A BLOCKCHAIN BASED SECURITY MECHANISM FOR EMERGENCY HEALTHCARE SYSTEMS

A Thesis

by

Elnaz Dadvar

Submitted to the
Graduate School of Sciences and Engineering
In Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in the
Department of Computer Science

Özyeğin University
December 2021

A BLOCKCHAIN BASED SECURITY MECHANISM FOR EMERGENCY HEALTHCARE SYSTEMS

Approved by:

Assistant Professor Kubra Kalkan, Advisor
Department of Computer Science
Özyeğin University

Associate Professor Hasan Sozer
Department of Computer Science
Özyeğin University

Assistant Professor Ahmet Onur Durahim
Department of Management Information systems
Boğaziçi University

Date Approved: 21 December 2021

Dedicated to the memory of my beloved grandfather, Jamshid, who have meant and continue to mean so much to me, whom I lost towards the end of my master studies, the one who always supported me with his endless love, and believed in my ability to be successful in the academic arena. I deeply wish I had the chance to to share this moment with him. You are gone but your belief in me has made this journey possible. I will make sure your memory lives on as long as I shall live.

ABSTRACT

Electronic health records (EHRs) play a crucial role in today's healthcare industry, these records include sensitive and private healthcare data assets which are prone to the breach of security and confidentiality. These potential data breaches may have lots of consequences such as violating patient's privacy, unauthorized access to EHR, data alteration and putting the patient's life in danger. Recently proposed EHR systems have came along with strong security features for preserving the patient's safety and security. However, there is still issues regarding access control management to Personal Health Records (PHRs). In previously presented systems patient plays the main role in controlling the access to the system. And this rises an uncertainty in emergency conditions, while the patient is incapable of issuing any access permission. In this study we suggest a new framework for maintaining medical healthcare records (MHRs) in emergency conditions by a secure and private access control architecture that is designed and employed based on a permissioned blockchain hyperledger sawtooth. Leveraging unique properties of blockchain our system manages to provide a tamper-proof and secure access to patient's medical data in a short period of time in an emergency scenario. For performance analysis the proposed architecture is implemented through hyperledger sawtooth blockchain network. The numerical results of our experiment demonstrated the feasibility and superiority of our proposed architecture in compared with the similar proposed healthcare systems concerning response time, memory consumption, throughput and overall privacy and security.

ÖZETÇE

Elektronik sağlık kayıtları (ESK'ler) günümüz sağlık endüstrisinde çok önemli bir rol oynamaktadır, bu kayıtlar güvenlik ve gizliliğin ihlaline açık hassas ve özel sağlık veri varlıklarını içermektedir. Bu olası veri ihlallerinin hastanın mahremiyetinin ihlali, ESK'ye yetkisiz erişim, veri değişikliği ve hastanın hayatını tehlikeye atma gibi birçok sonucu olabilir. Son zamanlarda önerilen ESK sistemleri, hastanın güvenliğini korumak için güçlü güvenlik özellikleri ile birlikte geliyor. Ancak, Kişisel Sağlık Kayıtlarına (KSK'ler) erişim kontrolü yönetimi ile ilgili hala sorunlar bulunmaktadır. Daha önce sunulan sistemlerde, sisteme erişimi kontrol etmede hasta ana rolü oynar, ve bu acil durumlarda bir belirsizliğe yol açarken, hasta herhangi bir erişim izni veremez. Bu çalışmada, Hyperledger Sawtooth blockzincir ile tasarlanan güvenli ve özel bir erişim kontrol mimarisi sunarak acil durumlarda tıbbi sağlık kayıtlarının (TSK'ler) tutulması için yeni bir çerçeve öneriyoruz. Blok zincirinin benzersiz özelliklerinden yararlanan sistemimiz, acil bir senaryoda hastanın tıbbi verilerine kısa sürede güvenli bir erişim sağlar. Simülasyonumuzun sayısal sonuçları, yanıt süresi, bellek tüketimi, verim, genel gizlilik ve güvenlik açısından benzer sağlık sistemleriyle karşılaştırıldığında önerilen mimarimizin performansının daha iyi olduğunu ve kullanılabilirleğini gösterir.

ACKNOWLEDGEMENTS

I would like to acknowledge and give my warmest gratitude to my supervisor, Professor Kubra Kalkan, for her continuous support, advice, and encouragements throughout the course of my master studies. Her immense knowledge and plentiful experience have encouraged me in all the time of my academic research and daily life. I could not have imagined having a better advisor and mentor for my master study.

I would also like to sincerely thank the members of my thesis committee, Professor Hasan Sozer, and Professor Ahmet Onur Durahim for their valuable time and comments on my thesis.

Last, but not least, I would like to express my very profound gratitude to my parents Houshang and Felor, my sisters Aysan and Aynaz, whom I owe my whole life for their endless support, encouragement, love and blessings.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Healthcare has a vital role in today's society, because it is concerned with improving the quality of life. Traditionally, healthcare information recording was paper based, which was prone to alteration and missing, it was also hard and time taking to access data when required. Researchers came up with the idea of digitizing the medical data and integrating it with IoT, to perform the tasks that may lead to serious breakthrough in healthcare, including (1) automated healthcare record system; (2) sharing reliable information between trusted parties; (3) analysing big data; and (4) collaboration in clinical practice and diagnosis [1]. However, along with these advancements a lot of security and privacy challenges have been risen. For example, there is storage limitation in databases for this huge and ever-growing amount of data which are also exposed to cyberattacks. Patient's sensitive data may be accessed by attackers which then can be altered or utilized to the detriment of the patient. So, it is not reliable to utilize a centralized database system, because chance of cyberattacks is rising in a centralized system. This is where a peer-to-peer (P2P) network comes in handy to enable the decentralization feature [2]. Therefore, blockchain technology has risen as a practical solution and transparent mechanism for storing and distributing the data with the potentiality of dealing with data security, privacy, and integrity issues in medical healthcare, such as tampering and data leakage threats [3]. Blockchain technology provides a distributed, immutable, and secure system for all transactions. It can revolutionize medical database interoperability, and overall blockchain technology has the potential to dramatically improve medical care systems [4]. In the coming section we propose and discuss the existing challenges of healthcare system precisely, and the motivation behind this research study.

1.1 Motivation

Due to the policy restrictions around sharing and accessing sensitive medical assets, there has always been problems when patients want to visit another medical center where there is no record of their medical history and sometimes this is leading to carry out the same medical examinations over and over again which has lots of drawbacks such as time loss in the diagnosis and treatment process and lots of unwanted extra costs and effort for both patient and medical center. As a result of this procedure, health records of patients are scattered all over different medical institutions prompting to a low level of integrity and interoperability. With the advancement of technologies and the vitality for tackling these problems, a new era in healthcare industry has began by moving from traditional-based records storing to the emerging technologies such as IoT devices and cloud computing. However in these systems the records are stored in a centralized manner and this rises other concerns and vulnerability of the system such as higher risks of privacy and security for patients, demanding longer access time to the data when the user is far from the server, and the system is prone to failure. This is where utilizing a peer-to-peer network can guarantee to handle these issues [2]. In recent years blockchain which is fully decentralized technology is introduced and applied as a feasible solution to meet these and many other challenges in healthcare systems [3]. Blockchain as a distributed ledger has the possibility of storing digital healthcare assets in a way that all records can be tracked by the asset owners anytime and anywhere, without a need for a central server [5] [6].

This technology is able to provide full control on the medical data managing in a scalable manner and handles the accessibility and interoperability issues in the data sharing process [7]. Blockchain is proved to offer a promising framework for storing Electronic Medical Records (EMRs). In [8] researchers have utilized the smart contract property of blockchain and designed a unique architecture for EMR storing that can guarantee the integrity in the stored records, and enhance the interoperability of the system. However in all latest research studies the proposed

systems are patient-centric, which leaves uncertainty for emergency conditions. When a patient is unconscious unlike other normal conditions the patient her/his self cannot give access permission to the medical data, but in an emergency case this access needs to be done in a secure and timely mannered way to fulfill the various threats in this situation. Some research works have been done to meet this challenge by utilizing blockchain technology in recent years. In these studies hyperledger fabric is used as a practical platform for providing secure PHR data storage and data access in emergency situations. However these proposed systems act poorly in response time and latency factors which is a critical parameter in emergency condition.

We proposed a secure emergency healthcare data sharing system (SEHDaS), which in compared to the mentioned studies and other available state-of-the-art researches, our proposed system utilizes hyperledger sawtooth, based on permissioned blockchain, which is a perfect platform for solving the PHR access challenge in emergency cases at a minimum time by preserving the patient's privacy. We proved that deploying the unique properties of hyperledger sawtooth platform would result in an efficient and secure system. So the main contribution of this study is introducing a new framework for guaranteeing a private, secure and yet a quick access to PHRs in emergency conditions by minimizing the latency and response time.

1.2 Background study

1.2.1 Blockchain

Blockchain technology originally has been introduced as an infrastructure for cryptocurrencies in 2008 [9]. A blockchain is a specific type of distributed ledger which is independent from any server controlling. So this technology's infrastructure is a peer-to-peer decentralized network [10], which is shared among different parties in a system that records transactions. The main features of this technology are decentralization, reliability, and immutability, which are required for managing

medical records [11]. The framework of blockchain technology is distributed in which we have a real-time sharing of captured data among trusted parties. In all transactions, each block holds a timestamp, transaction data, and a hash value as a pointer to the previous block [12]. Another characteristic of blockchain is transparency which makes the system autonomous and eliminates the need for any intermediary or third-party [13]. In blockchain, when a node receives the message it checks for accuracy of the message, if it is confirmed it will be stored in a separate block, then the consensus algorithm is applied for confirming the stored data in blocks; this action is called "Proof-of-work (PoW)". When the algorithm is finalized a new block joins into the chain. This process continues until it gets all network node's admission on the chain.

Blockchain can be called as a trust protocol as well. This protocol preserves the integrity of its transactions. And this integrity is not met through intermediaries or powerful servers, but it is met through a massive collaboration among distributed ledgers in the network plus cryptography and a clever written code which is called smart contract. These ledgers are immutable and public and they keep a record of the data which is shared and accessible by all network participants. There is cryptographic hash key that binds each block to the proceeding block to create a group of blocks. For a new block to be connected to the chain that block needs to be validated and all blocks must reach to a consensus by a consensus protocol [14]. By this approach non of the blocks and so no previously added data to the ledger can be altered, because alteration will break the consensus among network nodes. Blockchain technology is established with the connection of three different existing technologies including the distributed ledger [15], consensus mechanisms [16], and cryptographic algorithms [17]. [18] has formulated the blockchain as follow:

$$Blockchain = \int (DL, CP, C) \tag{1}$$

- "DL" parameter in this formula stands for Distributed Ledger, the foundational peer-to-peer technology of blockchain which provides us a fully

decentralized network that contributes to a more flawless system design by decreasing the probability of single point of failure in the system, as we pointed out before in the distributed system designs all nodes keep the copy record of the ledger synchronously, so there is no chance of loosing or altering data due to malicious node behaviors.

- "CP" parameter is the Consensus Protocols which are applied and utilized in all types of blockchain by participants for transaction validation, generating new blocks and finally ledger updating verification.

- "C" parameter stands for Cryptography technology which is a crucial factor for authentication and authorization processes in the blockchain.

Based on the technology combinations of blockchain, it eliminates the central point of attacks and failures. It delivers a 'secure by design' approach in which security and privacy is considered a fundamental target in its architecture.

1.2.2 Blockchain Types

Basically, blockchain is categorized into two groups: public permission-less and private permissioned blockchain. However, we have some other variations too, such as consortium and hybrid blockchain. Based on every system's requirement each one of these can be selected [19], but they all have some features in common, and that is each blockchain operates on a P2P network and has a cluster of nodes in which each node keeps a copy of the shared ledger and updates it promptly. Nodes has the capability to verify transactions, initiate or receive any process in the system independently [20]. Some detailed explanation of blockchain types are discussed as follows [10].

- Public Blockchain: This is a permission-less distributed ledger system. That means anyone can join the network and be an authorized node in the system. Public blockchain is secure only if the clients follow the security conventions.

Examples of this type of blockchain are: Bitcoin (BTC) and Ethereum (ETH).

- Private Blockchain: This type of blockchain is also called permissioned blockchain which operates only in a closed network. Private blockchain's usage is very similar to public blockchain, but its network is much smaller and restrictive. Examples of this type of blockchain are: Ripple (XRP), Multichain and Hyperledger (Fabric, Sawtooth), Corda, etc.

- Consortium Blockchain: Consortium blockchain has semi-decentralized framework. In contrast to private blockchain, here there is more than one organization managing the network. And each one can act as an independent node. Examples of consortium blockchain can be: Quorum, Energy Web Foundation, R3, etc.

- Hybrid Blockchain: This blockchain is created by merging private and public blockchain. It also integrates the features of both types of blockchain in a way that user can have both a private permission-based system along with a public permission-less system. Employing this network helps users to have control over who gets access to which data in the blockchain. Only partial data is allowed to be publicly available, but the rest of records are kept confidential using the private network. So, this type of blockchain can communicate with outer world easily while preserving network's privacy. Hybrid blockchain has gained a vast usage domain, Dragonchain can be a good example of a hybrid blockchain.

