IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 31, 2019, accepted May 12, 2019, date of publication May 20, 2019, date of current version July 12, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2917976

EACMS: Emergency Access Control Management System for Personal Health Record Based on Blockchain

AHMED RAZA RAJPUT[1], QIANMU LI[1,2,3], (Member, IEEE), MILAD TALEBY AHVANOOEY[1], (Student Member, IEEE), AND ISMA MASOOD[1]

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
[2]Intelligent Manufacturing Department, Wuyi University, Jiangmen 529020, China
[3]School of Information Engineering, Nanjing Xiaozhuang University, Nanjing 211171, China

Corresponding author: Qianmu Li (qianmu@njust.edu.cn)

This work was supported in part by the Chinese Government Scholarship under Grant 2015GXZW13, in part by the Project of ZTE Cooperation Research under Grant 2016ZTE04-11, in part by the Jiangsu Province Key Research and Development Program: Social Development Project under Grant BE2017739, in part by the Jiangsu Province Key Research and Development Program: Industry Outlook and Common Key Technology Projects under Grant BE2017100, and in part by the 2018 Jiangsu Province Major Technical Research Project "Information Security Simulation System."

ABSTRACT Personal health records (PHRs) are private and vital assets for every patient. There have been introduced many works on various aspects of managing and organizing the PHR so far. However, there is an uncertain remaining issue for the role of PHR in emergencies. In a traditional emergency access system, the patient cannot give consent to emergency staff for accessing his/her PHR. Moreover, there is no secured record management of patient's PHR, which reveals highly confidential personal information, such as what happened, when, and who has access to such information. This paper proposes an emergency access control management system (EACMS) based on permissioned blockchain hyperledger fabric and hyperledger composer. In the proposed system, we defined some rules using the smart contracts for emergency condition and time duration for the emergency access PHR data items that patient can assign some limitations for controlling the PHR permissions. We analyzed the performance of our proposed framework by implementing it through the hyperledger composer based on the response time, privacy, security, and accessibility. The experiments confirm that our framework provides better efficiency compared with the traditional emergency access system.

INDEX TERMS Personal health record, emergency access, access authorization, blockchain, hyperledger fabric, hyperledger composer, privacy & security.

I. INTRODUCTION

Some applications of big data for health care services and medications may be dealt with third parties or the public for surveys and the extraction of the useful report [1], [2]. In some cases, the healthcare data for sensitive patients may be exposed to malicious attacks, and the risks are involved in this process such as tampering and unauthorized access. Therefore, it is an essential issue that must be considered to ensure security and privacy during the system design for sensitive healthcare data [3].

A PHR is a tool for electronically saving a person's lifelong health data. It must grant specific access control for managing, tracing, and participating in their health care data [4]–[6]. PHR carries integrated and complete health care data such as healthcare history, medication procedures, and significant disease as well as allergy information, home-monitored data, family member's history, social records and lifestyle, immunization record, prescribed medications, laboratory experiments, and genetic data [7]–[10]. Secure data sharing is crucial to present sufficient collaborative treatment and care choices for patients [11]–[14]. The information in the PHR should be correct, trustworthy, and complete. In the emergency condition, the medical staff needs some necessary elementary and valuable health information about the patient for

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

increasing the chance to supply appropriate cure to save the patient's life or assuage in perilous conditions. Due to the sensitive and confidentiality system of PHR, access to the PHR is restricting for anyone included her/his physician. In the emergency condition, such pre-defined access control policies become less because there is no policy defined that would allow an emergency medical team to access the patients' health record. Also, the patient is not in a sense and cannot operate the access control for giving his/her PHR [15], [16]. Moreover, the patient's care is critical and his/her health record an essential aspect of the patient's security and privacy. One more significant challenge is in the emergency condition after accessing the PHR; misuse of personal information can be exploited illegally. In the traditional PHR emergency cases, the system not verifies the identification of the object either a people or organization sent the PHR request. In the traditional emergency access system, when the staff has done some activities on the PHR records, and later, the malicious users could try to obtain the patient's PHR information. There is no audit or transaction log where the patient can trace all the accesses of PHR information because, in the emergency condition, the patient cannot participate in the emergency access authorization.

To alleviate these problems and make sure secure access handling in emergency and maintain a secure ledger. So, we introduce a unique framework that leverage the distributed and immutable shared ledger called Blockchain technology. The blockchain is a decentralized architecture that features a distributed immutable ledger in which all transactions are recorded. More generally, blockchain is a secure and decentralized data store of ordered records, including events, called blocks [17]. In the traditional cloud storage system, the puzzle of the single mark of failure can resolve with the approach of decentralized storage system and enjoy many benefits over centralized storage system [18]. De-centralized data is stored on every single peer, and no one can alter it. The hash of data is generated to make mutability impossible. The Blockchain network is fundamentally resilient and has a no-single vulnerability for hackers to exploit. The blockchain system records the information transparent to each node, and it is also apparent to update the data, that's why blockchain can be trusted.

Our system is a hyperledger composer [19] based Blockchain application that alleviates these problems with an architecture that allows our system to be both extensible and scalable: a blockchain based EACMS, off chain data storage and a patient-centered mobile and web interface. To maintain a high application performance that is economically viable, all of the PHR data such as personal biodata, diagnostic images, lab test results, prescriptions, treatment drugs plans, etc. are stored on a secure cloud-based repository, and PHRs URI is stored in our client application. With the use of smart contracts, the PHR data owner (patient) set the condition for an emergency staff or team member (authentic doctor) who can access permissions to any of their PHR data items flexibly and securely with the time limitation. For the patient and his family doctor can easily access the system via a web browser and mobile interface with Rest API in hyperledger composer based application.

TABLE 1. Abbreviations.

Abbreviations	Description
PHR	Personal Health Record(s)
ED	Emergency Doctor
EMT	Emergency Medical Team
SC	Smart Contracts
PCHR	Personally Controlled Health Records
EHR	Electronic Health Records
JSON	JavaScript Object Notation
CA	Certificate Authority
BNA	Business Network Archive
SDK	Software Development Kit
TPF	Transaction Processing Function
HF	Hyperledger Fabric
API	Application Program Interface

The remaining part of the paper is organized as follows. Section II explains the background on the permissionless Blockchain architectures, Bitcoin, and ethereum and the permissioned blockchain architecture, Hyperledger Fabric (HF) and Hyperledger Composer. Section III presents a brief literature review of emergency access control. Section IV gives the proposed system and system design of our model. Section V presents system implementation details. Section VI provides performance and analysis of our framework. Section VII concludes the paper with a summary of research contribution.

II. BACKGROUND

This section reviews background on blockchain technology network, Hyperledger Fabric, and Hyperledger Composer. Table 1 summarizes the definitions of various symbols used in the paper.

A. BLOCKACHIAN TECHNOLOGY NETWORK

A decentralized network known as a peer-to-peer platform is a distributed architecture that allocates its resources to a host of nodes, functioning together to make decisions on behalf of the network. In a decentralized system [20], no centralized authority acts as an agent for all communications; instead, each node is free to perform peer-to-peer functions known as transactions. The blockchain is a decentralized architecture that features a distributed immutable ledger in which all transactions are recorded. More generally, the blockchain is an irrefutable inspired invention by an anonymous person or group of people alias SATOSHI NAKAMOTO. Although blockchain got the popularity in the world of finance through the popular cryptocurrency BITCOIN [21], the blockchain is a distributed shared ledger for recording the records of transactions, and that record cannot be altered or change. The operations are taking place at every moment. In the blockchain, each participant has his own distributed ledger.

Because having of various ledgers is a technique for frauds, errors, and ineptitudes. The aim is to watch a transaction peer-to-peer and reduce those vulnerabilities. The blockchain manages the continually increasing the list of records which are immutable and distributed. All of the cryptocurrencies appropriate what can sufficiently be defined as a public ledger that is improbable to fraudulent. Each participant or node entity has the equivalent accounting ledger as all of the other participants or entity nodes in the system network. This assures a full consensus from all participants or nodes in the analogous currencies blockchain [22], [23]. There is an intention that multiple systems are made on the platform of blockchain technology accomplish the secure distribution of resources between the untrusted parties. The assets exchange among the parties without having to depends on the third party or a middle person, the blockchain technology makes available a vital trust layer for business transactions [24]. The bitcoin blockchain case explains that the blockchain technology network describes a strategy as to how ownership of digital assets can be obtained [25]. In the current era [26], blockchain has extended to produce both inside and outside of the financial systems. The blockchain has promising solutions for different applications in the field of health care domain and medical study. Several forms in the area in healthcare have supported the profits of this new technology in building a protected podium for maintaining and evaluating confidential health care information.

