The Journal of Supercomputing
https://doi.org/10.1007/s11227-020-03570-x

Towards design and implementation of security and privacy framework for Internet of Medical Things (IoMT) by leveraging blockchain and IPFS technology

Randhir Kumar[1] · **Rakesh Tripathi**[1]

Accepted: 14 December 2020

Abstract
The Internet of Medical Things (IoMT) is the next frontier in the digital revolution and it leverages IoT in the healthcare domain. The underlying technology has changed the current healthcare system by collecting real-time data of patients and providing a patient motioning system. But IoMT also presents a big challenge for data storage management, security, and privacy due to cloud-based storage. Today, this large volume of IoMT generated medical data is stored in the centralized storage system. However, centralization of patient sensitive information leads to a single point of failure, privacy, and security concern. To address these issues, we propose a smart contracts enabled consortium blockchain network. We integrated interplanetary file systems (IPFS) cluster node where smart contracts are deployed at the initial stage for authentication of patient's and medical devices, the same cluster layer is also proposed as a distributed data storage layer for device-generated data after authentication and these data are securely transmitted over the consortium blockchain. The IPFS cluster node ensures the security and authentication of the devices and it also provides secure storage management in IoMT enabled healthcare system. The consortium network enables the privacy of data owing to hash-based storage in a block of IoMT enabled healthcare network.

Keywords Blockchain · IoMT · IPFS · Smart contract

✉ Randhir Kumar
rkumar.phd2018.it@nitrr.ac.in

Rakesh Tripathi
rtripathi.it@nitrr.ac.in

[1] Department of Information Technology, National Institute of Technology, Raipur, C.G 492010, India

Published online: 12 January 2021

1 Introduction

The number of connected devices is increasing rapidly day by day with its design and development. The modern network and infrastructure are taking into account with constant growth in connected devices. Today most of the manufacturing houses have adopted a new view of the internet through the Internet of Things (IoT). The IoT is expanding ever further and becoming an important part of the future internet. According to the Statistica report, by 2025 the total number of connected devices worldwide would be around 75.44 billion [1]. Substantial progress in the machine-to-machine interface is expected and which may be connected to a broader range of applications. The Internet of things plays a major role in the field of healthcare providing ease to patients and doctors [2]. It consists of various devices that communicate with each other and track the patient's vital data including their medical information. The IoT is growing as a potential component for revolutionizing the current healthcare system and providing immense benefits to the healthcare domain, ranging from drug discovery, disease predictive analysis, early warning epidemics, preventive healthcare, and patient health monitoring.

The Internet of Medical Things (IoMT) is one of the most fast-growing markets in the healthcare system [3]. The technology is leveraged from the IoT which is used in the processing and analyzing the driving innovation in the current healthcare system, with the advancement of connected medical devices. These devices can capture, create, analyze, and transfer the data. These connecting devices build the infrastructure of IoMT enabled healthcare system. The linked IoMT devices provide real-time monitoring and rescue the lives of many patients' from several critical diseases [4]. Besides, IoMT devices fulfill the shortage of doctors in the current healthcare system by automating the complete healthcare system from data collections to predictive analysis and report generation [5]. The IoMT combines the digital and physical world to speed up the diagnosis and care cycle with greater precision to enhance patient safety and to change patient behavior and health status in real-time which is shown in Fig. 1. Connection of medically related devices will have a profound impact on patients and clinicians. Moreover, the connected IoMT medical devices for prognosis, monitoring, and treatment of patients are anticipated to rise from $14.9 billion to $52.2 billion by 2022 [5]. It is expected that the total amount of data produced by IoMT devices will reach up to 847 ZB (Zettabytes) by 2021 according to recent study reports of Cisco [6]. This large volume of data is difficult for traditional approaches to analyze, store, and manage. Thus, there is a compelling need for a distributed data storage layer to propel IoMT enabled healthcare system.