Most of the research studies on applying blockchain in the healthcare domain have primarily focused on utilizing permission-less Ethereum blockchain technology, but this type of blockchain has some downsides such that it consumes heavy energy, it is very restricted in scalability, and the network throughput is very low. For fulfilling these gaps there is a crucial need for a much scalable, fault-tolerant, timely managed, secure, and private blockchain to meet the demands of the healthcare

systems. Based on the researches done so far, a permissioned blockchain has proved to be more compatible for healthcare data management with the results of reducing the computational overhead and so lowering energy consumption, by meeting the security and privacy factors much efficiently in compared with Ethereum blockchain. Because permission-less public blockchain is not capable of fulfilling this need. There are some blockchain platforms that are designed to be applied for enterprise use-cases. Hyperledger Fabric and Hyperledger Sawtooth are two of the popular enterprise blockchain platforms, which we are going to provide a short background study of both platforms in this study in section 1.2.4 and 1.2.5.

1.2.3 Blockchain in healthcare

In this section we are discussing how blockchain can meet healthcare requirements. In blockchain based healthcare systems, the user of blockchain syncs up with other nodes in the network. Then a particular server manages the transaction records. Data integrity and provenance is fulfilled by applying cryptographical algorithms. The massive scale of operation of blockchain software makes it almost impossible to break into the framework or other applications which are running on it. Hence, there is no need for a central third party to issue, authenticate and validate ownership of the data. When there is two or more nodes possessing same blocks in their own databases, we can consider them to be in consensus. So, according to the blockchain features and applications it can feasibly meet the healthcare requirements such as security, interoperability, data access and data sharing.

1.2.3.1 Compatible characteristics of blockchain for healthcare system

Here we provided important characteristics of blockchain that makes it suitable for healthcare system [1]:

- Decentralized: This is the main feature of blockchain that gives the system the opportunity to give open access control to anybody associated with the network. Due to this feature there is no single point of failure in blockchain,

this means if a node fails then other nodes are able to continue accessing their data.

- Immutable: When records are stored in blockchain they cannot be modified or deleted subsequently without having the admission of more than 51% of the system users.

- Secure: In blockchain encryption algorithms are utilized to encrypt data in order to give data access only to the authorized participants. Also, data integrity is satisfied from the very beginning process until the end.

- Autonomous: Every block can act independently; means they can alter their own data securely and safely. The execution of distributed transactions are done automatically under user-defined conditions.

- Anonymous: When data transferring takes place between nodes the identity of nodes remains anonymous which makes system more reliable.

- Synchronized: All nodes in blockchain have the same chance of accessing exactly the same data at the same time.

1.2.4 Hyperledger fabric

Hyperledger fabric is a framework that is used for deploying a permissioned blockchain for businesses that can guarantee the transparency and trust among parties of the business. Since this framework is based on permissioned blockchain, it allows only the concerned participants to trigger any transaction in the framework. The main components of Hyperledger fabric are ledger which keeps the history of the transactions to be traceable, channels that are used to gather a group of blockchain participant to have their separate transaction ledger, in other words it enables multilateral transactions, transactions which are the assets and business activities on the network, and finally the main part of Hyperledger fabric is the smart contact which executes the business logics and transactions based on the defined permissions, so smart contract can be considered as one of the core components

in blockchain that is governing the whole transactions. Smart contracts are then stored inside the chaincode to be implemented on the network. Chaincode can contain multiple smart contracts as well. So, chaincode is governing the smart contracts packaging, which should be deployed on the network for the smart contracts to be available for execution.

1.2.5 Hyperledger Sawtooth

Hyperledger Sawtooth is a type of enterprise blockchain for deploying, and running distributed ledgers which provides a digital record (such as asset ownership) that is maintained without a central authority. This platform offers more flexibility in its architecture, and the core system is isolated from application domain. Sawtooth supports parallel transaction execution, that means it can provide scheduled strategy to split the transaction and execute them in parallelized mode, this facet makes system resistant to double-spending problem and so resulting to an improved performance in compared with serial execution. Hyperledger Sawtooth has a modular architecture which give enterprises a high flexibility in choosing the transaction rules, logics and also consensus algorithms according to the specific application requirements. Consensus algorithms in this platform are highly dynamic and they can be modified on a running Sawtooth network, and it supports various consensus algorithms, including Practical Byzantine Fault Tolerance (PBFT) and Proof of Elapsed Time (PoET). PBFT is one of the famous consensus algorithms applied on permissioned blockchain. The brief workflow of PBFT is as follows: the database of the blockchain is partitioned into smaller "shards" to make the data managing easy. At each shard there are multiple blockchain participant, When each shard receives a message a computation is run to reach to a decision. In other words, the participants determine the validity of the message sent to them. Then the agreement is shared by other shards of the network. When all shards reach to a final decision, it is assumed that they reached to a consensus and the new block can be added to the blockchain. PBFT is adoptable to be used in both Hyperledger fabric and sawtooth blockchain. PoET is a consensus mechanism for

permissioned blockchain as well. In PoET a waiting time is generated randomly for each participant in the blockchain. During this waiting time the processor of the miner can go to sleep or switch to other tasks in that time. The node which its waiting time is over first, or wakes up sooner wins the opportunity of adding the new block to the blockchain. Since this waiting time is assigned randomly, this provides equal opportunity between nodes to create the new block. Also focusing of nodes on other tasks in that waiting time increases the efficiency of the blockchain. These features are making PoET a better fit for our case, since timing and latency factor is a critical concern in PHR data accessing during an emergency condition. PoET can fully satisfy this factor by minimum processing time and it's less resource consumption and energy efficiency.

Sawtooth network has three main components that are responsible for the entire network's functionality. More details of these components are discussed in the following part of this section.

- Validator: Validators are the core elements for Sawtooth network functionality. These peer nodes are responsible for harmonizing the overall network, such as transaction processing, communication within other validators in the network, and managing the consensus processes among nodes.

- Transaction Processors: This is one of the main components in Sawtooth platform. A set of business logic and rules are implemented in transaction processors which creates the smart contract. These transactions are validated based on the defined rules and policies called transaction families, and finally the state of the system is updated. Transaction processors are customized based on the requirements of every application. There is a rich variety in choosing the programming languages for deploying transaction processors business logics such as Go, Python, Java, JavaScript, C++ and Rust, which makes the smart contract developing much convenient.

- Rest API: The other component that collaborates in running the Sawtooth

network is REST API. With utilization of this service users can communicate with validators through HTTP and JSON standards.

A brief comparison of Hyperledger fabric and sawtooth key features are provided in the Table 1 for a clear understanding.

Table 1: Comparison of Hyperledger Fabric and Sawtooth framework key features

	Hyperledger Sawtooth	**Hyperledger Fabric**
Network join permission	Permissioned, permission-less	Permissioned
Consensus algorithm	PBFT, PoET, Raft	PBFT, Raft
Smart contract	Transaction families	Chaincode
Language support for smart contract	Javascript, Go, Python, Java, C++, Rust	Java, Javascript, Solidity, Go
Transaction validation	Validators	Endorsing peers
Transaction execution	Parallel	Serial

1.3 Thesis outline

The remainder of the thesis is organized as follows. In Chapter 2 we briefly discussed the state-of-the-art studies done so far on blockchain-based healthcare systems, and compared the research studies by highlighting their strengths and weaknesses. Besides we did another comparative study at the end of this chapter, on most-related studies and our proposed system, considering the privacy, scalabity, immutability, and throughput fulfilment of the blockchain systems applied on healthcare. In Chapter 3, first we discuss the designed system details and its characteristics. Then, we present the proposed system architecture along with the participant details of the implemented network. We finalize this chapter by providing and discussing the designed protocols for the main functionalities of the network. In Chapter 4 we present the performance results of the system by implementing the proposed protocols on hyperledger sawtooth platform and performing the tests with two scenarios of different number of clients and increasing the requested transaction load of the network, for evaluating the throughput and latency performances of the implemented network. We also perform a test for

taking the average response time and memory usage for the detailed transactions of the PHR emergency access, and compared the results with the similar state of the art work. At the end of this chapter we discuss our system's security and privacy fulfilment. Finally, we conclude and summarizes thesis by providing the remarks of our contribution in Chapter 5 .

CHAPTER II

LITERATURE REVIEW

In this chapter we provide a through literature review study on all existing researches done so far on the blockchain based healthcare systems, precisely by focusing on security and privacy aspects of health data records. We also compared all studies at Table 3, to identify the proposed solutions contribution and gaps. Then we end this chapter with providing a comparison study our proposed system and existing most related studies at Table 4, by taking the privacy, scalabity, immutability, and throughput fulfilment into consideration.

2.1 Overview of blockchain-based healthcare frameworks

As we mentioned before, healthcare data is very sensitive, and it is crucial to protect them in means of privacy and security. So, for processing, sharing, storing and handling medical information a secure platform should be designed [21]. Blockchain technology has been proposed and explained by several research studies in recent years for dealing with security and privacy risks, due to its potential to promote better data sharing and management and its assistance in treatment process. [22] have proposed MedRec as a novel system which utilizes blockchain for managing large medical data. It provides easy access to data and feasible patient data sharing. [23] has facilitated a Healthcare system that enables sharing data in a private and auditable manner by considering patient pseudonymity and hiding the identity of the user whose data is shared. A 3-layered architecture is also proposed for respectively storing patient's data in a web or cloud platform. In this architecture there is a cloud middle-ware layer which its task is data fetching from the cloud layer to the next layer, which is blockchain network, by utilizing an API and interacting with smart contracts. A smart contract is a self-executing protocol which is coded and executed inside of a blockchain and it contains terms

and conditions of an agreement between peers. It also consists of rules that helps to keep medical data secure in a way that the data is accessible only to the authorized person by digital signature [24]. The Ethereum platform is utilized for executing smart contracts. [25] has improved security and privacy related issues in biomedical systems by designing a blockchain and distributed ledger based secure scheme. Authors have proved that this system is capable of maximizing the data sharing ratio while minimizing the computation and response time by ensuring the privacy of users. [26] has proved that smart contract can be a potential solution to tackle the challenges of interoperability in healthcare, so that medical data are shared and communicated in a secure way. Ethereum smart contracts is also being used in [27], to design a MedRec prototype which is a proof-of-concept system, and to manage the access to healthcare records. Researchers in [8], have applied smart contract in their unique architecture, which is independent of any previously designed blockchain platforms, in this system they could guarantee the integrity in the records of EHR systems, and they enhanced the interoperability of the system. Smart contracts are considered to be dynamic in nature, so they are best fit to be employed in designing access control policies for example, authors in [28], proposed an Enhanced Bell–LaPadula model to classify the healthcare data and access control policies and so implement a scalable and secure healthcare network. Some hybrid methods are recently proposed by researchers in which they combine the power of blockchain technology with other existing technologies to fulfil the security and privacy gaps of healthcare system. [29] has utilized Blockchain technology along with machine learning to provide a secure framework in order to store and maintain patient's data such as providing accurate and authentic health records. In this system machine learning is applied to detect any anomality in the newly generated data of patients. [30] provides an approach for a private and scalable EMR storage by the incorporation of Public Key Infrastructure (PKI) with blockchain technology. [17] developed a key management scheme for healthcare blockchain to enhance security and privacy in healthcare system. They

integrated Body Sensor Network (BSN) with health blockchain and created an efficient recovery strategy for managing the keys. Another study related to key management by blockchain is done recently in [31], researchers designed a new sharing scheme based on blockchain for privacy protection and enhancing access control management process through symmetric and attribute-based encryption. Healthcare data are dynamic, and they are changing constantly and there should be a system that can adopt to the metadata alteration efficiently. [32] proposed a scheme called MedChain that integrates blockchain, digest chain and P2P network aspects to provide an efficient system for data sharing in healthcare.

[33] has also utilized blockchain to propose a framework for sensitive medical data sharing which is stored in cloud. The blockchain type used in this system is permissioned blockchain that gives access only to the verified users. Putting encrypted medical data into blockchain is first proposed by [35], an App was also designed to this purpose. Researchers in [36], have proposed the requirements along with their potential solution for having an efficient healthcare system with combining blockchain and cryptographic algorithms. And the details for applied cryptographic techniques are discussed at [34]. For example, ARX Symmetric Encryption Algorithm has been applied in order to have an efficient encryption of the sensitive data of blockchain. Digital Signature is used in authentication process and for keeping the users anonymous and protecting the privacy of users Digital Ring Signature is applied.