1) PERMISSIONLESS BLOCKCHAIN

In permissionless or public blockchain network system the identities and names of participating members are each pseudonymous or also unknown, and each participating members may add a new block to the ledger. To suffice privacy, security, and extra necessities, permissionless and permissioned chains survive in the blockchain society. A permissionless blockchain such as bitcoin is an open platform [27]. The internet is the best example of the permissionless or public blockchain system. Anybody can create the website with their choice on the internet. Each person or organization can choose to run a node for the blockchain and take part in transaction verification with the mining algorithm. Ethereum [28], [29] is the implementation of a permissionless blockchain system that permits a person to make and execute the code on the ethereum platform. Ariel Ekblaw and Asaph Azaria in the MedRec [30] developed a solution for electronic medical record management which built on the proof of work (POW), Ethereum based network in permissionless or public blockchain network. On the permissionless blockchain frequency-based analysis of the encrypted transaction, data could provide in the network to untrusted parties. The permissionless blockchain system has seen an explosion of governance models covering both industrial architecture and financial decisions. This puzzle fixed by the primary policy of permission blockchain network where only authorized users could see the causal movements.

TABLE 2. Permissionless vs permissioned blockchain.

Permissionless blockchain [33-35]	Permissioned Blockchain[33-35]
✓Anyone can join the network, read, write, and commit ✓Hosted on public servers ✓Anonymous highly resilience ✓Low scalability ✓No one can control it ✓No one can shut it down ✓Can hardly be reverted, Transparency, ✓Creates Marketplaces ✓Bitcoin, Ethereum	✓Only authorized participants can join the network and read, ✓Only the network operator can write and commit ✓Very High Scalability ✓Easy to Implement ✓Easy to Maintain ✓Additional security ✓No need to use cryptocurrency ✓Lower service cost ✓Hyperledger

2) PERMISSIONED BLOCKCHIAN

The permissioned blockchain is built to permits a person, an organization or a group of organizations to transfer information and record transactions efficiently. It adds a layer of privileged to decide who can participate in the network system, with the identity of each participant known to all participants [31]. In the permissioned blockchain network system, the participant does not have a chance to fraud as their identity is exposed to the management server. Also, within the financial sector Quorum, Ripple is the open source and highly swearing in their power but was not formed with healthcare- particular. Such as Quorum split the famous Ethereum the permissionless or public blockchain network or other designs centred on cryptocurrency and attached permissions and additional functionality as required. In the permissioned or private blockchain network, usually, apply an algorithm such as Byzantine Fault Tolerance (BFT) [32]. While in the permissionless blockchain system has used proof of work algorithm. In the distinction, The HF is designed for the secure handling of distributed ledger technology and enable permissioned. Table 2 depicts various features of permissionless and permissioned Blockchain.

B. HYPERLEDGER FABRIC

The HF is an implementation of a permissioned blockchain system [36] and open source blockchain initiative hosted by the Linux Foundation. It is one of the prime permissioned blockchain structures presently. Because it is based on the permissioned blockchain, it allows just concerned stakeholders as participant members and confines anyone to meet the network, temper the ledger, or invoke the transactions. The HF Network consists of several kinds of nodes, client nodes, peer nodes, and ordering nodes relating to the various organization. Every node's identity on the HF network which is presented by a membership service provider (MSP) [37], typically correlated with an organization. All nodes in the HF network have distinctness to the identities of all parties and approve them. The MSP issues the enrolment and transaction certificates to the client. Also it provides the opportunity to utilize a consensus mechanism that is much lighter

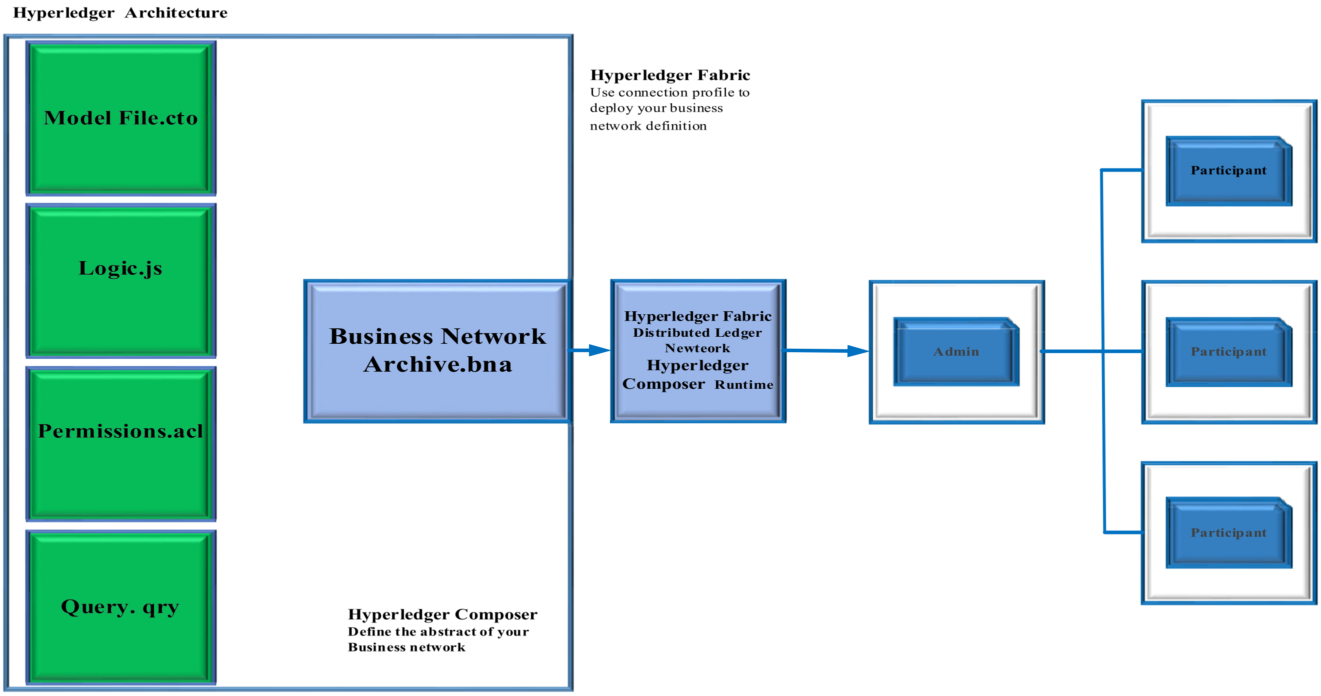
Hyperledger Architecture
Model File.cto
Logic.js
Permissions.acl
Query. qry
Business Network Archive.bna
Hyperledger Composer
Define the abstract of your Business network
Hyperledger Fabric
Use connection profile to deploy your business network definition
Hyperledger Fabric Distributed Ledger Newteork Hyperledger Composer Runtime
Admin
Participant
Participant
Participant

FIGURE 1. Hyperledger composer architecture.

computationally than the Proof of Work mechanism. HF also adds the capability to build trusted subnetworks, called channels, that can establish shared ledgers with a defined set of nodes and transact to the exclusion of the rest of the blockchain. The fabric has smart contract functionality, Chaincode, which enables participants to execute complex transactions as per defined permissions.

C. HYPERLEDGER COMPOSER

Hyperledger Composer is an open developing toolset for creating blockchain applications. The Hyperledger Composer supports the Hyperledger Fabric infrastructure and runtime and allows for quicker business network modeling, application implementation, and integration with existing systems [19].

The business network definition is exported as an archive (. bna file) when it is ready to be deployed. The definition of the network is made up of four principal files: model, script, access control, and query (Figure 1).