Along with rapid growth and diversified nature, security, privacy, and data storage management in IoMT have become a major challenge due to third-party authentication or cloud-based storage. Hence, security, privacy, and data storage of IoMT enabled healthcare system are our primary consideration [7–9]. Therefore, distributed security and privacy majors are needed to secure IoMT devices. The existing security and privacy techniques are not appropriate for IoMT

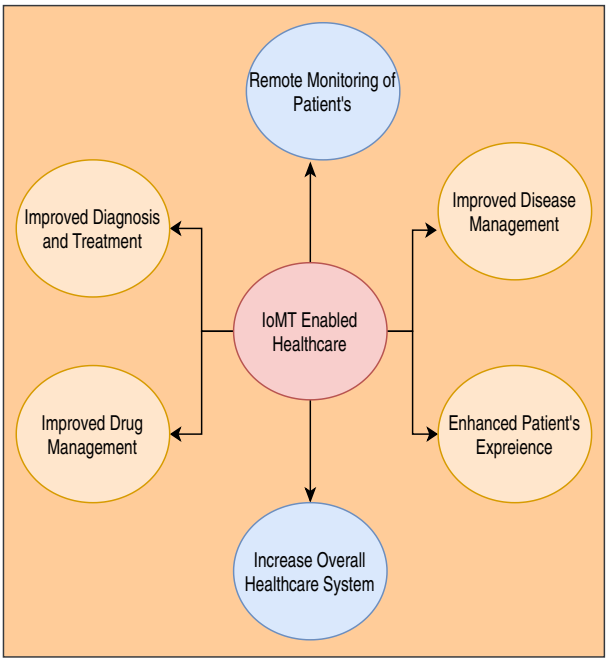

Fig. 1 Benefits of IoMT in healthcare system

devices because of their centralized working principle of authentication and storage. Thus, the requirements show a clear need for a distributed way of data sharing and storing where patients are more sure about their data privacy and device security. Besides, it should provide a holistic transaction and their interaction for healthcare providers in IoMT healthcare network [10].

The InterPlanetary File System (IPFS) cluster is a peer-to-peer distributed storage model which is designed to create persistent storage for shared transactions [11]. It facilitates the distributed hash table (DHT) and versioning control system where each transaction is denoted by unique fingerprints (hash value). IPFS cluster node mitigates the redundancy of the transaction in a network by matching the fingerprints in DHT. It secures and authenticates the data by applying encryption techniques.

The blockchain is an emerging technology that can provide a suitable solution for authentication and access services in IoMT enable healthcare network, owing to cryptographic properties and its decentralized nature. A blockchain consists of a chain of a block which is timestamped and linked by cryptographic hashes that are distributed among the peers of network [12, 13]. Initially, the blockchain protocol was designed and implemented for the exchange of money between two different parties. However, the security professionals around the globe are utilizing the blockchain to strengthen security and privacy issues in IoMT. The properties of blockchain include reliability, tempered-proof, and fault-resistance makes blockchain an appealing approach for authentication, authorization, and storage in various domain

[14]. Blockchain also enables the integration of smart contracts which offer access control mechanisms for IoMT devices in the healthcare domain. Therefore, blockchain technology and IPFS cluster provide good grounds for building and managing distributed and decentralized infrastructure, trust, integrity, authenticity, privacy, security, and storage solutions in IoMT systems [15].

1.1 Motivation

From Table 1, we notice that verification and authorization are two major problems that need to be considered in the IoMT network [12, 16–18]. The existing approach of authentication and authorization systems introduced for IoMT is centralized and rely on the third parties [19, 20]. The cloud-based storage structures are possibly vulnerable to a different security threat, scalability, privacy concern, and single point of failure that limits the system with less reliability and trust in data sharing [8, 21–23].