A three-layered architecture is proposed in [37], for ensuring a private and secure data management in healthcare IoT devices. The layers include Sensing, NEAR processing, and FAR processing layer. Sensing layer is for patient data acquisition and transmitting to IoT devices. NEAR processing layer is used for sensing the IoT devices, and FAR processing layer includes servers for cloud and high computations. Digital Ring Signature and SSS is applied in this system to fulfil security requirements in healthcare agent's communication process. Authors of [38], set up a shared key for providing a private access to medical data to be

Table 2: Comparison of the current studies on blockchain-based healthcare system security and privacy

Ref	Concept	Data Type	Blockchain Type	Contribution	Gap
[11]	Smart contracts for secure remote patient monitoring	PHI	Public Ethereum blockchain	Patient's data validation using PBFT consensus mechanism	Lack of verification step in consensus mechanism
[33]	Blockchain framework for sharing data of EMRs system in cloud environment	EMR	Permissioned blockchain	Guaranteeing user anonymity using identity-based authentication along with key agreement protocol	Communication and authentication protocols are not fully investigated
[12]	Providing e-health system based on blockchain for secure data sharing	PHI	Consortium blockchain	Secure sharing of PHR using cryptographic primitives for improving the diagnosis	Scalability in private blockchain is not addressed
[29]	Timely remote data fetching by considering the data privacy and security of patients	EMR	Consortium blockchain	Protecting the system from stakeholder's data manipulation and leakage while providing a timely mannered and secured access to data	Security and cryptographic details are not discussed
[22]	Sharing and managing medical data using blockchain	EMR	Public Blockchain	Guaranteeing privacy by combining access control protocol with encryption algorithms	No key replacement capability, and legal issues are not addressed
[23]	Blockchain-based system for sharing the medical data	Healthcare data	Consortium Blockchain	Enables private data sharing and enhances security by access permission management	Energy consumption is high due to the usage of PoW consensus

[26]	Software patterns for addressing interoperability issues in healthcare apps	Medical Records	Ethereum Blockchain	Maximizing sharing of resources and application scalability by applying foundational software patterns	Privacy and security concerns are not investigated enough
[27]	Blockchain-based record management system for handling EHR	EHR	Ethereum Blockchain	Secure and inter-operable medical record access system	Cryptographic techniques are not discussed
[18]	Scalable and efficient blockchain architecture to meet healthcare data management issues	PHR	Permissioned Blockchain	The architecture reduces the computational and communication delay in a more scalable environment	Security and privacy performance in this architecture is not fully investigated
[25]	Enhanced blockchain and distributed ledger based secure scheme for secure data sharing	EHR	Undefined	Classification-based authentication reduced the computation time and provides a secure and private data sharing	Scalablity is not addressed
[8]	A new scalable and adoptable architecture for dealing with interoperability	EHR	Private Blockchain	Guarantees the integrity of healthcare records with a multiple access system for block creation	Consensus mechanism detail is not fully investigated
[32]	Efficient and secure data sharing scheme	EHR	Consortium Blockchain	Secure data sharing between medical laboratories and institutions	Proposed system is not fully automated
[34]	Data preservation system for Medical Data	Medical Data Records	Ethereum Blockchain	Ensuring verifiability of stored data while protecting users' privacy	Authentication protocol is not fully investigated

used in the process of diagnosis and treatment. This shared key is created using sibling intractable function families (SIFF). The key is utilized for encrypting and storing data in a blockchain for protecting data. The evaluation of proposed system proves to have a good efficiency in aspects of integrity, availability, and privacy of medical data. In a most recent study [39], a Multi-Modal Secure Data Dissemination Framework is proposed for data access control management in IoMT devices. In this scheme data is encrypted in blockchain and decrypted only by private key of patient. By employing blockchain in this framework security and privacy is enhanced significantly compared to previously proposed methods. Due to the ever-growing amount of medical data in the world, security and privacy issues of health data have become a major research topic for academicians and security engineers. In Table 3 we carried out a detailed summary of the current blockchain-based healthcare systems proposed by researchers so far along with their contribution and gaps to form a point of comparison between methods.

In [40] authors proposed an architecture for accessing patient medical history in emergency cases. They utilized a cloud environment in which the patients store a summarized document of their own medical data and history with a state owned unique reference number. And medical centers use each patient's unique biometric identification data such as fingerprint to access and retrieve the data for medical treatment. Another solution is proposed by defining the confidentiality level of access to PHR by categorizing the data into three levels: secure, restricted and exclusive [41]. In this study an emergency server is defined in this system to control and manage the accesses. The authors assumed some emergency units to be trustworthy that authenticates the emergency staff to authorize the access to PHR data. [42] proposed a system for secure access in medical data by emergency medical centers by storing medical records in a cloud server in encrypted way which can be decrypted only by specific emergency centers without revealing the secret key. Although these solutions may seem to be practical but they rely on a centralized approach which makes system prone to single point of failure, and

these kinds of failures in PHR access are not acceptable due to the critical and life threatening cases in emergency conditions.This raised the demand for more decentralized approach to be implemented in this area. In recent years blockchain technology is proposed as an applicable method for managing healthcare data, for example [22] introduced MedRec which provides easy access to data and feasible patient data sharing utilizing Ethereum blockchain, however making use of public blockchain in this approach leaves a gap for security and privacy issues of the shared data. To overcome this challenge researchers came up with the solution of applying the healthcare system on a private blockchain rather than public to preserve the data privacy and security. Private blockchain is proved to guarantee the security and privacy of medical data in various research studies such as [23]. In this study researchers have introduced a healthcare system based on Consortium blockchain which is capable of private and auditable data sharing by preserving the identity of the patient data pseudonymity, the proposed method enabled a private PHR data sharing between patients and medical centers however security aspects are not fully investigated in this study. [12] proposed a secure and private mechanism for Personal Health Information (PHI) sharing based on a private and consortium blockchain, in which they utilized the private blockchain for storing the PHI and the consortium is responsible for keeping the secure indexes of the stored records. All data in this scheme including the PHI and patient's identity are encrypted by pubic-key encryption using the keyword search, to fulfill the data security challenges this mechanism has proved to improve the Diagnosis in treatment processes for e-health systems but scalabilty is not addressed in this research. To fulfil the scability gap [28] proposed an Enhanced Bell–LaPadula model to classify the healthcare data and access control policies and so implement a scalable and secure healthcare network by dynamic access control policies. Similarly, [26] has proposed a mechanism for addressing interoperability issues and enhancing the communication between users and medical applications by applying Software patterns on Ethereum blockchain, which proved to improve the sharing of resources

and application scalability, however the public blockchain utilization leaves a gap for Privacy and security concerns in this system. We did a comparative study between our proposed system in this study and the most recent proposed blockchain-based healthcare systems, the details are provided in Table 3. Some of these studies so far focused on providing access to medical records in emergency conditions, and some focused on applying blockchain technology for secure medical data sharing and access by utilizing smart contract characteristic of the blockchain and in some cases applying cryptographic techniques. Yet still an efficient system is required for a quick, secure and private access in emergency conditions. In previously proposed traditional systems for emergency access the security and privacy factors were not met because they were utilizing central authority, authors in [43] have investigated on this challenge by proposing a framework based on a private hyperledger fabric blockchain. In this study they proposed a system called EACMS for managing the emergency access to PHR data in a hospital, however this system may act poorly regarding scalability and flexibility. Also the serial execution of the transactions in their proposed system may require longer processing time, and it increases the workload of the chain and results a high resource consumption. The security and privacy aspects are not fully investigated in this system, and these issues leave a gap for a more private and secure system for PHR sharing in emergency cases. We brought all state-of-the-art studies together, and proposed a new system and mechanism called SEHDaS based on Hyperledger Sawtooth to meet these requirements. We proved that our proposed scheme guarantees the privacy and security in emergency PHR data sharing by minimizing the response time which is a critical factor in this case.

Table 3: Comparative analysis of SeHDaS vs the state-of-the-art blockchain-based healthcare systems

Ref	**Scalability**	**Throughput**	**Privacy**	**Immutability**	**Comments**
SeHDas (Our proposed system)	✓	✓	✓	✓	Secure and scalable emergency PHR data access system with higher throughput and lower latency
[12]	✗	No info	✓	✓	Diagnosis in e-health is improved but scalability is not investigated
[44]	✓	No info	✗	✓	Access control to the data and data auditing is not met and so system doesn't ensure the privacy
[23]	✗	No info	✓	✓	Shares medical data between research centers, but system's security needs to be more investigates
[22]	✗	No info	✗	✓	Utilizing public ethereum blockchain and PoW consensus results on high energy consumption and so inefficiency of system
[45]	✓	No info	✓	✓	Energy consumption is high due to the usage of PoW consensus
[38]	✗	No info	✓	✓	Privacy and integrity in data management are met to the fullest but scalability is not addressed in this system

CHAPTER III

SYSTEM DESIGN AND ARCHITECTURE

The conventional healthcare systems are patient-centric, and for every medical process it is only the patient who issues the access authorization to his/her medical reports, this approach satisfies the security and privacy issues regarding personal data. Yet still there is a chance of an emergency case happening, in which patient cannot take any action regarding access control and data sharing for a proper and fast treatment process. When the patient loses his/her consciousness, its hard and almost impossible for him to be able to issue any access authorization or share health related data with a medical service provider which may lead to a delay in diagnosis and treatment process and put a patient's life in danger. And this emergency scenario may happen anywhere and anytime. So we need to have an approach that guarantees PHR data sharing when there is an emergency case in a secure manner without compromising patient's privacy.

We considered fulfilling this gap by applying the permissoned blockchain hyperledger Sawtooth technology network for creating a framework of secure PHR data sharing. To implement our solution, we got a patient's health record sample data-set as an example [46] and restructured them to fulfill our proposed system's criteria, and run the proposed system's implementation on these data. Patient has full control over these data such that patient can preset the rules of what type of data and which identity can have access to them. This data-set contains a wide range of patient health records such as patient's personal identity information, disease and diagnosis history, laboratory results, medications, along with treatment dates and clinics information.

In traditional systems if emergency condition happens these data cannot be accessed without patient's permission. In our system patient defines one or several

trusted party identity along with the data types which patient wishes to share with them in her/his PHR records. Patient can also define weather the trusted-party has the right to share records with a third-party such as emergency doctor if an emergency case happens. After recovery from a treatment patient can trace to see who and to which part of data an emergency doctor or a trusted-party has accessed and whether the access identity belongs to the defined identities inside PHR records, although the access permission is given only to the registered users in the network. Every data access is stored on the blockchain and records are trackable. By deploying a permissioned blockchain technology our proposed system provides a secure and private access to an emergency doctor in a short period of time and it guarantees quick PHR data fetching from health records.

3.1 System design and architecture

We proposed a new framework for addressing the emergency PHR data access by defining the access rules implemented on Hyperledger sawtooth network. We utilized both off-chain and on-chain storage mechanisms in our system to provide an efficient and secure data sharing system with minimum cost. On-chain storage include all data which are stored in the blockchain and they are available to all blockchain members. In our case, since we utilized a permissioned hyperledger blockchain, on-chain data are only available to the registered members of the blockchain network. For on-chain transactions to be executed all blockchain nodes and participants must validate and verify the transactions to reach to a consensus in order for that to be considered valid, and then all completed transactions history ledger should be updated to be traceable, so that they cannot be changed or reversed back. For all of these processes the system requires more time and resource consumption. However for some kinds of data-sets and contracts in blockchain based systems, that doesn't have any role in the transaction processes, the optimum solution is to employ off-chain storage methodology.

According to [47], off-chain data storage is a method for storing any data that

is too large to be stored in the blockchain properly, and they may also require modification which makes it costly to be altered if they are stored on-chain, these data are structured or unstructured data in the form of files, contracts and documents which cannot be reside in blockchain. Off-chain storage mechanism decreases the load handling of big datasets in blockchain. Nevertheless, on-chain transactions are significantly slower because of the need for other node's confirmation to be completed. In addition to guaranteeing faster transactions, off-chain storage brings a huge value to our system by increasing the privacy and security, since transfers are not visible to the other participants of the blockchain. In our implemented system the goal is to introduce a system with better performances considering response time, scalability and security. And scalability factor for a blockchain based system can be satisfied with off-chain storage which can easily store any type of real sensitive medical data. And as mentioned before on-chain stored data are non-negotiable in terms of any alteration. In our implemented system we utilized off-chain storage method for storing the encrypted PHR along with access rules to deliver a faster and more secure data sharing system for healthcare. In SEHDaS system, off-chain storage does not certainly mean nothing is on the blockchain, but it implies that PHR data are not available for all blockchain participants. We utilized symmetric key cryptography for PHR encryption before these data are uploaded into off-chain database through the blockchain.

Each PHR data file stored on off-chain database belongs to a single patient and contains health records of that patient. Blockchain network plays a vital role for keeping unauthorized parties away from accessing data and it contributes for data sharing only with authorized parties who have access permission based on the patient rules. Emergency condition can happen anywhere and anytime, so we need a system which is distributed and decentralized that paves the way for easy and quick access to patient's PHR for an emergency doctors regardless of time and place, considering this challenge blockchain is the only private, secure and decentralized technology that can meet these requirements. Also blockchain

provides a thorough history log of access records and patient can track these records through the blockchain ledger.