- *The model file* is responsible for outlining the structure of the network. It has three main components: assets, participants, and transactions. Assets are often the variables stored in the network. Participants are the nodes of the network and can interact with assets and other participants through transactions. Transactions are the functions of the network and are invoked to update the network (e.g., transferring an asset).
- *The script file* defines various transaction functions in the network. It is written in JavaScript and handles the transaction logic, including which types of participants interact (different categories of participants have different levels of access in the network) and which kinds of assets are transferred.
- *The access control file* delineates the specific scopes of access users have in the business network. In this file the role of the user (participant) is described, determining their role in creating, reading, updating, or deleting elements of the network.
- *The query file* defines the structure and function of queries from this network. Queries can be set to extrapolate transactions from the historian, which is a ledger of all past transactions in the system.

Once the network is defined, it can be exported as an archive, downloaded, and run on another machine. A network card is used to connect to the system. Network cards can take the form of a participant type or an admin. Participant cards generally have a more controlled scope of access in the network, while the admin can perform more high-clearance functions such as adding new participants or deleting participants. This card type defines the node that uses the card to connect to the network and, thus, outlines what kind of role the node plays.

III. RELATED WORKS

A. EMERGENCY ACCESS CONTROL FOR PHR

A system for online polling to make available for emergency access control system to personally controlled health records (PCHR) presented by Zhang *et al.* [38]. For every request of emergency access, manage the access right according to the collected views of the patient's predefined emergency contacts information and additional online enrolled physicians. Because their operation is based on the demographic amount of the healthcare provider association nationwide, it presents a constant emergency access control at all times. Thummavet and Vasupongayya [15] gave a framework to deal a PHR data in an emergency condition. Because its owner manages a PHR, the crucial challenge in dealing with the PHR in emergency condition is how emergency staffs can obtain PHR data information, even when the PHR owner is incapable of providing his/her consent. The recommended system permits each PHR to classify into three categories. Every category performs a separate limitation and, the emergency medical staffs can access every type according to the procedure established by the PHR owner via trusted members who are granted by PHR owner also. In this work, they adopted threshold cryptosystem to measure the number of grant permissions from the delegated members which are selected by the patient in their PHR. Guan *et al.* [39] proposed a design leveraging the fingerprints of patients to help doctors to obtain short access permission of PCHRs. The doctor gets the fingerprint of the patient as the permission key and uploads it with some necessary information of the patient such as the name and the ID number to the server of PCHRs. The server administers a match among the presented data and the original data saved in the database. If the data are sanctioned, the doctor is allowed short access to the PCHRs. To decrease the space absorbed by the fingerprint and update the authentication performance, they further proposed adopting the principal components analysis (PCA) method to collect and meet the fingerprint.

Rabieh *et al.* [40] proposed a secure medical records access plan that provides emergency access for the patient using cloud server. In this work, an emergency medical center can decrypt a patient's medical records without exposing the secret key used to encrypt them with the guidance of the patient's smartphone and the cloud server. Chen *et al.* [41] to advance the practice of the emergency access control system, this research suggests an automatic add token and identifier scheme (AATIS). When the final token has been moved out the patient can personally update define of the token and identifier. Maw *et al.* [42] proposed the break-the-glass access control (BTG-AC) design that is a remodeled and redesigned variant of the (Ba TG-RBAC) design to approach records availability problem and to discover the security procedure breaches from both approved and unapproved participants. Various modifications in the access control engine are formed in BTG-RBAC to perform the current BTG-AC employ and adjust in WSNs. This study presented a particular design system and enlargement of the BTG-AC model based on a healthcare plot. Blocking and discovery tool and necessity produce more flexible access than other access systems in WSNs.

B. BLOCKCHAIN ACCESS CONTRL AND SHARING

Xia *et al.* [43] presented a blockchain based framework which permits access to only requested, and hereafter approved participants. At the end of this structure, other accountability guarantees as all participants are already known, and the blockchain keeps a log of their movements. The system sanctions participants to request the application from the shared pool after their identification, and cryptographic keys are accepted. The evaluation of the system explains that the system is efficient, scalable and lightweight. Ichikawa *et al.* [44] developed a mHealth scheme for cognitive behavioral treatment for sleeplessness practicing a smartphone application. The volunteer's data information obtained by the application was saved in JSON format and forwarded to the blockchain HF system. Besides, the estimated tamper resistance of the data toward the inequalities produced by artificial flaws. EMR obtained utilizing smartphones were successfully transferred to a permissioned blockchain HF network. They confirmed the data update process under circumstances where all the validating peers were working routinely. Azaria *et al.* [30] until now MedRec functioning prototype is the first and only prototype has been proposed. Xia *et al.* [45], MeDShare, a scheme that discusses the problem of medical information sharing between medical Big data caregivers in a trust-less situation. This blockchain based policy provides control, auditing, and data derivation for shared medical data in cloud silos with necessary and primary information objects. MeDShare observes malicious purposes that access data for use from a data keeper system. In MeDShare, data transformations and sharing from one object to the other, on with all operations completed on the MeDShare system, are recorded in a tamper-proof manner. Dubovitskaya *et al.* [46] presents the prospects on permission-based blockchain healthcare data management, inappropriate, for electronic medical record data sharing within healthcare caregiver and research investigations. The authors proposed a structure on maintaining and sharing electronic medical record data for the care of the cancer patient. In collaboration with Stony Brook University Hospital, and performed their structure in a model that guarantees privacy, security, availability, and fine-grained access control over electronic medical record data. The proposed work can significantly decrease the turnaround time for electronic medical record sharing, enhance decision making for medical care, and decrease the overall expense. Xiao Yue et al. [47] their structure facilitates by employing the blockchain network as a warehouse system, purpose-centric access as one access-control model, unified and straightforward Indicator-centric structure as storage model. Based on their structure, patients don't need to believe any unknown third party and are informed forever who have access to his/her data system and how that user will be used that data. Healthcare data gateways (HDG) system is a decentralized platform, making judicial and administrative judgments for collecting, storing and sharing patient medical data easier. HDG presents an essential way to share healthcare data information while maintaining privacy. These existing studies in emergency access for PHR without blockchain are the traditional system, and also there are some studies based on blockchain but not access to PHR in an emergency condition.

IV. THE PROPOSED SYSTEM

During the emergency scenario, a PHR can support a patient to achieve necessary medical treatment. However, when a victim is unconscious, he/she unable to operate the access control process for providing the PHR to the emergency doctor (ED). Hence, it is essential for the patient to manage the medical record and to be capable of accessing or sharing the health record during the medication and post-treatment monitoring. Due to the movement of a patient, the administration of the data created during the patient's visit can be cumbersome particularly given the sensitive environment of the healthcare data. Now, what is the guarantee that the patient's data are complete, securely saved, and can be accessed just only according to the patient's approval quickly and expediently? We consider this puzzle by applying the permissioned blockchain Hyperledger Composer technology network to create a framework of a PHR sharing system. To perform our solution, we suppose as an example of a Patients PHR system, My HealtheVet Blue button [48]. My HealtheVet proffers it simple for the patient to access and download a copy of their Veterans Affairs (VA) health records. Patients can choose the date range and type of data to cover and generate a single electronic file that consists of all their possible personal health information. This data comprises both the data from the VA electronic health records (EHR) and self-entered data from the patient. The blue button characteristic permits veterans to access and download their data into a simple text file or PDF that can be read, printed, or saved on any computer. The documents that contain the data form patient self-entered and data from the VA EHR such as admission and discharges, problem list, demographic information, emergency contact information, history and physical exams, laboratory results, treatment facilities, medications, herbals, and supplements. Currently, if any of these PHR data required for the emergency condition from the patient PHR and the patient's information is not accessible without the permission of the patient. In the PHR system, the patient introduced the emergency contact for the emergency condition because the patient is unconscious and unable to permit his PHR access. Consent management and data fetching are complicated and inconvenient the EMT staff need to contact emergency contact person in the PHR, and the emergency contact person did not reply on time at any cost, or some other problem may occur with the emergency contact person then what will do the EMT staff at the emergency condition? The patient after cure can trace the existing information in a PHR system that if any doctor already entered with the permission of emergency contact person behalf of the patient, there is no any record to show the tracking. By operating permissioned blockchain technology, our solution provides an ED access for expediting the consent management and speeding up PHR data fetch from the PHR system. We developed a smart contract that will enable a patient to impose permission access control policy for his/her data items easily and permits dynamic PHR data for sharing to ED during the emergency condition.