The security and privacy standards of the IoMT healthcare networks are even more strict than that of the traditional healthcare infrastructures. IoMT healthcare systems have several additional security requirements, such as patient authentication, device authentication, to ensure system security and privacy. Each level of IoMT healthcare systems has different functionalities along with security and privacy requirements [18]. These challenges must be incorporated with the IoMT enabled healthcare systems. To address these issues of security, privacy, and storage management in IoMT enabled healthcare system, we propose an enhanced security and privacy framework for IoMT by leveraging blockchain and IPFS technology.

Contribution These are the following contributions of the article which are listed below.

1. A new decentralized framework based on IPFS cluster node and smart contract is used for authentication and access control in IoMT enabled healthcare system. This authentication process ensures security in the system.
2. A proof of identity-based authorization model is designed and implemented to meet the IoMT security requirement.
3. A data storage layer is presented with IPFS cluster nodes which are distributed in nature and mitigate the single point of failure in IoMT enable healthcare system. The integration with smart contracts verifies the security concern in the IoMT healthcare system.
4. A consortium blockchain is designed and implemented to preserve the privacy of patient medical data using the ethereum Ropsten network.

The rest of the paper is organized as follows: Sect. 2 discusses the background and related Study. The proposed framework for IoMT healthcare by leveraging blockchain and IPFS technology is presented in Sect. 3. The security and privacy majors in IoMT enabled healthcare are described in Sect. 4. The secure transmission of data in the IoMT blockchain network is described in Sect. 5. We present the implementation of the IoMT application interface in Sect. 6. The result analysis of

Table 1 Summary of storage, privacy, and security issue in IoMT

References	Year	Paper objective	Services	Advantages	Limitations
[8]	2017	Proposed risk assessment approach in a IoMT environment	Cloud computing	Support decision making	Loss of data
[21]	2018	Presented a privacy scheme for an IoT smart healthcare network	Cloud computing	Better privacy preservation of health data	Not scalable and lead to single point of failure
[16]	2018	Presented a health IoT architecture	-NA-	Better quality of care	Lack of privacy
[22]	2018	Establish a trust of different layers on healthcare IoT	Cloud	Support machine-to-machine communication	Scalability and privacy issue
[10]	2018	Discussed the IoMT privacy and security challenges	Cloud storage	Designed decision system in healthcare	Privacy issue
[17]	2019	Offer a perception of new research endeavors for the development and advancement IoMT ecosystem	Cloud-based centralized storage	Server/cloud	Privacy and security issue
[43]	2019	Monitoring the health of patients and enabling the practice of precision medicine that can yield a more informed diagnosis and optimized treatment for patients	-NA-	Monitoring system for remote patient	Data privacy and integrity issue
[2]	2019	Stepping on a human-centric perspective of designing IoMT systems	-NA-	Remote patient's healthcare monitoring	Not suitable in real-time monitoring due to data privacy
[18]	2020	The comparative study of healthcare 's latest protection and privacy survey 4.0 is carried out	-NA-	Provide perspectives on healthcare 4.0 security and privacy issues	security and privacy issue
[23]	2020	Discussed various challenge in IoMT enabled healthcare	Cloud	IoMT enabled healthcare in today life with various privacy and security concerns	Privacy of patients and the confidentiality, integrity and availability of medical services is the challenge
[12]	2020	Discussed advantage of blockchain in healthcare and IoMT	Blockchain storage for accountability of information	How blockchain can be part of IoMT	Not given proven implementation of security and privacy issue

Table 1 (continued)

References	Year	Paper objective	Services	Advantages	Limitations
[44]	2019	The authors have explored the use of IoMT in healthcare monitoring system	Cloud services	Monitoring of wearable devices in healthcare	The cloud-based storage makes limits with various security and privacy issues
[45]	2018	The authors have shown the individual patient health monitoring using biomedical sensor signal	Centralized database	Current healthcare system can be improved by capturing the biomedical signal of patient	The centralized storage model is used to capture the signal, which can be not good for the healthcare system
[46]	2019	The authors have proposed cluster-based IoMT edge services for healthcare monitoring	Cluster-based services	Cluster services are applied to increase the efficiency in healthcare system	The cluster-based structure can have limitation like privacy leakage of information due do centralized cluster head

IoMT operations is presented in Sects. 7 and 8 presented comparison of different IoMT enabled healthcare network, and Sect. 9 concludes the paper.

