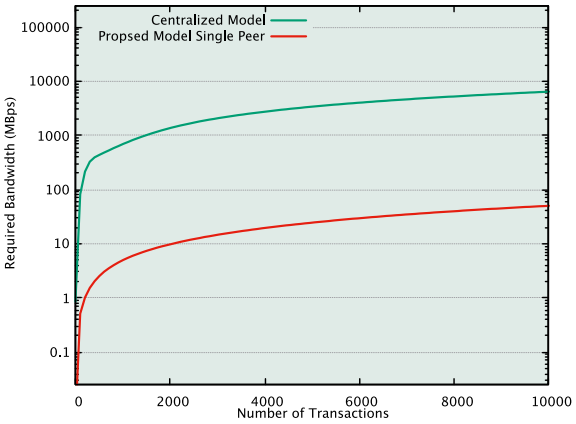


FIGURE 7. Required bandwidth for consensus.

F. CHALLENGES AND LIMITATIONS

It is unavoidable that blockchain is a crucial security technology for health data security. But still, it suffers scalability and ledger optimization challenges. Heavy medical images, continuous transactions, and open interoperability are added to new challenges in the production environment. Although this article shows an intelligent adoption with typical EHS, there are still challenges due to platform dependency of various consensus mechanisms. Although many lightweight consensus algorithms have been proposed in terms of scalability, the more smooth open platform-independent consensus is crucial.

VI. CONCLUSION

Health records are very private but precious for public research development and advancement. The interoperability of healthcare scattered across a country is crucial for ensuring the quality of healthcare services and modern research. This study highlights all the technical details of building a national health data center without compromising security. Our proposed blockchain framework bridges typical e-healthcare. Our proposed Triple methodology solves the synchronization challenges between blockchain and typical EHS. Different algebraic equation proves the effectiveness of the proposal. Moreover, implementation results and discussion demonstrate the effectiveness of the proposed approach. Furthermore, Protocol independent, plug and play consensus supported interoperable blockchain-based EHS system can ahead of the contribution one step more.

REFERENCES

[1] CPNI. *Critical National Infrastructure*. Accessed: Oct. 1, 2021. [Online]. Available: https://www.cpni.gov.U.K./critical-national-infrastructure-0
[2] X. Liu, C. Qian, W. G. Hatcher, H. Xu, W. Liao, and W. Yu, "Secure Internet of Things (IoT)-based smart-world critical infrastructures: Survey, case study and research opportunities," *IEEE Access*, vol. 7, pp. 79523–79544, 2019.
[3] S. Biswas, K. Sharif, F. Li, A. K. Bairagi, Z. Latif, and S. P. Mohanty, "GlobeChain: An interoperable blockchain for global sharing of healthcare data—A COVID-19 perspective," *IEEE Consum. Electron. Mag.*, vol. 10, no. 5, pp. 64–69, Sep. 2021.
[4] A. Rubin, *Hacking Health: Security in Healthcare IT Systems*. San Francisco, CA, USA: USENIX Association, 2016.

[5] S. Biswas, K. Sharif, F. Li, B. Nour, and Y. Wang, "A scalable blockchain framework for secure transactions in IoT," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4650–4659, Jun. 2019.

[6] X. Xiang, M. Wang, and W. Fan, "A permissioned blockchain-based identity management and user authentication scheme for e-health systems," *IEEE Access*, vol. 8, pp. 171771–171783, 2020.

[7] S. Biswas, K. Sharif, F. Li, I. Alam, and S. Mohanty, "DAAC: Digital asset access control in a unified blockchain based E-health system," *IEEE Trans. Big Data*, early access, Nov. 16, 2020, doi: 10.1109/TBDATA.2020.3037914.

[8] P. P. Ray, D. Dash, K. Salah, and N. Kumar, "Blockchain for IoT-based healthcare: Background, consensus, platforms, and use cases," *IEEE Syst. J.*, vol. 15, no. 1, pp. 85–94, Mar. 2021.

[9] K. Zhang and H.-A. Jacobsen, "Towards dependable, scalable, and pervasive distributed ledgers with blockchains," in *Proc. IEEE 38th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2018, pp. 1337–1346.

[10] S. Biswas, K. Sharif, F. Li, S. Maharjan, S. P. Mohanty, and Y. Wang, "PoBT: A lightweight consensus algorithm for scalable IoT business blockchain," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2343–2355, Mar. 2020.

[11] Z. Zheng, J. Pan, and L. Cai, "Lightweight blockchain consensus protocols for vehicular social networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5736–5748, Jun. 2020.

[12] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Blockchain for secure EHRs sharing of mobile cloud based E-health systems," *IEEE Access*, vol. 7, pp. 66792–66806, 2019.

[13] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in *Proc. IEEE 28th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Oct. 2017, pp. 1–5.

[14] F. Barbarito, F. Pinciroli, J. Mason, S. Marceglia, L. Mazzola, and S. Bonacina, "Implementing standards for the interoperability among healthcare providers in the public regionalized healthcare information system of the Lombardy region," *J. Biomed. Informat.*, vol. 45, no. 4, pp. 736–745, Aug. 2012.