A. SYSTEM DESIGN

We address the challenge of emergency access for PHR by proposing a framework which defines some permission access rules through the HF. In this framework, the EMT staff can get the access for PHR in an emergency condition under the restrictions of patient's rules through framework. The patient saves the PHR data through the blockchain network and in other words, a PHR carries health data related to an individual patient. Moreover, it is frequently accessed by multiple participants, such as the patient, family members, and family doctor. The blockchain network only gives the PHR data to authorized EMT staff which have the granular access rights from the database according the permissions (patient's rules). The blockchain network stores or updates data requests from the EMT staffs, and provides data access record from the PHR. This record is accountable and traceable from the ledger.

In our system, all the transactions are concerned with authorization and data fetching from the ledger are executed through the smart contracts (a business, logic). The proposed framework operates based on the smart contracts of the ledger which makes the system protected, effective and auditable. Figure 2. shows the proposed architecture of the EACMS for PHR. This architecture consists of some actors such as patient, doctor, ED, database for storing (or updating) the PHR data, smart contracts, ledger, and API for participant interact the system with different activities. In this case, the patient's rules involve the EMT staff during the emergency condition to access his/her PHR. The detail explanation of the entities are as follows.

- *The patient* generally, in a PHR system, the patient is the responsible of the PHR data for authorizing or rejecting the request for his/her data access to any other parties. Also, he/she could select some users such as a family doctor to access his/her PHR. In the EACMS the patient defined the access control policy for users such as a doctor (not in an emergency) and ED. For the doctor (not in an emergency), the patient can permit in normal condition when the doctor (not in an emergency) wants to access the PHR. For the emergency condition, the patient needs to seek medical treatment as soon as possible because the patient has no sense or he is unconscious for authorizing to anyone. So, he/she pre-defined an emergency condition via smart contract in his PHR system to make a decision on behalf of the patient.
- *The doctor* (not in an emergency), is another participant in the EACMS. The patient predefined the access control policy in the smart contracts for the primary physician that he/she can enter in the system and can access the PHR data.
- *Emergency doctor* ED is the leading participant who requests at the time of emergency condition of the

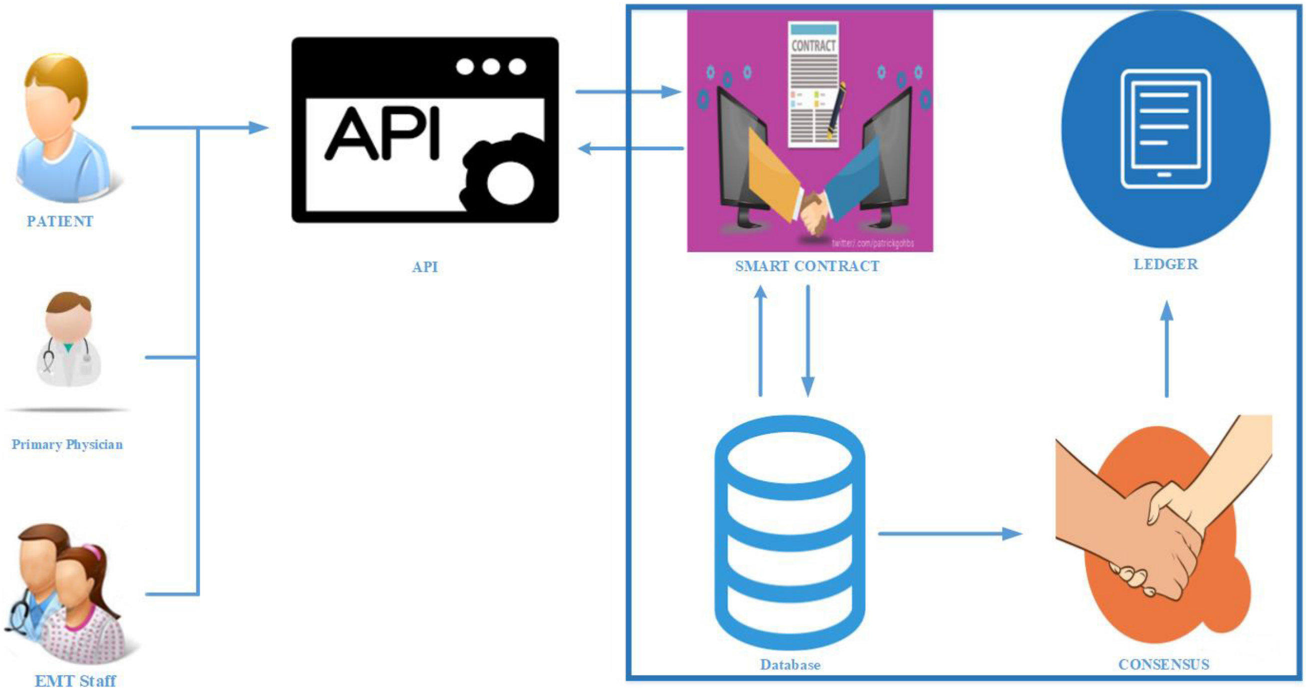

FIGURE 2. Proposed architecture for emergency access for PHR.

patient in this system. The EMT staff gives the request from a simple application program interface API to get access to the system.

- *The Rest API* Hyperledger composer rest server, composer rest-server, can be used to generate a REST API from a deployed blockchain business network that can be easily consumed by HTTP or REST client. In our proposed EACMS the participants accessed through a REST client Interface (postman).
- *The smart contracts* are the rules which are defined by the user that governing transactions [49]. Smart contracts can be considered as digitized variants of contracts that enforce specific laws for transactions by a system network of peers based on described methods [50], [51]. A smart contract is transformed into a computer code, saved, and simulated in the blockchain. It can be made entirely or somewhat self-enforcing and self-executing. In the EACMS, patient-defined the access control policy via smart contracts. When the participant sends the request from API, smart contracts match the endpoint from the database and response to participants.
- *The consensus* presents the following core functionality, in the proposed EACMS all the accuracy of the transactions according to endorsement and consensus strategies confirms by consensus. When the transaction has done in EACMS, the consensus agrees on order and regularity results of execution. The process of access of the assets (PHR) has to reach to the consensus means all the peers nodes have to agree before updating the main distributed ledger.
- *The ledger* is a consequence, temper immune history of all state transactions. State transactions are a fruit of smart contracts, requests sent by participating participants. The result of every transaction in a set of asset key-value pairs that are committed to the ledger as creates, updates, or delete. ED initiates the transactions for reading or writing into the ledger including software development kit SDK client API calls. The geography presented by chaincode is accepted upon by the patient and explain the functionalities provided to the member of the network. It permits granular access inspections to confirm the authentication of the recommended ED.

The back-end comprises of a single server, which has configured routes and these routes are detected and can be called by clients. Every route has connected a similar method with it. These methods utilize the HF NodeSDK to interact with the blockchain network. The back-end is created using NodeJS and Hyperledger Composer-SDK. It is capable of managing client requests and return the response from the blockchain network. To access the database through the system, there is a need for an authorization to access the blockchain network. This is achieved by applying information in java script object notation (JSON) web created during the login process. This network uses a single peer node with the world state database, a single ordering node, and certificate authority (CA). When the network is started, it creates a channel connecting a peer node to install a smart contract (a custom one) on a peer's file system and execute the initialization function on that contract.