2 Background and related work

This section discusses the challenges of security, privacy, and storage in IoMT enabled healthcare system and is divided into two different parts. The first part discusses the security and privacy issue in IoMT. The second part explains the requirements of storage in IoMT.

2.1 Security and privacy in IoMT

Related health measuring and comparison work have a relatively long history. However, there is currently very little work on recommending and evaluating the security of IoMT applications, due to the newness of IoMT. The following paragraph discusses the previous work in this area and highlights their shortcomings and challenges. Some of these efforts are completely focused on patient monitoring system and do not fulfill the requirement of other use cases in healthcare [24, 25]. Other researchers have been developing tools to evaluate and compare IoMT solutions concerning their security features, but these tools are not available to the public and maybe a little complex for novice users [16, 26–30]. Moreover, some of the suggested tools that recommend evaluating IoMT are based on hypothetical and generalized recommendations for security. They are focused mostly on the physical security of the IoMT without mentioning the privacy of the data [31–35]. Some other techniques are proposed which do not have the security requirement for users in the healthcare system [36]. Besides, some researchers have built their tools based on IoT architectures to construct evaluation criteria for security. However, they do not sufficiently guarantee the IoMT's security because system architecture is interconnected with no clear boundaries [37].

2.2 Storage in IoMT

Nowadays, due to technological advancement, the traditional system generates huge amounts of data that cannot store and process such massive amounts of data. Thus, data storage management and its processing is, therefore, a challenging issue, which is currently solved by the use of a cloud-based centralized system. Zhou et al. [38] proposed privacy-preserving techniques for the healthcare system by using the cloud-enabled healthcare system. The underlying approach is secure for the input and output of data, but it doesn't build trust in the system.

The storage of IoMT generated data is also the biggest challenge in IoMT enable healthcare system. The current-state-of-art is working completely on centralized storage like cloud infrastructure and authenticated by third-party [18]. Some of the work is presented for the storage and authentication of the data. Ziglari et al. [39] proposed security aspects for healthcare system deployment models. The authors

have conducted a security analysis between service providers and cloud service providers. They also provided an architecture for implementing information management systems in the cloud with different cloud providers. However, they have not mentioned any cryptographic algorithms and methods of access control to preserve the privacy and security of the medical data. Similarly, Requena et al. [40] presented a cloud-assisted gateway architecture to allow the patients access to cloud health resources and diagnostic reports.

Deshmukh et al. [41] developed a system for the management of health records, where patients and healthcare providers (doctors) can access records using key control techniques. The authors have not discussed a safe environment for the privacy of data and security in the system. Authors in [42], proposed a proof-based authentication approach for data privacy and security assistance in terms of authentication and authorization. However, they targeted mostly on integrity of data, authentication, but very little discussions are done on the other security aspects like access and audit trails confidentiality.

2.3 Existing IoMT model design and it's challenges

The existing centralized model of the IoMT enabled healthcare system is shown in Fig. 2. The working model is divided into three different layers viz., sensor layer, authentication layer, and storage layer. The sensor layer generates the data of patient health activity and this data are authenticated by the third-party authentication layer before adding it to the storage layer (cloud) for further accessibility. The current state-of-the-art for IoMT-based healthcare systems has been using third-party resources for authentication of the data and their visibility. However, these resources lack trust. The current storage layer for the authenticated data is done by cloud storage. However, the accessibility of the data from the cloud is an on-demand service, and the healthcare providers have to depend on the third-party. The current structure is not feasible in terms of security, privacy, trust, and storage (centralized cloud layer) as shown in Table 1. Thus, it demands a distributed structure for trust and secure storage layer considering privacy and security issues.