We have developed a transaction processor which stores user's identities, and access-sharing rules for encrypted data. Transaction processor in our system manages authorization and data fetching from off-chain database process. Transactions are validated based on the rules defined by patient through transaction families and the state of the data on blockchain is updated. Operating our proposed framework based on transaction processor of the blockchain ledger ensures the system to be protected, auditable and verifiable. Since blockchain is an immutable technology and the transaction processors are applied on-chain, this guarantees that the records were created and stored using the same transaction processor which is used for creating the original block, and the same global states are utilized for executing all transactions, so records cannot be modified or deleted.

In our proposed architecture Fig. 1 the system users are classified into three main categories, we did this classification based on access level permissions to the PHR data. The actors of our system are a Patient, a Trusted-party and an Emergency doctor. Other components of our system architecture are an off-chain database, and a hyperledger sawtooth network containing the validator including the transaction processor and consensus process, a REST API and a ledger. Detailed explanation of the main components are provided as below.

- Patient or data-owner is the owner of the PHR data who has control over data. Patient authorizes access to his/her data partially or completely and defines rules as a group of trusted parties and specifies partial access to these parties for emergency cases. These rules and conditions are defined along with PHR data and they are stored and processed as transaction families and processors. This way patient is transferring partial-ownership of PHR data to the trusted-party so he/she can make decisions of data sharing with emergency staff on behalf of patient when an emergency condition happens.

- Trusted-party can be one or group of users including a doctor, a friend or a

family member that patient has listed and defined in her trusted emergency lists. This party can have different level of access permissions to PHR data. Each of these parties can have a full access or partial access to the different data types that the patient is pre-specifying.

- Emergency doctor is who issues access request to PHR, only in emergency conditions.

- Off-chain database is used in our proposed system to store the encrypted PHR and patient access permission rules. For this purpose, we have utilized MongoDB which is a NoSQL database and it is practical for managing large distributed datasets.

- Hyperledger Sawtooth REST API is used to access the users of the system and manage client communications with validator using HTTP/JSON standards. REST API acts as an intermediary to pass the access requests to the validator, these requests are then evaluated and authorized based on a signature verification or schemes defined in transaction processor.

- Transaction processor manages the transaction families and holds data sharing metadata and rules that control which user can access the data. Request is sent through the API to the transaction processor and then it is validated and processed based on patient-defined access control policies. Transaction processor in sawtooth has the role of chaincode in hyperledger fabric, and the transaction families are the smart contract of sawtooth network.

- In SEHDaS system Proof of Elapsed Time (PoET) consensus algorithm is applied for all transactions accuracy based on consensus-specific functionalities, such that PoET has the capability to support our large distributed network with the least resource consumption and guarantees a more energy efficient consensus procedure in our system, since before updating the ledger state by transaction processors all network nodes must reach to a consensus and

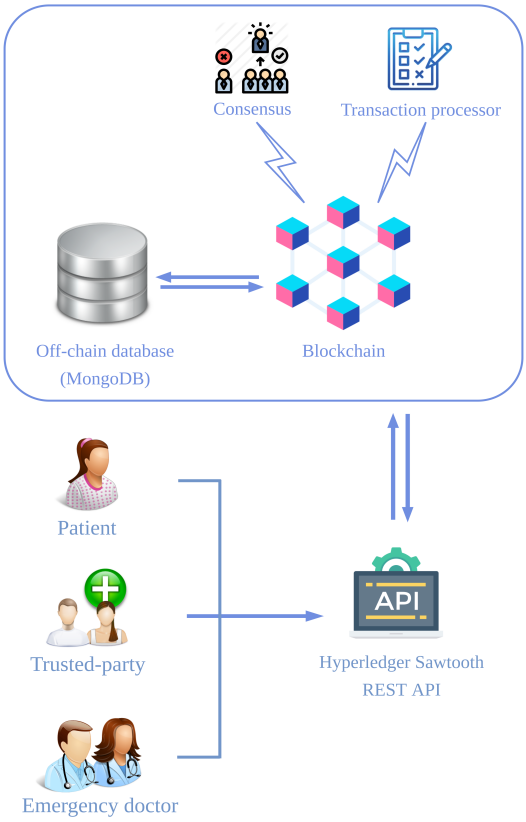

Figure 1: Architecture of the proposed system

agreement for processing an access request and PHR sharing transaction.

3.2 System workflow protocols

SEHDaS system has two main components a permissioned blockchain, Hyperledger Sawtooth for managing the transactions in a distributed manner, and an off-chain database for storing encrypted PHR data. Flow diagrams for the implemented protocols are provided in the following of this section which shows the protocols for secure emergency access to patient's health records. First all users join the network using joining protocol 3.2.1, then there is a key agreement protocol between Patient and Trusted-party used for encrypting the partial data of patient 3.2.2, and finally Trusted-party issues access to partial data using partial data access protocol 3.2.3. We assumed that all users get their private and public keys from the hospital when they are registering to the system.

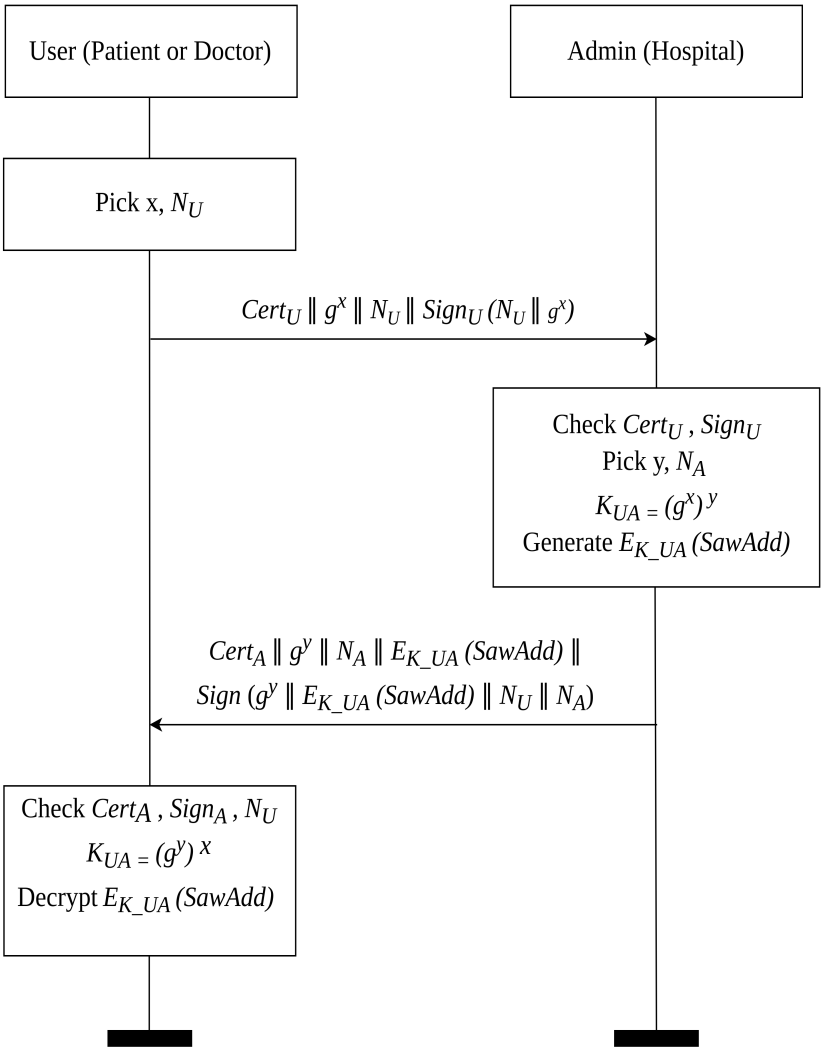

Figure 2: User joining protocol

3.2.1 User joining protocol

We have utilized a private permissioned blockchain in SEHDaS. Therefore, in the very first step in our protocol, users including patient, doctor (trusted-party), and emergency doctor must join as a valid and authorized user to the blockchain network before they can initialize and trigger any transaction. Fig. 2 depicts user joining protocol into the permissioned private blockchain network. In this protocol users join to blockchain by receiving their Sawtooth Address as $SawAdd$ from the Admin which in our case we assumed as Hospital. In order to send the sawtooth address securely Admin encrypts the assigned sawtooth address. For generating the encryption keys Diffie-Hellman key exchange algorithm is applied. Such that, there

is a predefined prime number as modulo, p, and accordingly the base number g, which is the primitive root modulo of p. User chooses a secret integer value x and also picks a random nonce N_U. Then he sends his certificate, $g^x \ mod \ p$, and the fresh nonce N_U and signature of the message to the Admin, which in our case we assumed the hospital. After receiving the message, Admin checks the correlation of the certificate and signature. After the check is successful, Admin chooses a secret value y, and picks a random nonce N_A. Then he calculates the shared secret key $K_{UA} = (g^{x)y} \ mod \ p$, and encrypts the sawtooth address of the user $SawAdd$ using shared secret key. He then sends back his fresh nonce, $g^y \ mod \ p$, certificate, and encrypted sawtooth address $E_{K\_UA}(SawAdd)$ along with the signature of his message to the User. User upon receiving the message from Admin checks the correlation of the Admin's certificate with the provided signature and the returned nonce, afterwards he calculates the shared secret key $K_{UA} = (g^{y)x} \ mod \ p$, then he uses this shared key to decrypt the received encrypted sawtooth address. Therefore User joins the blockchain network successfully and receives his sawtooth address.

3.2.2 Patient and trusted party key agreement protocol

After joining the network, Patient encrypts his/her PHR data with a symmetric key and uploads the encrypted data to off-chain database, MongoDB. In the next protocol, patient and trusted-party must agree on a shared secret key which is used by patient to encrypt the PHR before uploading it to the off-chain database MongoDB. As it is depicted in Fig. 3, The protocol is based on the Diffie-Hellman algorithm as well. We assumed that, there is a predefined prime number as modulo, p, and the base number g, which is the primitive root modulo of p. Patient picks a nonce N_P and chooses a secret value a, then he sends his certificate, $g^a \ mod \ p$, and fresh randomly chosen nonce N_P, along with the signature of his message to the Trusted-party. Once receiving the message by trusted-party, it is checked for the correspondence of the signature and certificate. After the successful check, trusted-party picks his secret value b and picks a random nonce N_T. He then uses the chosen secret value b to calculate the shared secret key $K_{PT} = (g^{a)^b} \ mod \ p$. In the next

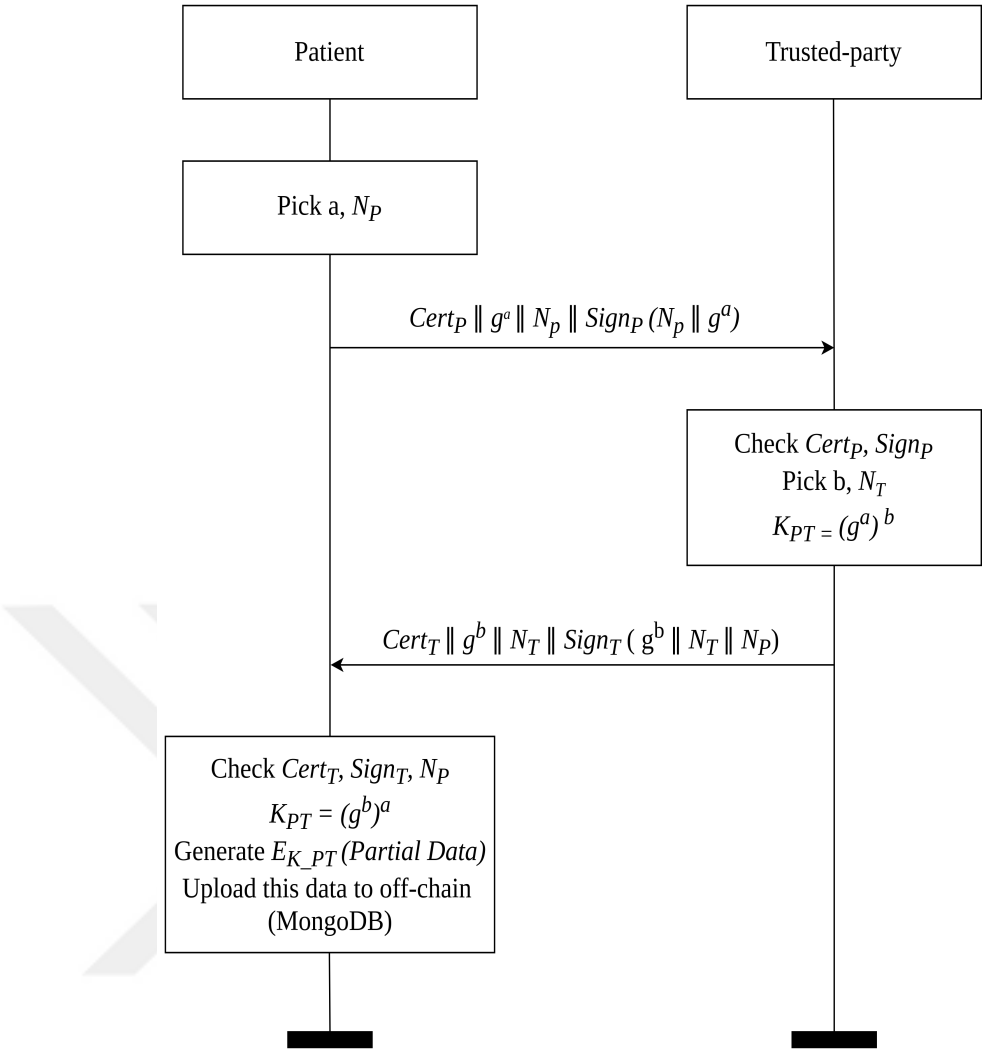

Figure 3: Patient and trusted party key agreement protocol

step trusted-party responds back to the patient by sending his certificate, $g^b \ mod \ p$, and chosen nonce, along with the signature of his message and patient's nonce N_P. Patient upon receiving the message checks the correspondence of the received certificate and the provided signature, along with the returned nonce. If the check is successfully passed, patient calculates the shared secret key, $K_{PT} = (g^{b)^a} \ mod \ p$. When the shared secret key is generated uses this key to encrypt his partial PHR data that he intends to share with the trusted-party. After the encrypted partial PHR is generated $E_{K\_PT}(PartialData)$, the data are uploaded to the off-chain database (MongoDB) for future access.