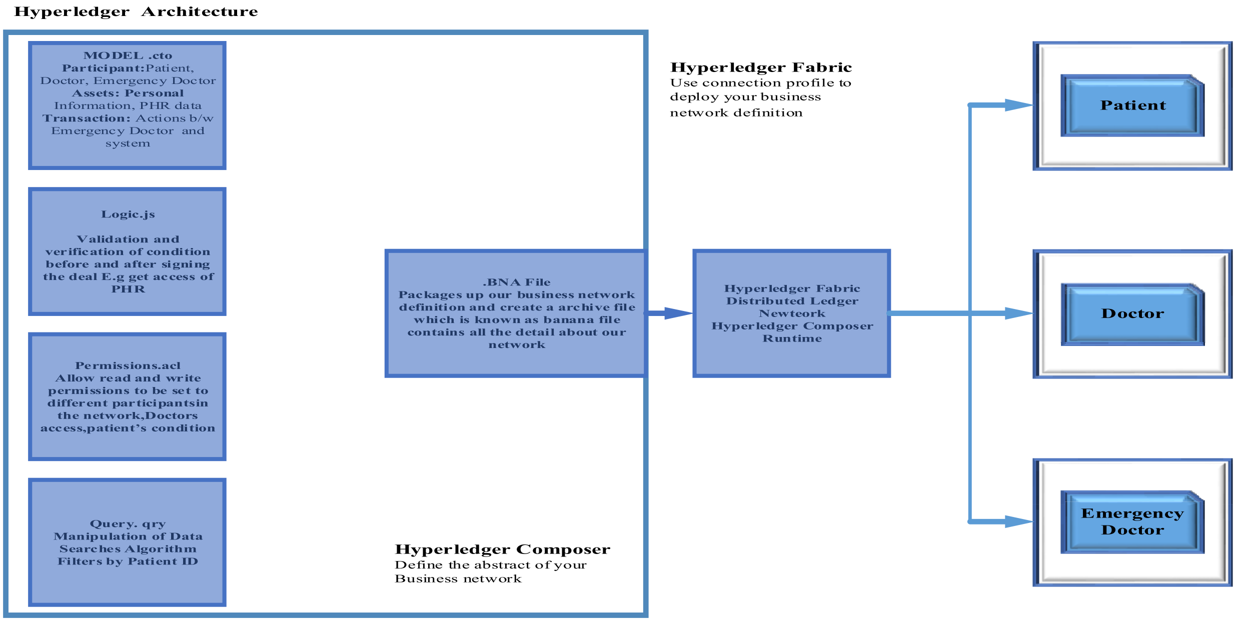
Hyperledger Architecture
MODEL .cto
Participant:Patient, Doctor, Emergency Doctor
Assets: Personal Information, PHR data
Transaction: Actions b/w Emergency Doctor and system
Logic.js
Validation and verification of condition before and after signing the deal E.g get access of PHR
Permissions.acl
Allow read and write permissions to be set to different participantsin the network,Doctors access,patient's condition
Query. qry
Manipulation of Data
Searches Algorithm
Filters by Patient ID
.BNA File
Packages up our business network definition and create a archive file which is known as banana file contains all the detail about our network
Hyperledger Composer
Define the abstract of your Business network
Hyperledger Fabric
Use connection profile to deploy your business network definition
Hyperledger Fabric Distributed Ledger Newteork Hyperledger Composer Runtime
Patient
Doctor
Emergency Doctor

FIGURE 3. Emergency access for PHR with hyperledger composer.

V. SYSTEM IMPLEMENTATION

The HF is a permissioned-based network that offers some limited services for specific users. Our system includes three main components a permissioned blockchain, off-chain storage, and a patient-centric user experience accessed through a REST client interface (postman). Hyperledger composer used to create the business network archive (BNA) that determines the characteristics and capabilities of our system. Hyperledger Composer is also utilized to runtime version of the BNA onto the HF instance. There are three main files created with the composer: A model file, a script file, and the access control list. Figure 3. Shows the detailed description of EACMS for PHR based on the blockchain. In this architecture, the model file consists of three main components: participant that can interact with the system (patient, doctor and ED). Assets are often the variables stored in the variables. The assets are personal information and PHR data items of the patient. Participants are the nodes of the network and can interact with assets and other participants through transactions. Transactions are the functions of the network and are invoked to update the network. The transactions are in the EACMS are the actions between patient, doctor and ED with the system. The logic.js file or the script file defines the various transaction functions in the network. It is written in JavaScript and handles the transaction logic, including validation and verification of the participants, different levels of access in the network and which kind of assets are used in transactions. The "permission.acl" file delineates the specific scopes of access users have in the EACMS for PHR. The access control policy for the EACMS defines in the "permission.acl"file. In this file, authorities are to be set to different participants in the EACMS, such as which participant (patient, doctor and ED) can access the PHR data in normal condition and an emergency condition. Patient defined the access control policy for the doctor that he/she can access with the permission when the patient is conscious. For the emergency condition, the patient defined the rules for an ED that he/she can access with his certified license number. The ED triggered the smart contract and obtained the PHR data with the function of emergency access time constraints. After ending the time limit which is defined in the function of emergency access time constraints the ED cannot access the PHR data items. The query file defines the structure and function of queries from this EACMS. Queries are set to extrapolate transactions from the historian, which is a ledger record for all the past transactions in the EACMS for PHR.

Once the network is defined, it can be exported as an archive, downloaded, and run on another machine. A network card is used to connect to the system. Network cards can take the form of a participant type. Participant cards generally have a more controlled scope of access in the EACMS, while the patient can perform more high-clearance functions such as adding new participants or deleting participants such as doctor and ED. This card type defines the node that uses the card to connect to the network and, thus, allows to permit that what kind of doctor and ED role the node plays.

we described Figure 4. In detail that participants are admitted to be the original artists in the network which require to save transaction information. In our emergency access control model, the participants are patient, doctor and ED. In Hyperledger composer the formation of a participant is described

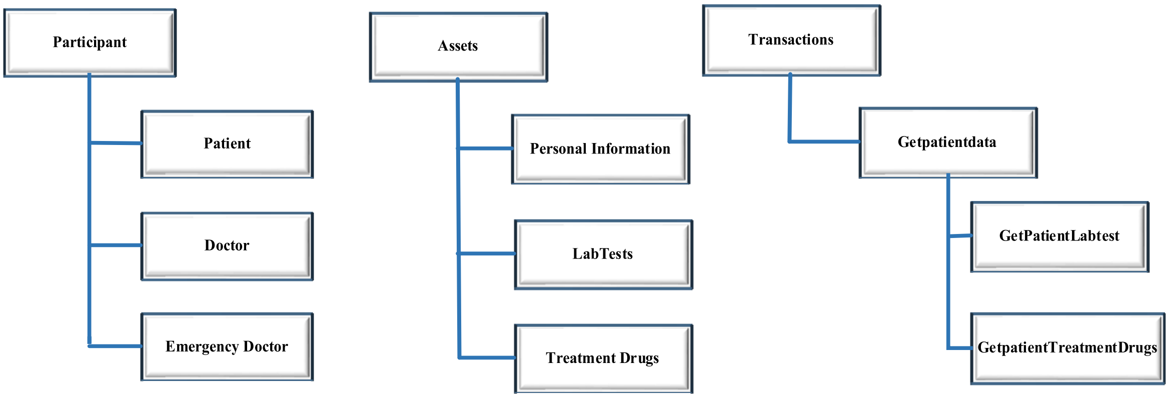

FIGURE 4. Participant, assets and transaction in EACMS.

in the model file. New cases of the modeled participant can be generated and appended in the participant registry. Hyperledger composer network also required blockchain identities as a form of credentials and a set of mappings of characters to participants are stored in identity registry. All the identity management services are conducted by admin peer controlled by consortium organization which deploys blockchain. In our EACMS, at any instance of time, new participants (Doctor, ED) with suitable identity functions can be added to manage a particular case by admin peer administered by an organization of Hyperledger composer consortium. Access control policy for participants that, what features exist and which services can perform which type of transactions are defined in "permissions.acl" file of Hyperledger composer network. Authorization to an ED will be granted by the patient controlled by an organization of blockchain consortium.

Assets are anything of content that can be accomplished or partaken over the network. Also, they can be physical ones like a table, houses, cars or gold to intangible ones like securities, mental attribute. In our framework, the PHR items such as, personal basic information lab test and treatment drugs, etc. are the assets and are saved in the assets registry of the hyperledger composer network.

Transactions explain the operations that could be executed on the assets by participants as they run along the network. In EACMS, the transactions are registration process for the participants and process for the PHR data items such as "getpatientlabtest" and "getpatienttreatment-drugs". In addition, each specific record the particulars about the patient's data items and display PHR on the network.

Our system comprises four functions for register patient, register doctor, register ED and get_patient_data from the Hyperledger composer blockchain. These functions can be seen as Hyperledger composer transaction triggered by the participants (Patient, Doctor, ED) in the network. The constraint like who(participant) should access what role and under which condition access should be given to participants, are all presets in access control policy in "permission.acl" file of emergency access control network in the Hyperledger composer.