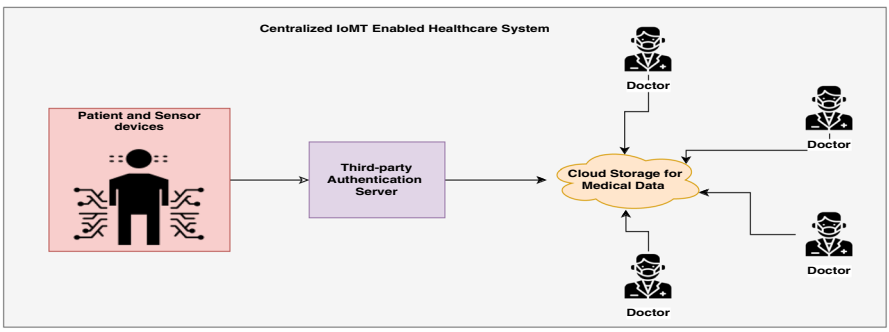

Fig. 2 Existing model for IoMT enabled healthcare using third-party authentication server and centralized storage

3 Proposed framework for IoMT healthcare

The main aim of this work is to provide a distributed framework for storage and authentication of medical devices that can prevent various security and privacy obstacles in IoMT enabled healthcare. The proposed model is designed into two different parts (1) authentication and authorizations of the patient registration and medical devices (2) dissemination of the information in the blockchain network to ensure the privacy of the patient data.

3.1 Architecture of framework

Figure 3 illustrates the proposed mechanism and discusses two major parts, namely medical device parts and IPFS cluster parts which are later discussed in this article.

Medical device This part is responsible for deploying different medical devices in the IoMT to enable healthcare (individual patient's medical devices) for communication by sensing and actuating. These medical devices generate data that may be further transmitted in the blockchain network.

IPFS cluster This part is responsible for the authentication of patients and their medical devices. The IPFS cluster not only authenticates the information, but also provides secure storage of information in the IoMT system. The IPFS cluster nodes facilitate the synchronization of data associated with medical device

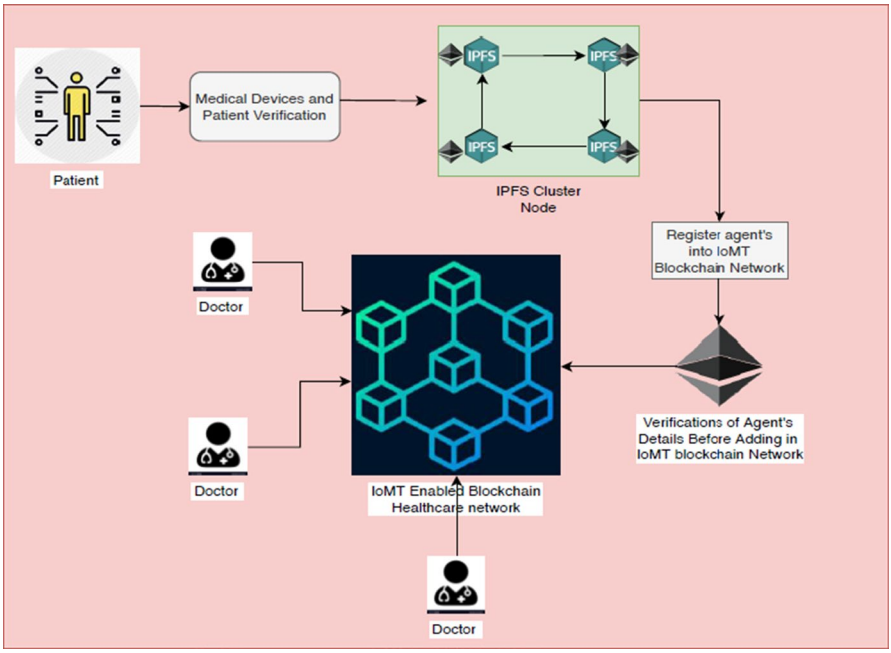

Fig. 3 Proposed model of blockchain for IoMT enabled healthcare

authentications and authorizations. Furthermore, the cluster nodes communicate with the smart contract to execute consensus, validation of transaction mapping, and block creation in IoMT blockchain network.