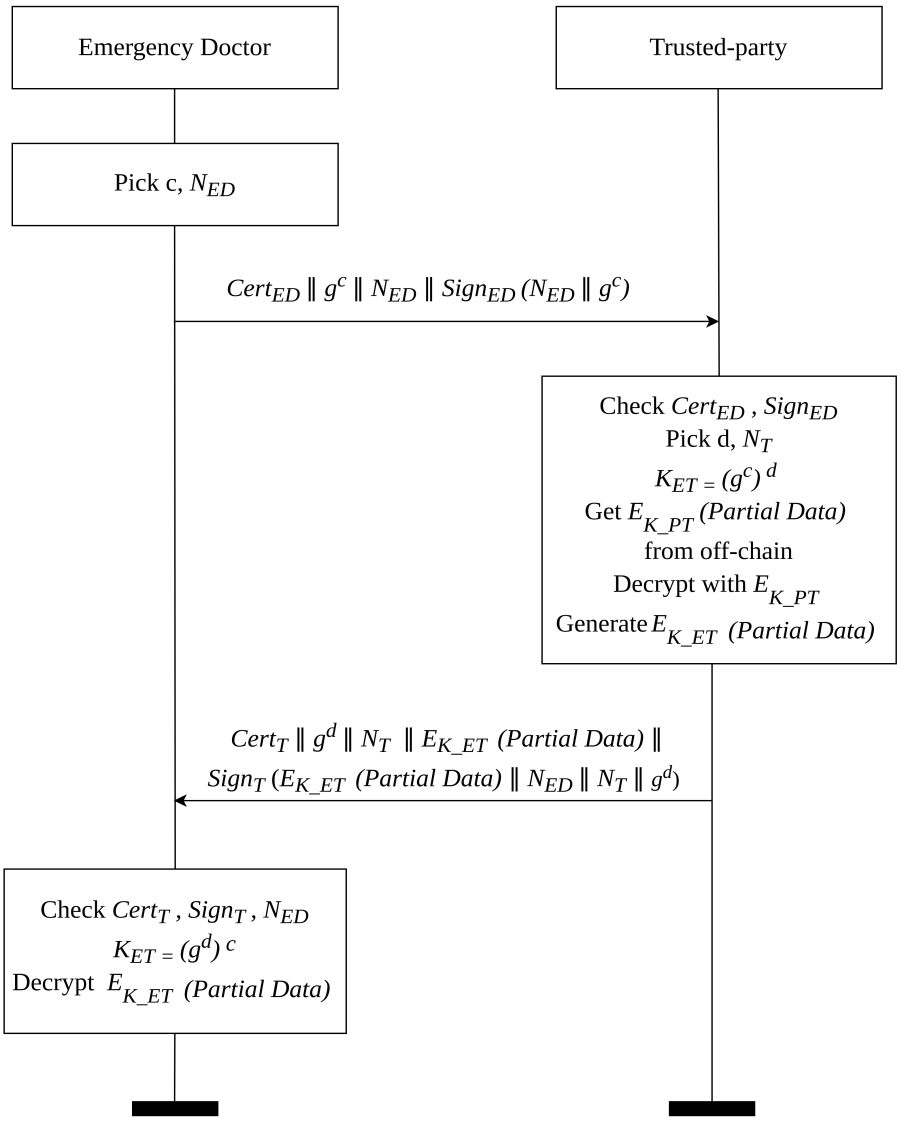

Figure 4: Partial data access protocol for emergency condition

3.2.3 Partial data access protocol for emergency condition

In this protocol we considered the emergency case where the patient or data owner is unconscious and there is need for an intermediary (trusted-party in SEHDaS), to issue the access permission and share the partial PHR data with the Emergency doctor considering the privacy of the patient and security of the data to be shared. Fig.4 shows the details of this protocol, we utilized the Diffie-Hellman algorithm to create a shared secret key between Emergency doctor and Trusted-party. Like previous protocols, there is a predefined prime number as modulo, p, and the primitive root modulo of p as base number. Whensoever the Emergency

doctor (ED) demands to access the patient's PHR data, he chooses a secret value c and picks a random fresh nonce N_{ED}. He sends his certificate, $g^c \ mod \ p$, and the nonce along with the signature of the message to the Trusted-party. Once trusted party receives the message he first checks the Emergency doctor's certificate and provided signature's correspondence. Immediately after the check is successful, Trusted-party chooses his secret value d and his random nonce N_T. The selected secret value is then used to generate the shared secret key $K_{ET} = (g^{c)^d} \ mod \ p$ which will be utilized to encrypt the requested data. However, before Trusted-party can encrypt the partial data, he gets the partial data from the off-chain database, and uses the shared secret key of K_{PT}, which was created in key agreement protocol 3.2.2, to decrypt the partial PHR data. Next, he uses the secret shared key K_{ET} to encrypt the partial PHR data. When the data is encrypted, Trusted-party sends back the encrypted data $E_{K\_ET}(PartialData)$, his certificate $Cert_T$, his fresh nonce N_T together with the signature of his message and nonce of Emergency doctor and Trusted-Party. Once Emergency doctor receives the credentials, he first passes the check procedure of provided signature and certificate correspondence and returned nonce. If the check is successful, Emergency doctor starts to calculate the shared secret key $K_{ET} = (g^{d)^c} \ mod \ p$, and then uses this shared key to decrypt the received encrypted partial PHR data. When this protocol terminates, a temporary and partial access to patient's PHR data is provided for the emergency doctor.

In SEHDaS we have defined a time threshold for data-access in emergency conditions, based on this threshold Emergency doctor can have access to PHR data for a limited amount of time which starts from the beginning of the access. When the access time expires the data are no longer available to the Emergency doctor.

A workflow of proposed protocols for SEHDaS is depicted in Fig. 5 for a secure and private PHR data sharing among Patient, Trusted-party and Emergency doctor. In the very first step, users (Patient and Trusted Party) join protocol by Admin authorization and receive their sawtooth address, (protocol 3.2.1). After joining,

Patient encrypts his/her data by a symmetric encryption algorithm and uploads the encrypted PHR data to MongoDB. In the third step Patient and Trusted-party agree on a secret shared key which Patient uses that key to encrypt the partial PHR data that he/she wishes to share with the Trusted-party. Later, the encrypted partial data are stored in MongoDB, (protocol 3.2.2. Whenever an emergency case happens and Emergency doctor requests to access PHR data, Trusted party fetches partial data from MongoDB and after decrypting and encrypting by a shared secret key with Emergency doctor, issues access to the partial PHR data with the protocol 3.2.3.

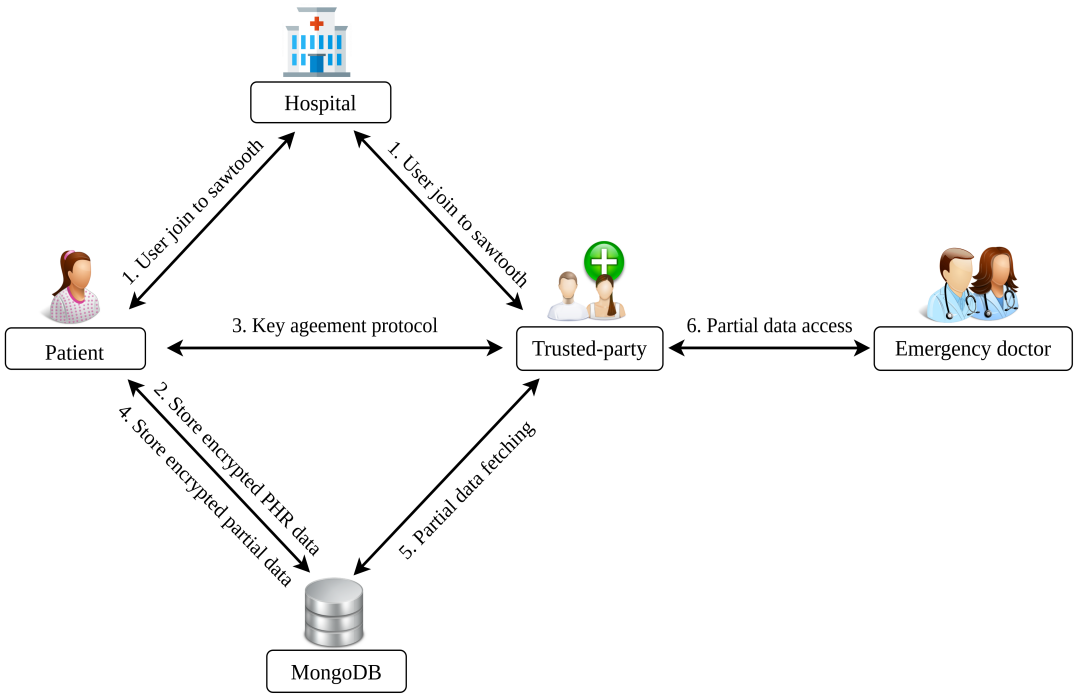

Figure 5: Workflow diagram of SEHDaS protocols for secure PHR access

CHAPTER IV

PERFORMANCE ANALYSIS AND EXPERIMENTAL RESULTS

4.1 Experimental environment and setup

We assessed our framework by developing and implementing a hyperledger sawtooth network customized for our system's protocols. The back-end is created using Golang and Hyperledger Sawtooth SDK. It is one of the top SDKs regarding maturity level and API stability for client and transaction processor managing. For testing the performance of SEHDaS we ran the network on a local personal computer with Ubuntu 18.04.6 LTS, 8 GB RAM and 1.80 GHz COREi7 8th generation CPU. All components of the network are deployed in the Docker containers and for that we used Docker-engine version 20.10.8 as virtualization tool, and Docker-compose version 1.29.2 for managing the docker containers. Patient's PHR are entered as CSV file and after they are encrypted and uploaded in MongoDB. We performed a benchmark testing on our implemented system, and evaluated the performance of SEHDaS in terms of the parameters that can impact the system performance in analysis procedure, such as latency and throughput. These parameters are evaluated considering two scenarios, one for the case when number of clients increase and the second one for the case when the send rate increases. We kept the client number constant at 1 client and increased the send rate from 5 up to 50 by 5 intervals. Then we measured the above mentioned metrics over increasing send rate. We repeated the same experiment for the 3 and 5 number of clients as well. Furthermore, the tests are performed for the proposed protocols of SEHDaS; User joining protocol, key agreement protocol and partial data access protocol. In addition to these experiments, we performed a test for observing the total execution and response time for detailed data replication and ledger update in each process,

and the total memory usage in all protocols, and finally we compared SEHDaS results with the similar work EACMs [43].