- *Patient registration:* For the patient, registration function enters the (card_Id, first_name, last_name, address, patient_Id, Emergency_Access_Time_Costraints) as input and submits the request through the API in the system. Whenever the client hit the API the node server will search for its end point matching method in app.js file. After getting all parameters from body of the requested method, this function call blockchain transaction processing function (TPF) define in "network.js". In this file, all the functions are TPFs which utilize Hyperledger composer NodeSDK functions to register patient as a network participant. After registering the patient, it creates identity card for the patient as well and stored in the identity wallet. Algorithm 1 summarizes the patient registration function in details.

Algorithm 1 Patient Registration

```
Input: Patient ID, First Name, Last Name
Output: Register the Patient as a participant
1. Patient ID ← Patient;
2. First Name ← Patient First Name;
3.       Last Name ← Patient Last Name;
4.       Patient ID ← Request for the registration to the
           system;
5.   If (Patient description match) then
6.       Return Success;
7.   else
8.       Return "error";
9. end if
```

- *Doctor registration:* For the doctor, registration function enters the (Doctor_Id, first_name, last_name) as input and submits the request through the API in the system. Whenever the client hits the API, then the node server will search for its end point matching method in "app.js" file. After getting all parameters from the body of the requested method, this function call blockchain

Algorithm 2 Doctor Registration

```
Input: Doctor ID, First Name, Last Name
Output: Register the Doctor as a participant
1. Doctor ID $\leftarrow$ Doctor;
2. First Name $\leftarrow$ Doctor First Name;
3. Last Name $\leftarrow$ Doctor Last Name;
4.     Doctor ID $\leftarrow$ Request for the registration to the
           system;
5.  If (Doctor description match) then
6.      Return Success;
7.  else
8.      Return "error";
9. end if
```

Algorithm 3 Emergency Doctor Registration

```
Input: Emergency Doctor ID, License Number
Output: Register the Emergency Doctor as a participant
1. Patient ID $\leftarrow$ Emergency doctor;
2. License $\leftarrow$ Emergency Doctor License Number;
3.  Emergency Doctor ID $\leftarrow$ Request for the registration
     to the system;
4.  If (Emergency Doctor License Number match) then
5.      Return Success;
6.  else
7.      Return "Unauthorized Person";
8. end if
```

Algorithm 4 Get the Patient Data

```
Input: Patient ID, Emergency Doctor ID
Output: Display the Patient PHR data items
1. Patient ID $\leftarrow$ Discover (Registration ID from the participant registry);
2. Emergency Doctor ID $\leftarrow$ (Registration ID from the participant registry);
3.     Get Patient Data $\leftarrow$ Emergency Doctor request to get patient data;
4.     Start time $\leftarrow$ get the correct time date;
5.   If (Emergency Doctor request = true) then
6.     Result $\leftarrow$ check the Emergency Access Time constraint condition according to the start time;
7.     else
8.     Return "Access Denied";
9.   end if
```

TPF define in "network.js". In this file, all the functions are TPF which employ the Hyperledger composer NodeSDK functions to register doctor as a network participant. After registering the doctor, it creates identity card for the doctor as well and stored in the identity wallet. Algorithm 2 summarizes the doctor registration function in details.

- *Emergency doctor registration:* For the ED registration function enters the (Emerency_doctor_id and License_number) as input and submits the request in the system. Whenever the client hits the API the node server will search for its end point matching method in "app.js" file. After getting all parameters from body of requested method, this function call blockchain transaction processing function define in network.js. In the "network.js" file all the functions are TPF which are using Hyperledger composer NodeSDK functions to register emergency_doctor as a network participant. After registering the emergency_ doctor, it creates identity card for the emergency_doctor as well and stored in the identity wallet. Algorithm 3 summarizes the ED registration function in details.

- *Get patient data:* For the access of the patient data in the emergency condition, get_patient_data function enters the (Patient_Id and currently participant emergency_doctor_Id ID) as input. Whenever the endpoint hit from the client, the "network.getpatientdata" defined in the "app.js" file will have triggered. After getting the required fields from the request method body, it will transmit these records to the TPFs "networkgetpatientdata" defined in "network.js" file, which return the data according to checking defined rules. The TPF again using Hyperledger composer NodeSDK, first it checks whether the participant has permission to access to the PHR, i.e., according to permissions of the network, it returns patient data. The smart contract assign duration based on the time limitation as defined by the patient. Algorithm 4 summarizes the get_patient_data function in details.

Additionally, it creates an instance of "EmergencyTimeConstratints" which user could view from playground using admin identity card. During the defined time limitation, the current "Emergency_doctor" can see patient data. Emergecy_Access_End_Time always be two hours greater than Emergency_Access_Start_Time. make them equal and recheck by hitting get_Patient_data now ED will receive message "Access Denied".

VI. PERFORMANCE ANALYSIS AND DISCUSSION

In this study, we have implemented the proposed EACMS using the blockchain technology. The HF is permissioned-based network invited only known user as a participant. We then evaluated our framework by employing hyperledger composer business network based on the HF blockchain network. For the test, the client data format was JSON, and in the experiments, the data were input using rest client, i.e., postman servers. Each server was constructed in the virtual environment EC2 instance on AWS, which ran in the same local personal computer with Linux 16.04.1, 16 GB RAM and 1. GHz single vCPU. For the construction of the virtual environment, we utilized Docker version 1.10.2, Oracle Virtual Box version 5.1.21, and we used Docker-compose version 1.5.2 to manage Docker. The state was the key-value store database and recorded the result of the transactions.

We evaluate the system performance, regarding response time, the privacy of the PHR and data accessibility.

TABLE 3. Highlights and limitations of the EACMS vs traditional emergency access system.

Factors	Traditional Emergency Access System [15, 16]	The Proposed EACMS based on Blockchain
Response Time	➢ There are some trusted members in the emergency contact list, that the EMT staff requests to the system for calling trusted members and waiting the response. In this case, there is unpredictable delay of time for emergency access.	➢ The ED does not need to call or conform to any person or third party for approving the PHR access in emergency condition because the patient predefined the emergency access policy (permissions) via smart contracts.
Privacy	➢ The EMT staff can only access to whole data through the PHRs, that all of them may not be authorized members.	➢ The EACMS assures the patient's privacy by providing feasibility for specifying granular access control across his/her PHR data via smart contracts.
Security	➢ From the security point of view, malicious users could have unauthorized access to the system by using an EMT account. ➢ The administration of each Emergency response unit staff is employed as a key performance indicator during the emergency authority evaluation process, any impropriety by the ERU staff impacts its EA efficiency.	➢ It works based on a decentralized network topology, which does not have a single point of failure. The EACMS protects the PHR data against security breaches such as unauthorized access.
Accessibility	➢ Emergency staff does not directly involve with the PHR owner. Thus, these persons are not in the PHR system, so, there is no access right for them. Hence, the emergency staffs are not able to access a PHR of the patients even in emergencies. ➢ The traditional systems are centralized if one of the nodes down all system will be down.	➢ The EACMS provides a high speed and protected access to the PHR data under patient's rules. It assures the availability of PHR data items without validation by any trusted member or third party.

The blockchain is a moderately new development, and framework such as our recommended framework is entirely separate from existing systems. In the previous studies, emergency access systems have been proposed in [15], [16] In their experiments, it can be observe that such systems requires much time for providing access to PHR data, and they have security problems due to depending on a single trusted authority. The EACMS prevents these risks by employing a distributed blockchain technology network. Table 3 summarizes the highlights and limitations of our EACMS versus traditional emergency system.

A. RESPONSE TIME

Smart contracts include several rules such as time limitation, identification, etc. that play a significant role in response time. In the traditional emergency access systems [15], [16], the authors introduced emergency contact members or numbers for accessing the patient's PHR data. But, in our framework, there are no any trusted member for emergency contact because we use Hyperledger composer in which the patient assigns smart contracts to an emergency condition for the ED request. So, the ED can get the patients PHR access without any time delay.

As we already described, the EACMS is designed for the emergency condition at the hospital to treat the patient when he/she could not operate the access authorization at the emergency location. Therefore, there is a priority between patient arrival at the hospital during which the ED can register in the EACMS and get patient data from the PHRs.