[15] W. J. Gordon and C. Catalini, "Blockchain technology for healthcare: Facilitating the transition to patient-driven interoperability," *Comput. Struct. Biotechnol. J.*, vol. 16, pp. 224–230, Oct. 2018.

[16] T. Van Der Vorst, "Catena: SQL on a blockchain," Pixel Spark, Catena Project, CA, USA, Dec. 2017. [Online]. Available: https://github.com/pixelspark/catena

[17] A. Bhardwaj, Vanraj, A. Kumar, Y. Narayan, and P. Kumar, "Big data emerging technologies: A CaseStudy with analyzing Twitter data using apache hive," in *Proc. 2nd Int. Conf. Recent Adv. Eng. Comput. Sci. (RAECS)*, Dec. 2015, pp. 1–6.

[18] (Jun. 2018). *DaTax: Complete Data Ecosystem on Blockchain*. [Online]. Available: https://datax.io/datax_whitepaper_0.5.5.pdf.

[19] G. Zhao, Q. Lin, L. Li, and Z. Li, "Schema conversion model of SQL database to NoSQL," in *Proc. 9th Int. Conf. P2P, Parallel, Grid, Cloud Internet Comput.*, Nov. 2014, pp. 355–362.

[20] M. Madine, K. Salah, R. Jayaraman, Y. Al-Hammadi, J. Arshad, and I. Yaqoob, "AppxChain: Application-level interoperability for blockchain networks," *IEEE Access*, vol. 9, pp. 87777–87791, 2021.

[21] P. P. Ray, B. Chowhan, N. Kumar, and A. Almogren, "BIoTHR: Electronic health record servicing scheme in IoT-blockchain ecosystem," *IEEE Internet Things J.*, vol. 8, no. 13, pp. 10857–10872, Jul. 2021.

[22] L. Cocco, A. Pinna, and M. Marchesi, "Banking on blockchain: Costs savings thanks to the blockchain technology," *Future Internet*, vol. 9, no. 3, p. 25, Jun. 2017.

[23] A. Musamih, K. Salah, R. Jayaraman, J. Arshad, M. Debe, Y. Al-Hammadi, and S. Ellahham, "A blockchain-based approach for drug traceability in healthcare supply chain," *IEEE Access*, vol. 9, pp. 9728–9743, 2021.

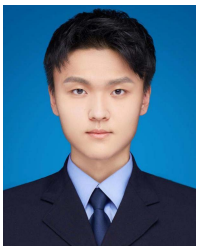
[24] J. Seo and Y. Cho, "Medical image sharing system using hyperledger fabric blockchain," in *Proc. 22nd Int. Conf. Adv. Commun. Technol. (ICACT)*, Feb. 2020, pp. 62–64.

[25] M. Bae, H. Park, G. Lee, J. Eum, and S. Oh, "Scalable RDF triple store using summary of hashed information and bit comparison," in *Proc. IEEE Pacific Rim Conf. Commun., Comput. Signal Process. (PACRIM)*, Aug. 2015, pp. 163–168.

[26] Y. Tian, M. Carey, and I. Maxon, "Benchmarking HOAP for scalable document data management: A first step," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2020, pp. 2833–2842.

YIYING LIU received the bachelor's degree in computer science and technology from the Changsha University of Science and Technology, in 2006, and the master's degree in environment and resources law from the Kunming University of Science and Technology, in 2012. She is currently an Assistant Researcher with the Kunming University of Science and Technology.

GUANGXING SHAN received the bachelor's degree in network engineering from the Xi'an University of Posts and Telecommunications, in 2007, and the master's degree in software engineering from the University of Science and Technology of China, in 2010. He is currently a Senior Engineer with the Wuhan Real Estate Information Center.



YUCHENG LIU is currently pursuing the bachelor's degree with the School of Police Administration, People's Public Security University of China.

ABDULLAH ALGHAMDI received the B.Sc. degree in information systems from Al-Imam University, Saudi Arabia, the M.Sc. degree in networking and systems administration from the Rochester Institute of Technology, Rochester, NY, USA, and the Ph.D. degree in computer and information systems engineering from Tennessee State University, Nashville, TN, USA. He is currently working as an Assistant Professor at the Information Systems Department, Najran University, Najran, Saudi Arabia. He is also a Trustee and an Executive Secretary of Scientific Board. His current research interests include security, privacy, the IoT, interdisciplinary applications, and data analytics.

IQBAL ALAM received the bachelor's degree in business information and communication system from London Metropolitan University, in 2009, and the master's degree in software engineering and the Ph.D. degree in computer science and technology from the Beijing Institute of Technology, in 2016 and 2020, respectively. He is currently a Publication Manager with the Nan Yang Academy of Sciences (NASS).

SUJIT BISWAS (Member, IEEE) received the Ph.D. degree in computer science and technology from the Beijing Institute of Technology, China. He is currently a Research Fellow of blockchain and AI with the University of Surrey, U.K. He is also an Assistant Professor with the Faridpur Engineering College, University of Dhaka, Bangladesh. His research interests include the IoT, blockchain, sensor networks, machine learning, and mobile computing security and privacy. He is a Life Member of the Bangladesh Computer Society (BCS).