4.2 Performance Results

4.2.1 User joining performance results

We performed the experiments and testing through the implemented hyperledger sawtooth for the joining process of users when they send request for entering the permissioned blockchain. We performed these tests for two scenarios, incremental number of clients and increasing send rate. In this experiment we analyzed the throughput and latency for 1, 2 and 5 clients, and for each case we increased the

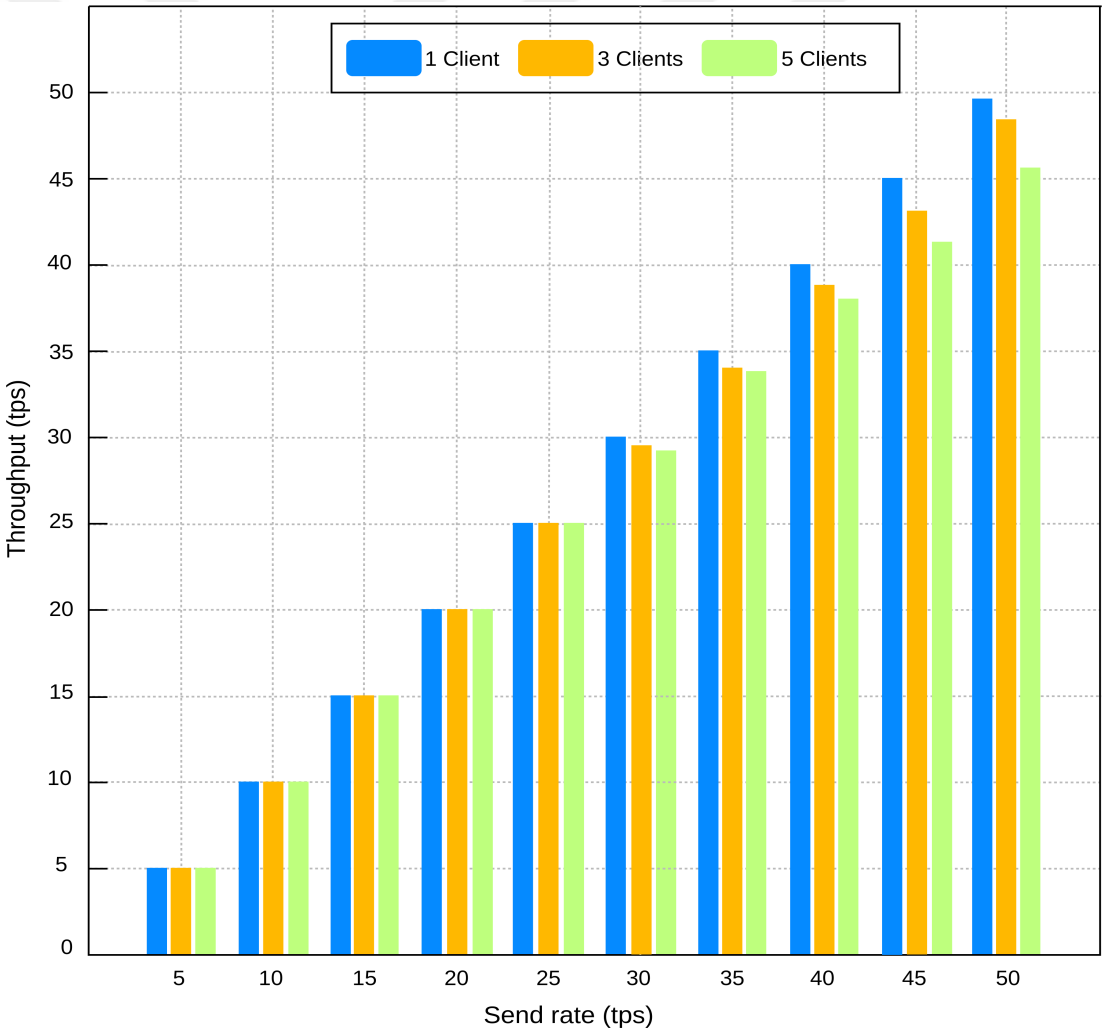

Figure 6: User joining throughput for increasing number of clients, over increasing send rate

send rate to experiment the system's throughput over these factors.

Figure 6 shows the plot of average throughput of transactions for varying clients (1, 3, 5), which is run on increasing send rate of transactions starting from 5 tps to 50 tps. As shown in the figure when the client number increase to 1 ,3 and 5, the throughput is increasing linearly by increasing send rate. However, when the send rate reaches 30 tps throughput of 3 and 5 clients start to degrade mildly. On average based on results, the proposed joining protocol can satisfy 1 client scenario almost to the fullest over increasing send rate, however regarding 3 and 5 clients we experienced mild decrease of throughput because of the increasing send rate and network congestion, such that the throughput of 1, 3, and 5 clients is 49.6 tps, 47.8, and 45.8 tps at send rate of 50 tps respectively. Overall joining protocol is proved to be an efficient protocol in SEHDaS.

Latency results of joining protocol is provided in Fig. 7, as it was expected latency in joining protocol is increasing linearly over increasing send rate. However,

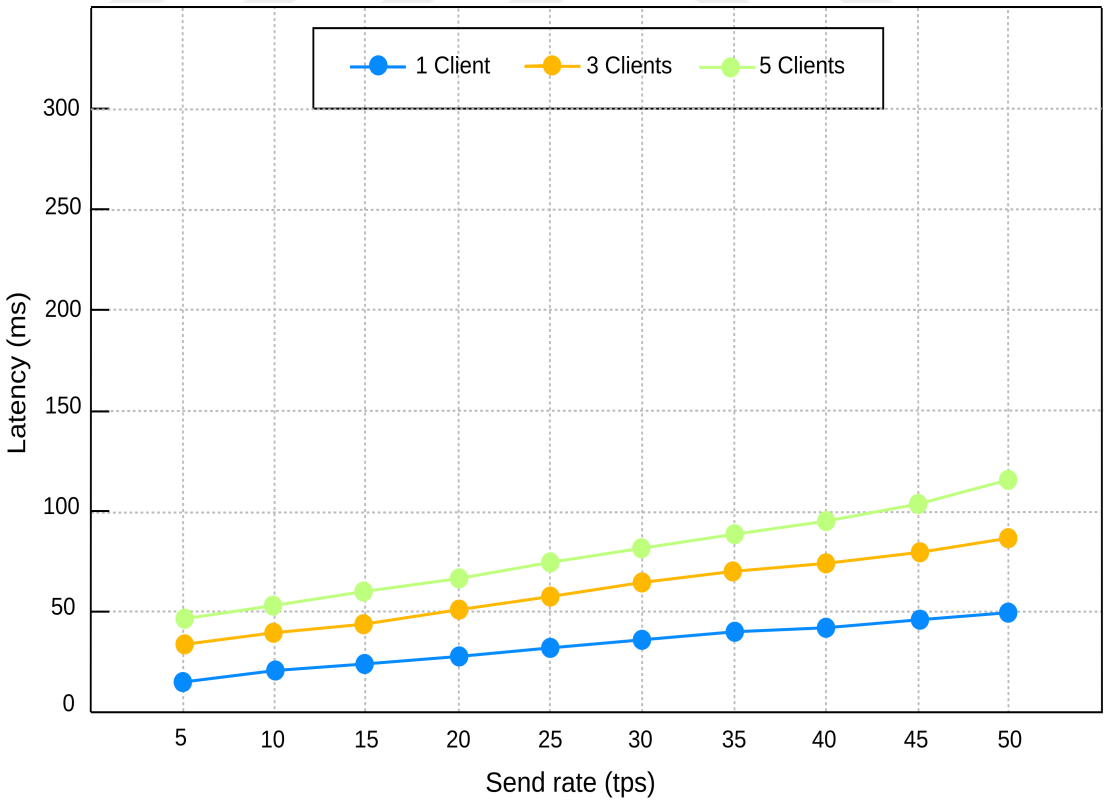

Figure 7: User joining latency for increasing number of clients, over increasing send rate

we experienced more stable results on one client scenario. And overall we are not experiencing huge latency increase differences by increasing the client numbers. In general, the latency in all cases in joining protocol is on acceptable level, such that latency of 1, 3 and 5 clients is respectively 19.3 ms, 38.9 ms, and 48.2 ms at 5 tps, and by increasing the send rate to 50 tps we experienced the latency increase to 49.8 ms, 87.6 ms, and 120.3 ms respectively.

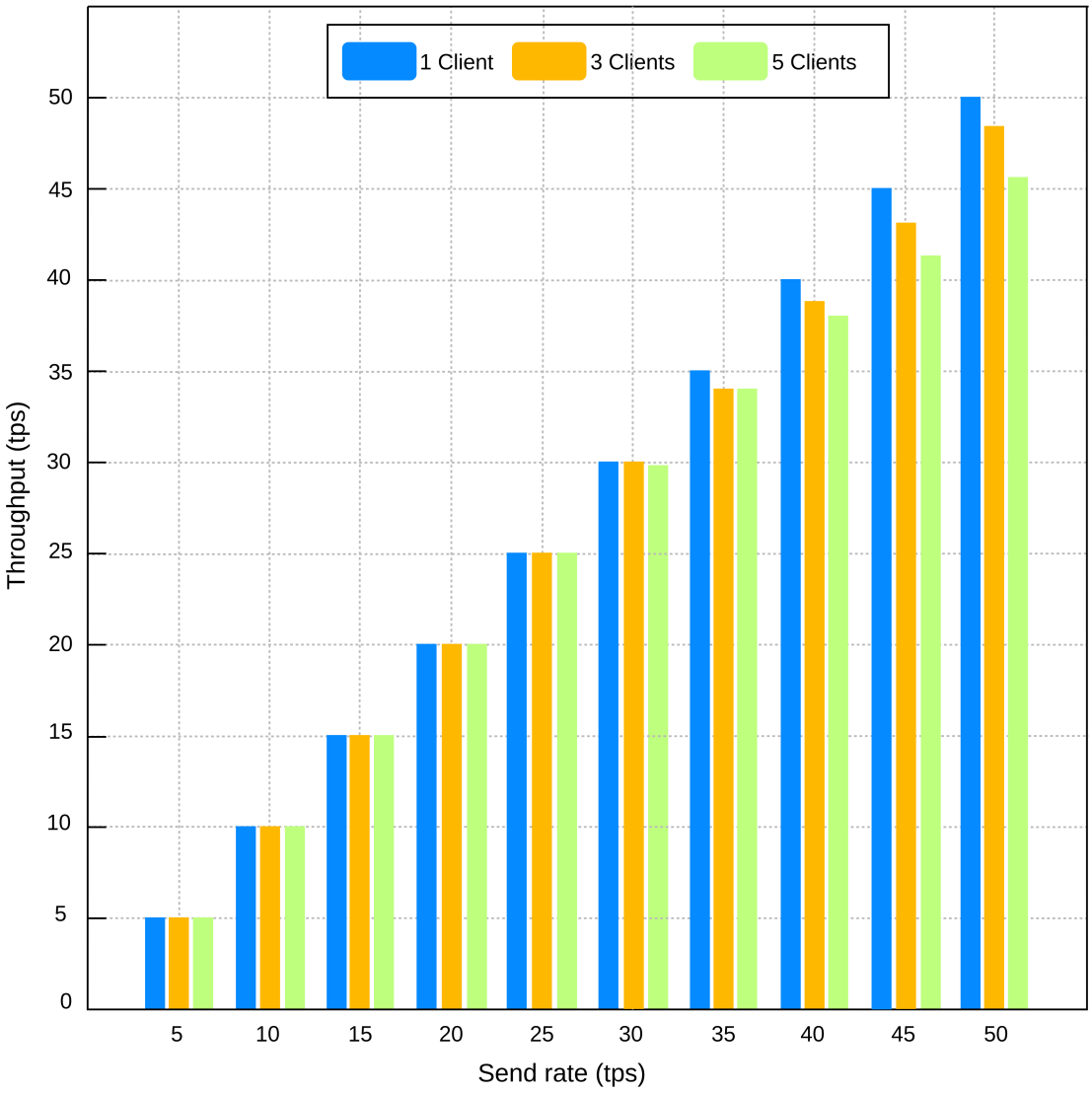

Figure 8: Key agreement throughput for increasing number of clients, over increasing send rate

4.2.2 Key agreement performance results

The throughput performance of proposed key agreement protocol is depicted in Fig. 8, based on experiment results in this protocol for one client case we experienced a complete throughput over increasing send rate. However, with increasing client numbers to 3 and 5 over increasing send rate, throughput increased up to 25 tps send rate, after reaching to 30 tps of send rate the throughput starts to degrade slowly over the rate goes up, until at 50 tps the throughput for 1, 3 and 5 clients reach to 50, 48.1 and 45.7 respectively. We experienced this decrease because of the increasing workload over increasing send rate.

Key agreement protocol latency is plotted in Fig. 9, in this protocol we experienced a similar latency to the user joining protocol. However, in key exchange protocol the latency is slightly higher in compared with user joining protocol, and as depicted there is a linear increase between the client numbers increase over

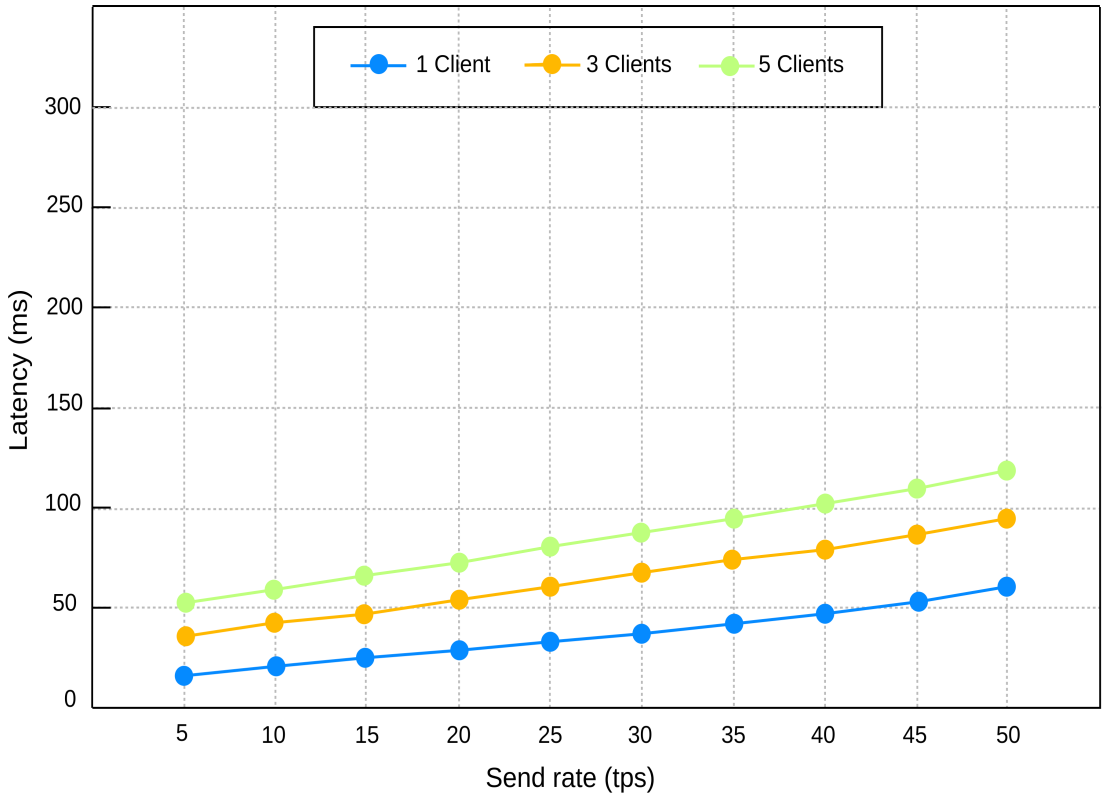

Figure 9: Key agreement latency for increasing number of clients, over increasing send rate

increasing send rate and latency, such that at 50 tps send rate, the latency of 1, 3, and 5 clients are 58 ms, 94 ms and 123 ms respectively.