The processing time to retrieve the registration and get access to the PHR are included the following procedures:

1) The ED sends the registration request along with the "emergency_doctor_Id" and "License_number",
2) The node server searches for its endpoint matching method in "app.js" file,
3) After getting all parameters from body of requested method, this function call blockchain TPF define in "network.js",
4) Using Hyperledger composer NodeSDK functions to register patient as network participant,
5) Get_Patient_data function enters the Patient_Id and current participant Emergency_Doctor_Id,
6) After getting the required fields from request method body, it transmits these field to TPF "networkgetpatientdata" defined in "network.js",
7) TPF utilizes the Hyperledger composer "NodeSDK" to check whether the participant has permission access to the patient data,
8) If the participant has the permission to access the patient's data according to permissions of the network, then it returns PHR data.

In [15], [16], the average response time of receiving text messages from the trusted members is "431.28s" during simultaneous conversations. Therefore, it can take up to

TABLE 4. Response time and memory usage.

Transaction	Response Time	Memory Usage
Patient Registration	6002 ms	236 B
Doctor Registration	6231 ms	236 B
ED Registration	5963 ms	236 B
Get patient Data	5683 ms	568 B

"7 minutes". The trusted user gives the respond to emergency staff for approving the PHR access, it is considering the "8-minutes", for receiving call response time. But, in practice, the EACMS takes the time for average registration time is "6.9 s" and after registration, get patient data average time is "6 s." Therefore, our system takes up to "15 s" to "18 s" for replying to ED. Table 4 depicts the response time & memory usage of the proposed framework during the request and get PHR data processes.

B. PRIVACY AND SECURITY

The EACMS assures the patient's privacy by providing feasibility for specifying granular access control across his/her PHR data. Moreover, it considers the access control management by including smart contracts. In the Hyperledger composer network, the proposed model operates based on the defined participant's identities. Therefore, there are no ways to access the PHR data for malicious users. Channels in the HF are constructed according to access policies that dictate access to the channel's stores such as smart contracts, transactions, and ledger state. Thus, these channels consist of nodes in which the privacy protection and confidentiality of PHR defined within them. The EACMS protects the PHR data against ransomware and similar security breaches such as unauthorized access. Because it is the decentralized network topology, and does not have a single point of failure or central repository for intruders to infiltrate. The ED has just short, timely access in the system, after the time limitation of his/her access data the ED could not access the PHR data.

C. ACCESSABILITY

Our framework provides a high speed and protected access to the PHR data under patient's rules. It assurances the availability of PHR data items without validation by any trusted member or third party. This system could reduce the time cost in emergency compared to the process of phone calling or messaging to trusted users for authorizing the access control of PHR. In general, patients often have the burden of recalling their past medication history by memory or carry around medical documents in hard copies. Using the decentralized ledger system, prescribers can easily update medication histories through a simple client user interface. When patients visit different medical institutions, prescribers can query medication histories easily with the approval of the patient. The decentralized network eliminates the need to cooperate with a set of privatized central repositories.

VII. CONCLUSION

This paper proposes an Emergency Access Control Management System called EACMS which provides privacy protection and security policies for the patient's PHR in emergency condition. Technically, the EACMS operates based on the Hyperledger Composer network which is a permissioned based blockchain technology. Hence, the PHR data is only confined to the users who are the known members of the blockchain, accepted by admin peer declared by organizations of the consortium. The proposed framework considers how to get access to a patient's PHR in an emergency condition using blockchain the HF and Hyperledger composer. It initializes some rules for accessing the emergency control management of PHR. The EACMS stores patient's data in an immutable and transparent ledger, which grips track of all the transactions on the system, that could enhance the management of health data. We implemented the EACMS through the HF blockchain to evaluate the efficiency of our framework. The experimental results confirm that this framework provides the security of sensitive patient's PHR data items so that it guarantees time efficiency, and privacy, accessibility, and granular access control management.

REFERENCES

[1] X. Zhang, C. Yang, S. Nepal, C. Liu, W. Dou, and J. Chen, "A MapReduce based approach of scalable multidimensional anonymization for big data privacy preservation on cloud," in *Proc. 3rd Int. Conf. Cloud Green Comput. (CGC)*, Sep. 2013, pp. 105–112.

[2] S. Wan, Y. Zhao, T. Wang, Z. Gu, Q. H. Abbasi, and K.-K. R. Choo, "Multi-dimensional data indexing and range query processing via Voronoi diagram for Internet of Things," *Future Gener. Comput. Syst.*, vol. 91, pp. 382–391, Feb. 2019.

[3] U.S. Department of Health and Human Services. (2008). *Personal Health Records and the HIPAA Privacy Rule*. [Online]. Available: http://www.hhs.gov

[4] I. C. Señor, J. L. Fernández-Alemán, and A. Toval, "Are personal health records safe? A review of free Web-accessible personal health record privacy policies," *J. Med. Internet Res.*, vol. 14, no. 4, p. e114, 2012.

[5] D. A. Jones, J. P. Shipman, D. A. Plaut, and C. R. Selden, "Characteristics of personal health records: Findings of the medical library association/National library of medicine joint electronic personal health record task force," *J. Med. Library Assoc.*, vol. 98, no. 3, pp. 243–249, 2010.

[6] L. Fernandez-Luque, R. Karlsen, and J. Bonander, "Review of extracting information from the social Web for health personalization," *J. Med. Internet Res.*, vol. 13, no. 1, p. e15, 2011.

[7] P. C. Tang, J. S. Ash, D. W. Bates, J. M. Overhage, and D. Z. Sands, "Personal health records: Definitions, benefits, and strategies for overcoming barriers to adoption," *J. Amer. Med. Inform. Assoc.*, vol. 13, no. 2, pp. 121–126, 2006.

[8] Z. Gao, D. Y. Wang, S. H. Wan, H. Zhang, and Y. L. Wang, "Cognitive-inspired class-statistic matching with triple-constrain for camera free 3D object retrieval," *Future Gener. Comput. Syst.*, vol. 94, pp. 641–653, May 2019.

[9] Y. Xia, S. Qu, and S. Wan, "Scene guided colorization using neural networks," *Neural Comput. Appl.*, pp. 1–14, Oct. 2018. doi: 10.1007/s00521-018-3828-z.

[10] B. Adida and I. S. Kohane, "GenePING: Secure, scalable management of personal genomic data," *BMC Genomics*, vol. 7, no. 1, p. 93, 2006.

[11] P. Zhang, J. White, D. C. Schmidt, G. Lenz, and S. T. Rosenbloom, "FHIRChain: Applying blockchain to securely and scalably share clinical data," *Comput. Struct. Biotechnol. J.*, vol. 16, pp. 267–278, Jul. 2018.

[12] S. Ding, S. Qu, Y. Xi, and S. Wan, "A long video caption generation algorithm for big video data retrieval," *Future Gener. Comput. Syst.*, vol. 93, pp. 583–595, Apr. 2019.

[13] Y. Zhao, H. Li, S. Wan, A. Sekuboyina, X. Hu, G. Tetteh, M. Piraud, and B. Menze, "Knowledge-aided convolutional neural network for small organ segmentation," *IEEE J. Biomed. Health Inform.*, to be published.

[14] I. Masood, Y. Wang, A. Daud, N. R. Aljohani, and H. Dawood, "Towards smart healthcare: Patient data privacy and security in sensor-cloud infrastructure," *Wireless Commun. Mobile Comput.*, vol. 2018, Nov. 2018, Art. no. 2143897.

[15] P. Thummavet and S. Vasupongayya, "A novel personal health record system for handling emergency situations," in *Proc. Int. Comput. Sci. Eng. Conf. (ICSEC)*, Sep. 2013, pp. 266–271.

[16] P. Thummavet and S. Vasupongayya, "Privacy-preserving emergency access control for personal health records," *Maejo Int. J. Sci. Technol.*, vol. 9, no. 1, pp. 108–120, 2015.

[17] M. Benchoufi and P. Ravaud, "Blockchain technology for improving clinical research quality," *Trials*, vol. 18, no. 1, p. 335, 2017.

[18] S. Wang, Y. Zhang, and Y. Zhang, "A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems," *IEEE Access*, vol. 6, pp. 38437–38450, Jun. 2018.

[19] (2018). *Welcome to the Hyperledger Composer*. [Online]. Available: https://hyperledger.github.io/composer/latest/introduction/introduction.html

[20] D. Puthal, N. Malik, S. P. Mohanty, E. Kougianos, and C. Yang, "The blockchain as a decentralized security framework [future directions]," *IEEE Consum. Electron. Mag.*, vol. 7, no. 2, pp. 18–21, Mar. 2018.

[21] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Tech. Rep., 2008. [Online]. Available: http://bitcoin.org/bitcoin.pdf

[22] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "FairAccess: A new blockchain-based access control framework for the Internet of Things," *Secur. Commun. Netw.*, vol. 9, no. 18, pp. 5943–5964, 2016.

[23] A. Ouaddah, A. A. Elkalam, and A. A. Ouahman, "Towards a novel privacy-preserving access control model based on blockchain technology in IoT," in *Europe and MENA Cooperation Advances in Information and Communication Technologies*. Cham, Switzerland: Springer, 2017, pp. 523–533.

[24] 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, Jan. 2018.

[25] M. Mettler, "Blockchain technology in healthcare: The revolution starts here," in *Proc. IEEE 18th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Sep. 2016, pp. 1–3.

[26] M. Hölbl, M. Kompara, A. Kamišalić, and L. N. Zlatolas, "A systematic review of the use of blockchain in healthcare," *Symmetry*, vol. 10, no. 10, p. 470, 2018.

[27] N. Kshetri, "Blockchain's roles in strengthening cybersecurity and protecting privacy," *Telecommun. Policy*, vol. 41, no. 10, pp. 1027–1038, 2017.

[28] V. Buterin, "A next-generation smart contract and decentralized application platform," White Paper, 2014. [Online]. Available: https://ethereum.org/

[29] X. Liang, S. Shetty, J. Zhao, D. Bowden, D. Li, and J. Liu, "Towards decentralized accountability and self-sovereignty in healthcare systems," in *Proc. Int. Conf. Inf. Commun. Secur.* Cham, Switzerland: Springer, 2017, pp. 387–398.

[30] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," in *Proc. Int. Conf. Open Big Data (OBD)*, Aug. 2016, pp. 25–30.

[31] H. Kakavand, N. Kost De Sevres, and B. Chilton, "The blockchain revolution: An analysis of regulation and technology related to distributed ledger technologies," Jan. 2017. doi: 10.2139/ssrn.2849251.

[32] M. Castro and B. Liskov, "Practical byzantine fault tolerance and proactive recovery," *ACM Trans. Comput. Syst.*, vol. 20, no. 4, pp. 398–461, 2002.

[33] A. Kapur. (2019). *Blockchain Explained: What it is and How it Works*. [Online]. Available: https://ggktech.com/blockchain-explained-how-works/

[34] (2018). *Permissionless vs. Permissioned Blockchain Networks*. Accessed: Jul. 4, 2018. [Online]. Available: https://nxxtech.com/newsroom/blog/Permissionless-vs-permissioned-blockchain-networks.php

[35] SEPA. (2017). *The Difference Between Permissioned and Permissionless Blockchains*. [Online]. Available: https://www.sepaforcorporates.com/thoughts/difference-between-permissioned-permissionless-blockchains/

[36] P. Thakkar, S. Nathan, and B. Vishwanathan, "Performance benchmarking and optimizing hyperledger fabric blockchain platform," 2018, *arXiv:1805.11390*. [Online]. Available: https://arxiv.org/abs/1805.11390

[37] Hyperledger. (2017). *Architecture Explained Read the Docs*. [Online]. Available: https://hyperledger-fabric.readthedocs.io/en/release-1.2/arch-deep-dive.html

[38] Y. Zhang, S. Dhileepan, M. Schmidt, and S. Zhong, "Emergency access for online personally controlled health records system," *Inform. Health Social Care*, vol. 37, no. 3, pp. 190–202, 2012.

[39] S. Guan, Y. Wang, and J. Shen, "Fingerprint-based access to personally controlled health records in emergency situations," *Sci. China Inf. Sci.*, vol. 61, no. 5, 2018, Art. no. 059103.

[40] K. Rabieh, K. Akkaya, U. Karabiyik, and J. Qamruddin, "A secure and cloud-based medical records access scheme for on-road emergencies," in *Proc. 15th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2018, pp. 1–8.

[41] Y.-Y. Chen, C.-C. Huang, and J.-K. Jan, "The design of AATIS emergency access authorization for personally controlled online health records," *J. Med. Biol. Eng.*, vol. 35, no. 6, pp. 765–774, 2015.

[42] H. A. Maw, H. Xiao, B. Christianson, and J. A. Malcolm, "BTG-AC: Break-the-glass access control model for medical data in wireless sensor networks," *IEEE J. Biomed. Health Inform.*, vol. 20, no. 3, pp. 763–774, May 2016.

[43] Q. Xia, E. B. Sifah, A. Smahi, S. Amofa, and X. Zhang, "BBDS: Blockchain-based data sharing for electronic medical records in cloud environments," *Information*, vol. 8, no. 2, p. 44, 2017.

[44] D. Ichikawa, M. Kashiyama, and T. Ueno, "Tamper-resistant mobile health using blockchain technology," *JMIR mHealth uHealth*, vol. 5, no. 7, p. e111, 2017.

[45] Q. I. 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.

[46] A. Dubovitskaya, Z. Xu, S. Ryu, M. Schumacher, and F. Wang, "Secure and trustable electronic medical records sharing using blockchain," in *Proc. AMIA Annu. Symp.*, 2017, pp. 650–659.

[47] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare data gateways: Found healthcare intelligence on blockchain with novel privacy risk control," *J. Med. Syst.*, vol. 40, no. 10, p. 218, 2016.

[48] HealtheVet. (2017). *Manage Your Health Records*. [Online]. Available: https://www.myhealth.va.gov/mhv-portal-web/web/myhealthevet/download-your-own-va-medical-records

[49] K. Delmolino, M. Arnett, A. Kosba, A. Miller, and E. Shi, "Step by step towards creating a safe smart contract: Lessons and insights from a cryptocurrency lab," in *Proc. Int. Conf. Financial Cryptogr. Data Secur.* Springer, 2016, pp. 79–94.

[50] N. Szabo, "Smart contracts: Building blocks for digital markets," *EXTROPY, J. Transhumanist Thought*, no. 16, p. 18, 1996.

[51] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

AHMED RAZA RAJPUT received the B.S.I.T. degree from the Information Technology Center, Sindh Agriculture University Tando Jam, Pakistan, and the M.Sc. degree in computer application technology from the Nanjing University of Science and Technology, China, in 2015. He is currently pursuing the Ph.D. degree in computer science with the School of Computer Science and Engineering. His research interests include the Internet of Things, bitcoin, and blockchain security.

QIANMU LI received the B.Sc. and Ph.D. degrees from the Nanjing University of Science and Technology, China, in 2001 and 2005, respectively, where he is currently a Full Professor with the School of Computer Science and Engineering. His research interests include information security, computing system management, and data mining. He received the China Network and Information Security Outstanding Talent Award, in 2016, and multiple Education Ministry Science and Technology Awards, in 2012. He is the author or the coauthor of more than 100 high indexed (SCIE/E-SCI/EI) journal/conference papers and eight books.

MILAD TALEBY AHVANOOEY received the B.Sc. degree in software engineering from UAST, Semnan, Iran, in 2012, and the M.Sc. degree in computer engineering from IAU Science & Research, Tehran, Iran, in 2014. He is currently pursuing the Ph.D. degree in computer science with the Nanjing University of Science and Technology, Nanjing, China. He is also a Senior Programmer, a Cybersecurity Researcher, and a Practitioner with industry and academic experience. His research interests include modern coding theory, text hiding, text mining, and genetic programming. He is an external Reviewer of various international journals, including the IEEE Access, *Computers in Human Behavior*, and *KSII Transactions on Internet and Information Systems*.

ISMA MASOOD received the B.S. and M.S. degrees from International Islamic University Islamabad. She is currently pursuing the Ph.D. degree with the Nanjing University of Science and Technology, China. Her research interests include patient data privacy and security, sensor cloud infrastructure, and blockchain.

• • •