4.2.3 Partial data access performance results

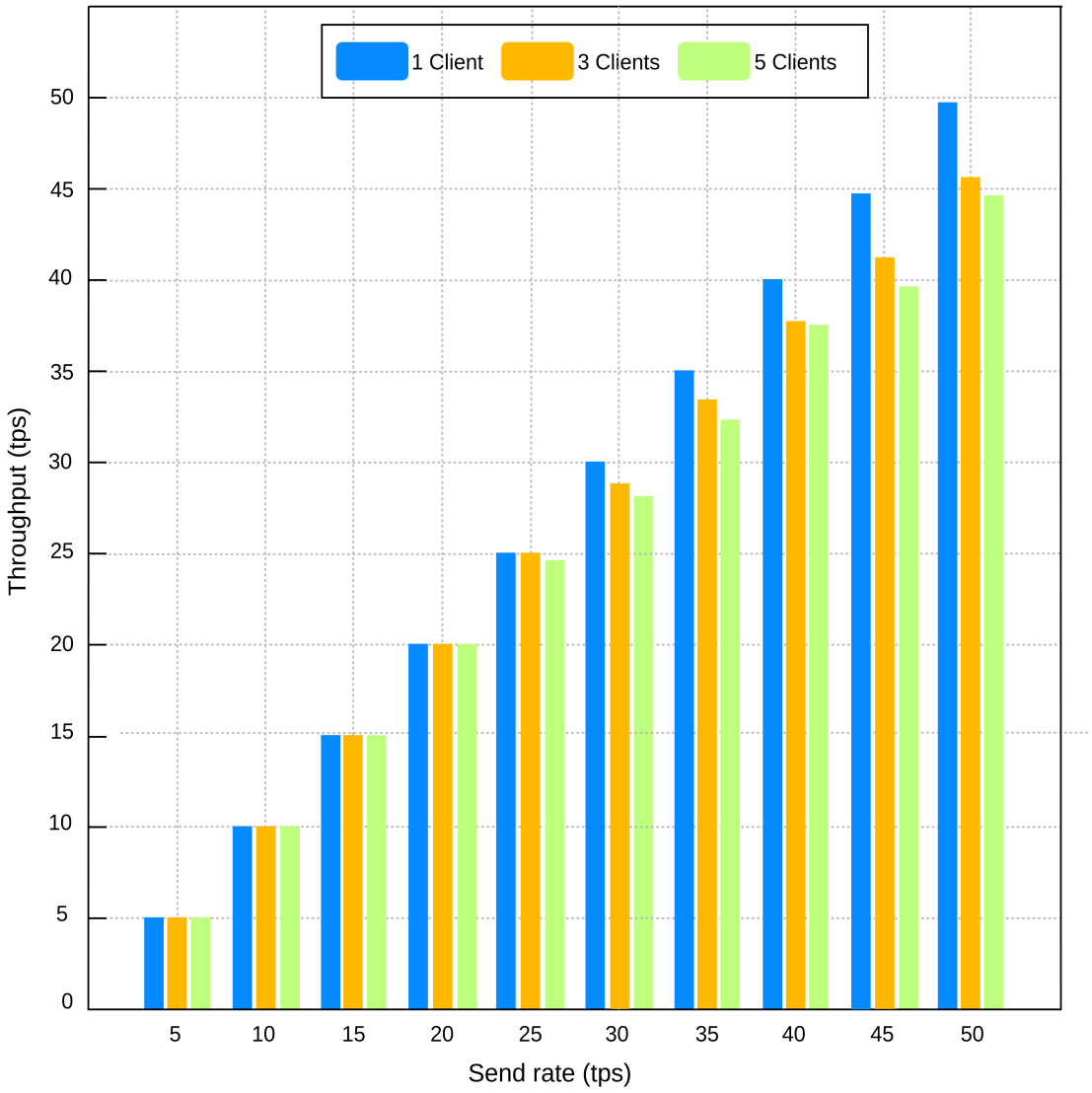

Figure 10: Partial access throughput for increasing number of clients, over increasing send rate

Once an emergency case happens the Emergency doctor requests access to the PHR data and this request is processed by Trusted-party and then access to the data is issued. We tested this protocol's throughput for various number of clients over different transaction send rate as well, Fig. 10 shows the average transaction throughput for partial data access. Unlike other protocols we experienced higher

throughput decreasing in this protocol, and this is because of the slightly higher computations in this protocol in compared with user joining protocol. The results showed a mild drop in throughput of 5 clients starting from 25 tps which is 24.8, and this decreasing continues over the send rate increase until we experienced 44.5 tps for throughput when the send rate reached 50 tps. For the 3 clients the results were similar to 5 clients, but the throughput decrease started from 30 tps send rate. However, the throughput of this protocol is almost fully satisfied in 1 client case.

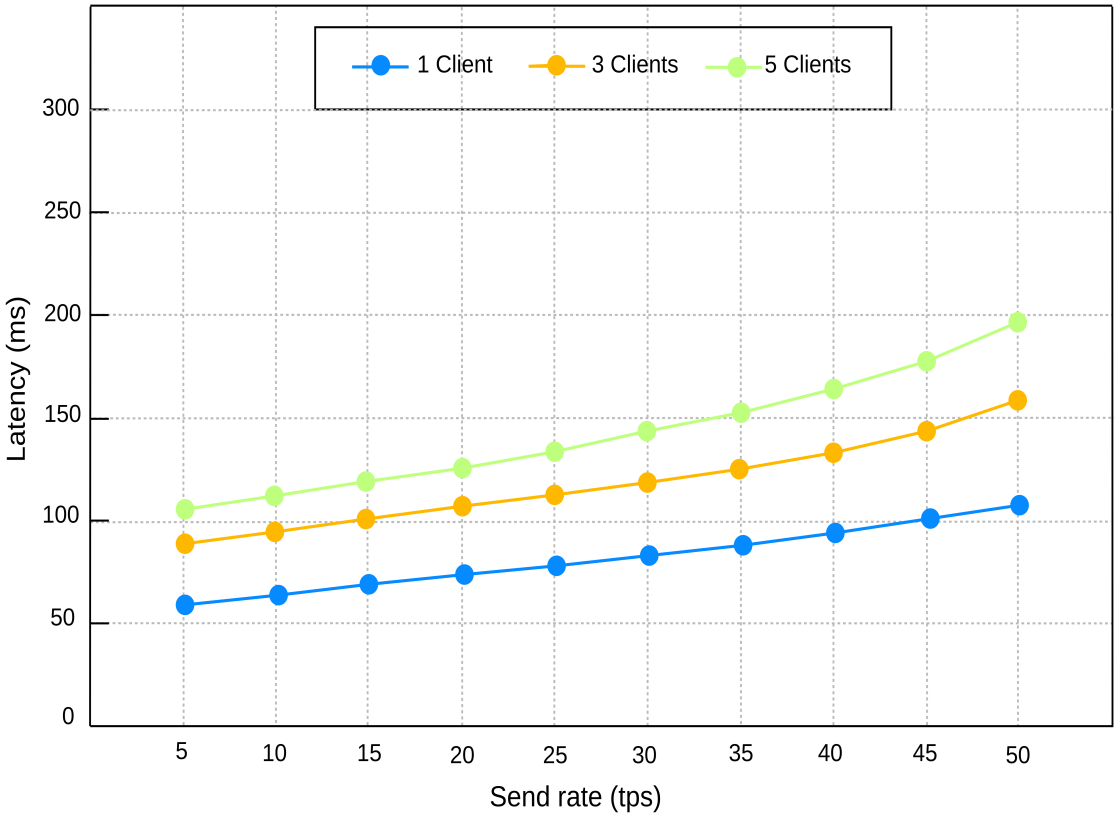

Figure 11: Partial access latency for increasing number of clients, over increasing send rate

Figure 11 shows the transaction latency for partial data access protocol, we observed that latency scales over increasing send rate in different number of clients. specifically in 3 and 5 number of clients we experienced a sharp increase in compared with one client. Although the overall latency in this protocol is almost doubled in compared with previous protocols, but still network shows an efficient performance for increasing the number of clients, however increasing the send rate too much may have a significant impact on the latency increase because of the increased traffic

load and computation in data access protocol. Possible reasons for experimenting low latency in our system in compared with the existing research works is parallel transaction execution and the off-chain database mechanism we employed for SeHDaS, because it offloads the heavy lifting of huge data-sets from blockchain nodes.

Based on experimental results provided for each protocol, the overall system throughput is proved to be high even by increasing the number of clients over increasing send rate. One reason for the high throughput in our system is because of the parallel scheduling opportunity of Hyperledger Sawtooth network that we utilized for implementing our architecture, which leads to a higher throughput and lower latency by parallel execution of transactions regarding the data load amount in the network. so this produces better performance results in compared with serial executions of other blockchain networks. Another key feature in SEHDaS that resulted on an efficient system and low latency is the off-chain database mechanism we employed, since this approach offloads the heavy lifting of huge data-sets from blockchain nodes.

4.2.4 Comparison of average response time and memory usage for complete transactions

Finally, the average response time and memory usage are calculated precisely for each transaction in the process of emergency PHR accessing. These transaction functionalities involves user joining including patient, trusted-party and emergency doctor (ED) joining to the blockchain, and finally PHR data accessing by Emergency doctor. For computing the average values of provided data, we repeated this experiment 100 times. In EACMs, a proposed system by [43], the average response time from the beginning of the emergency data request process starting from user joining until accessing PHR data takes 15 up to 18 seconds. And The average memory usage in this process takes 750 up to 900 bytes. Whereas, in practice for SEHDaS it takes 1.86 seconds for user joining, and for sharing the data it takes up to 2.42 seconds. Therefore, in our proposed system the whole

Table 4: Average latency and memory usage comparison of SeHDaS and EACMs

Transactions	**SEHDaS performance (avg.)**		**EACMs performance (avg.)**	
	Latency (msec)	***Memory Usage (byte)***	***Latency (msec)***	***Memory Usage (byte)***
Patient-joining	1863	154	6002	236
Trusted-party-joining	1863	154	6231	236
Emergency-doctor-joining	1863	154	5963	236
Patient-data-accessing	2423	231	5683	568

process time from all users joining to getting data for replying to emergency data request is completed in 8 up to 10 seconds. And the average memory usage for completing the emergency request process for SEHDaS is around 650 up to 700 bytes. The average response time and memory usage in each step of joining and PHR accessing process of SEHDaS and EACMs for a complete transaction is depicted in Table 4 for comparison. Based on experimental results SEHDaS has proved to significantly improve the response time and memory usage for PHR accessing system in emergency conditions.

4.3 Security and Privacy Discussion

Our proposed system guarantees user privacy by controlling the access to PHR data with pre-specified rules using transaction families and transaction processors. Furthermore permissioned blockchain framework lets only the authorized users to enter the blockchain and trigger transaction as a network participant. Even the participated users can access the PHR partially or a a whole based on policies defined by the PHR data owner. Besides, all transactions' log are updated and stored in the blockchain ledger by transaction processors, and later the PHR owner can audit which user have accessed to which part of his/her data. Our framework protects network security and data breaches such as unauthorized access. In our proposed system all batches and transactions are signed by the client and submitted to the validator node by using his private key, with this approach

data confidentiality and integrity are preserved. Also, before any transaction the metadata and parameters in the payload of blockchain are validated for avoiding buffer overflows. Validator in transaction families ignore any unauthorized transaction processors, hence improves the network's security. More importantly we utilized an off-chain storage approach for storing the encrypted PHR data which makes the network to run faster with less latencies by avoiding unnecessary network confirmations for personal data information. It also makes storage infrastructure more scalable. So, we can have off-chain data to be in any size of data which is too large to be stored efficiently on blockchain. As mentioned before, storing data off-chain brings huge value to both privacy and security aspects of our system. It increases the privacy of the data by not saving personal data on-chain in an available mode for all participants, and by off-chain storing we are having a tamper-proof storage that prevents unauthorized access and modifications. Whereas, using on-chain storage for data rises the probability of accidental disclosure of confidential health data. Furthermore, utilizing a time constraint in our system for data access rises the security in the system, since after the time expiration the data will not be accessible by the Emergency doctor anymore.

CHAPTER V

CONCLUSION

Emergency access to PHR data has been a challenge for many years and several system solutions have been proposed in the recent years. Blockchain technology is a proved solution to enhance the healthcare data management and address the potential challenges of medical data systems such as single point of failure, patient privacy concerns, sensitive healthcare data security, authentication and access management, and overall system efficiency that were not addressed in previous cloud-based and central-server-based approaches. Nevertheless, most of the research investigations are focused on public blockchain based systems, and as we pointed out in the study this type of systems are not efficient and cannot fully satisfy security and privacy concerns, besides they suffer from high execution time and non-scalability issues. In the most recent years some researchers have slightly dedicated on private blockchain solutions for medical data, yet still the proposed systems are not fulfilling the patient health records access for emergency conditions. Hence, there is a crucial need for a more efficient, secure and private system to satisfy all challenges of healthcare data management especially when there is an emergency case. For example latency factor is a vital parameter in patient treatment process especially when facing with an emergency case, as any delay in this process may lead to failure in treatment and end up to a life-threatening situation.

In this paper, we proposed a new Blockchian-Based Secure Emergency Healthcare Data Sharing system called SEHDaS, which guarantees patient's privacy and data security in emergency conditions data access. To this end we designed a secure and private emergency PHR data access protocol and implemented it on hyperledger sawtooth network, a permissioned private blockchain technology.

Utilizing hyperledger sawtooth special properties such as transaction families and validators, we are able to manage the patient defined access rules and policies and validate all transactions and user authentication procedure. Hence the PHR data is only accessible by the valid members of the blockchain network which are validated by the admin during joining process. Our architecture uses off-chain database approach for storing healthcare records in order to enhance the scalability, reduce the network congestion and so improve the response time by avoiding the waiting time for network confirmation. So utilizing this approach along with parallel transaction execution in hyperledger sawtooth network PHR access is conducted with minimum delay and instantaneously while preserving the patient's privacy.

We discussed the effectiveness of our proposed protocol for addressing the private and secure emergency access to PHR data. We also implemented the proposed architecture through the hyperledger sawtooth blockchain in order to assess the SEHDaS efficiency and compare the performance metrics with EACMs. We carried out performance benchmark testing on our implemented blockchain network and analyzed the latency and throughput in different scenarios of number of client's increment over increasing send rate for each case. The experimental results demonstrated that our framework guarantees a secure and private data access in emergency conditions where patient is not conscious and cannot take part in authentication process, by minimizing the disk usage and latency significantly, along with increasing the network throughput, which leads to an efficient system, and can be contemplated as a potential adoption for future healthcare systems which are developed based on blockchain technology.

Bibliography

[1] J. M. Roman-Belmonte, H. De la Corte-Rodriguez, and E. C. Rodriguez-Merchan, "How blockchain technology can change medicine," *Postgraduate medicine*, vol. 130, no. 4, pp. 420–427, 2018.

[2] A. Al Omar, M. S. Rahman, A. Basu, and S. Kiyomoto, "Medibchain: A blockchain based privacy preserving platform for healthcare data," in *International conference on security, privacy and anonymity in computation, communication and storage*, pp. 534–543, Springer, 2017.

[3] K. Yaeger, M. Martini, J. Rasouli, and A. Costa, "Emerging blockchain technology solutions for modern healthcare infrastructure," *Journal of Scientific Innovation in Medicine*, vol. 2, no. 1, 2019.

[4] T. Heston, "Why blockchain technology is important for healthcare professionals," *Available at SSRN 3006389*, 2017.

[5] S. Ji, Z. Cai, M. Han, and R. Beyah, "Whitespace measurement and virtual backbone construction for cognitive radio networks: From the social perspective," in *2015 12th annual IEEE international conference on sensing, communication, and networking (SECON)*, pp. 435–443, IEEE, 2015.

[6] M. Han, M. Yan, J. Li, S. Ji, and Y. Li, "Generating uncertain networks based on historical network snapshots," in *International Computing and Combinatorics Conference*, pp. 747–758, Springer, 2013.

[7] S. Underwood, "Blockchain beyond bitcoin," *Communications of the ACM*, vol. 59, no. 11, pp. 15–17, 2016.

[8] G. Yang, C. Li, and K. E. Marstein, "A blockchain-based architecture for securing electronic health record systems," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 14, p. e5479, 2021.

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

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

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

[12] A. Zhang and X. Lin, "Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain," *Journal of medical systems*, vol. 42, no. 8, pp. 1–18, 2018.

[13] M. J. Rennock, A. Cohn, and J. R. Butcher, "Blockchain technology and regulatory investigations," *Practical Law Litigation*, pp. 35–44, 2018.

[14] U. Khalid, M. Asim, T. Baker, P. C. Hung, M. A. Tariq, and L. Rafferty, "A decentralized lightweight blockchain-based authentication mechanism for iot systems," *Cluster Computing*, pp. 1–21, 2020.

[15] A. Sunyaev, "Distributed ledger technology," in *Internet Computing*, pp. 265–299, Springer, 2020.

[16] C. Cachin and M. Vukolić, "Blockchain consensus protocols in the wild," *arXiv preprint arXiv:1707.01873*, 2017.

[17] S. Zhai, Y. Yang, J. Li, C. Qiu, and J. Zhao, "Research on the application of cryptography on the blockchain," in *Journal of Physics: Conference Series*, vol. 1168, p. 032077, IOP Publishing, 2019.

[18] L. Ismail, H. Materwala, and S. Zeadally, "Lightweight blockchain for healthcare," *IEEE Access*, vol. 7, pp. 149935–149951, 2019.

[19] E. Dadvar and K. Kalkan, "A survey: blockchain utilization for securing healthcare system," *International Advanced Researches and Engineering Journal*, vol. 5, no. 2, pp. 324–333, 2021.

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

[21] M. Puppala, T. He, X. Yu, S. Chen, R. Ogunti, and S. T. Wong, "Data security and privacy management in healthcare applications and clinical data warehouse environment," in *2016 IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, pp. 5–8, IEEE, 2016.

[22] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," in *2016 2nd international conference on open and big data (OBD)*, pp. 25–30, IEEE, 2016.

[23] A. Theodouli, S. Arakliotis, K. Moschou, K. Votis, and D. Tzovaras, "On the design of a blockchain-based system to facilitate healthcare data sharing," in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pp. 1374–1379, IEEE, 2018.

[24] M. Wohrer and U. Zdun, "Smart contracts: security patterns in the ethereum ecosystem and solidity," in *2018 International Workshop on Blockchain Oriented Software Engineering (IWBOSE)*, pp. 2–8, IEEE, 2018.

[25] H. Liu, R. G. Crespo, and O. S. Martínez, "Enhancing privacy and data security across healthcare applications using blockchain and distributed ledger concepts," in *Healthcare*, vol. 8, p. 243, Multidisciplinary Digital Publishing Institute, 2020.

[26] P. Zhang, J. White, D. C. Schmidt, and G. Lenz, "Applying software patterns to address interoperability in blockchain-based healthcare apps," *arXiv preprint arXiv:1706.03700*, 2017.

[27] A. Ekblaw, A. Azaria, J. D. Halamka, and A. Lippman, "A case study for blockchain in healthcare:"medrec" prototype for electronic health records and medical research data," in *Proceedings of IEEE open & big data conference*, vol. 13, p. 13, 2016.

[28] R. Kumar and R. Tripathi, "Scalable and secure access control policy for healthcare system using blockchain and enhanced bell–lapadula model," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 2, pp. 2321–2338, 2021.

[29] S. Chakraborty, S. Aich, and H.-C. Kim, "A secure healthcare system design framework using blockchain technology," in *2019 21st International Conference on Advanced Communication Technology (ICACT)*, pp. 260–264, IEEE, 2019.

[30] M. T. de Oliveira, L. H. Reis, R. C. Carrano, F. L. Seixas, D. C. Saade, C. V. Albuquerque, N. C. Fernandes, S. D. Olabarriaga, D. S. Medeiros, and D. M. Mattos, "Towards a blockchain-based secure electronic medical record for healthcare applications," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1–6, IEEE, 2019.

[31] S. Wang, D. Zhang, and Y. Zhang, "Blockchain-based personal health records sharing scheme with data integrity verifiable," *IEEE Access*, vol. 7, pp. 102887–102901, 2019.

[32] B. Shen, J. Guo, and Y. Yang, "Medchain: Efficient healthcare data sharing via blockchain," *Applied sciences*, vol. 9, no. 6, p. 1207, 2019.

[33] 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.

[34] H. Li, L. Zhu, M. Shen, F. Gao, X. Tao, and S. Liu, "Blockchain-based data preservation system for medical data," *Journal of medical systems*, vol. 42, no. 8, pp. 1–13, 2018.

[35] X. Yue, H. Wang, D. Jin, M. Li, and W. Jiang, "Healthcare data gateways: found healthcare intelligence on blockchain with novel privacy risk control," *Journal of medical systems*, vol. 40, no. 10, pp. 1–8, 2016.

[36] A. D. Dwivedi, G. Srivastava, S. Dhar, and R. Singh, "A decentralized privacy-preserving healthcare blockchain for iot," *Sensors*, vol. 19, no. 2, p. 326, 2019.

[37] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "Blockchain leveraged decentralized iot ehealth framework," *Internet of Things*, vol. 9, p. 100159, 2020.

[38] H. Tian, J. He, and Y. Ding, "Medical data management on blockchain with privacy," *Journal of medical systems*, vol. 43, no. 2, pp. 1–6, 2019.

[39] R. Arul, Y. D. Al-Otaibi, W. S. Alnumay, U. Tariq, U. Shoaib, and M. J. Piran, "Multi-modal secure healthcare data dissemination framework using blockchain in iomt," *Personal and Ubiquitous Computing*, pp. 1–13, 2021.

[40] A. Banerjee, P. Agrawal, and R. Rajkumar, "Design of a cloud based emergency healthcare service model," *International Journal of Applied Engineering Research*, vol. 8, no. 19, pp. 2261–2264, 2013.

[41] P. Thummavet and S. Vasupongayya, "A novel personal health record system for handling emergency situations," in *2013 International Computer Science and Engineering Conference (ICSEC)*, pp. 266–271, IEEE, 2013.

[42] K. Rabieh, K. Akkaya, U. Karabiyik, and J. Qamruddin, "A secure and cloud-based medical records access scheme for on-road emergencies," in *2018 15th*

IEEE Annual Consumer Communications & Networking Conference (CCNC), pp. 1–8, IEEE, 2018.

[43] A. R. Rajput, Q. Li, M. T. Ahvanooey, and I. Masood, "Eacms: Emergency access control management system for personal health record based on blockchain," *IEEE Access*, vol. 7, pp. 84304–84317, 2019.

[44] A. F. Hussein, N. ArunKumar, G. Ramirez-Gonzalez, E. Abdulhay, J. M. R. Tavares, and V. H. C. de Albuquerque, "A medical records managing and securing blockchain based system supported by a genetic algorithm and discrete wavelet transform," *Cognitive Systems Research*, vol. 52, pp. 1–11, 2018.

[45] G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," *Sustainable cities and society*, vol. 39, pp. 283–297, 2018.

[46] A. natarajan, "Medical records 10 yrs." `https://data.world/arvin6/medical-records-10-yrs/activity`, 2017. The dataset accessed August 7, 2021.

[47] J. Eckard, "Storage for blockchain and modern distributed database processing." `https://www.ibm.com/blogs/systems/storage-for-blockchain-and-modern-distributed-database-processing/`, 2019. The article accessed June, 2021.

[48] H. Zhao, P. Bai, Y. Peng, and R. Xu, "Efficient key management scheme for health blockchain," *CAAI Transactions on Intelligence Technology*, vol. 3, no. 2, pp. 114–118, 2018.

[49] J. Michael, A. Cohn, and J. R. Butcher, "Blockchain technology," *The Journal*, vol. 1, no. 7, 2018.

[50] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges.," *Int. J. Netw. Secur.*, vol. 19, no. 5, pp. 653–659, 2017.

VITA

Elnaz Dadvar received her B.Sc. degree in Information Technology Engineering from Urmia Payam Noor University, Urmia, Iran in 2015. She is currently working toward the M.Sc. degree in Computer Science department as a research assistant under the supervision of Professor Kubra Kalkan, at Ozyegin University, Istanbul, Turkey. Her current research interests include Network Security, Information Security, and Blockchain Technology Applications in Healthcare Domain.