3.2 Communications types in proposed framework

In this proposed framework, there are primarily three types of communication which include medical-device-to-IPFS cluster node communication, IPFS cluster node-to-smart contracts communication, and smart contracts-to-blockchain network communication.

Medical-device-to-IPFS cluster node communication This communication is responsible for obtaining two different aims in the model. The first aim is to register the patients and their medical devices. The second aim is to authenticate the medical devices before communication in the main IoMT blockchain network.

IPFS cluster node-to-Smart contract communication This communication is responsible for synchronization of the authentication and authorization of the medical device's data and their mapping for ensuring privacy in IoMT blockchain network.

Smart contracts-to-blockchain network communication This communication is responsible to disseminate the information into the blockchain network after successful authentication and authorization to ensure secure transmission of data among different agents (patients' and doctors') in IoMT blockchain network. This communication ensures the privacy in IoMT blockchain network.

4 Security and privacy majors in IoMT enabled healthcare

4.1 IoMT devices authentication

Each IoMT device has a corresponding smart contracts enabled IPFS cluster node that is nearest to the devices and is used for the medical device registration and authentication in the IoMT blockchain network belonging to the specific patient. The medical devices initially register to smart contract enabled IPFS cluster node. once the medical device identity is preserved in the IPFS cluster node as a transaction then it is forwarded to create a block in the IoMT blockchain network. Finally, the blocks are disseminated among all other peers in IoMT blockchain healthcare network for further access. If any devices try to interact with the system then authentication is required and the device must have to provide its security credentials to the smart contracts enabled IPFS cluster node. The smart contracts validate the credentials provided, and if the credentials match then the devices are authenticated successfully. Otherwise, the devices will not be permitted to enter the IoMT blockchain network and will be rejected.

4.2 Security for IoMT devices

The proposed framework facilitates access control security to the medical devices in IoMT blockchain healthcare system. In this framework, only those devices can connect to the IoMT network that is registered using smart contract-enabled IPFS cluster nodes and are authenticated successfully. The medical devices not listed on the IPFS cluster node cannot authenticate and are not allowed to connect to the IoMT network. This will reduce the risks for the malicious medical device to interact with IoMT network.

4.3 Generation of keys for patients and medical devices

The proposed model uses the algorithm Elliptic Curve Digital Signature Algorithm (ECDSA) to create private and public keys for medical devices. The algorithm is selected based on a comparison of the various encryption algorithms presented by the [47, 48]. The ECDSA algorithm needs fewer resources than others and has the same level of security as Rivest–Shamir–Adleman (RSA). The same algorithm has been successfully used in the Bitcoin framework.

4.4 Functioning of IoMT devices

This subsection explains briefly the working of the proposed framework which is pictorially illustrated in Fig. 4. The proposed model consists of two main stages, the first level includes the initialization level that is responsible for patient's and their medical device registration and the second level responsible for patient medical device authentication. The patient initialization level facilitates new patient registration in the IoMT healthcare system and ensures unique identification for each patient. Once a new patient is registered with the IoMT healthcare system, the next level is to register their medical device. Smart medical devices are required to register with the IoMT healthcare network during the device registration process. To become part of the IoMT network, these medical devices are registered against their respective (already registered) patients. The registration process of the medical device is followed by the authentication step of the patient in which the medical devices get authenticated by smart contracts enabled IPFS cluster node. The registration level provides a certificate to the medical device for the authentication on the IPFS cluster node. At this level, the medical devices will be analyzed with their authentication certificate. If all authentication requirements are matched, then the medical device can become a part of the IoMT healthcare system. This level guarantees that only authorized medical devices are allowed to access the IoMT healthcare system. List of the symbol which is used as a notation in the proposed model for the functioning of IoMT devices is shown in Table 2.

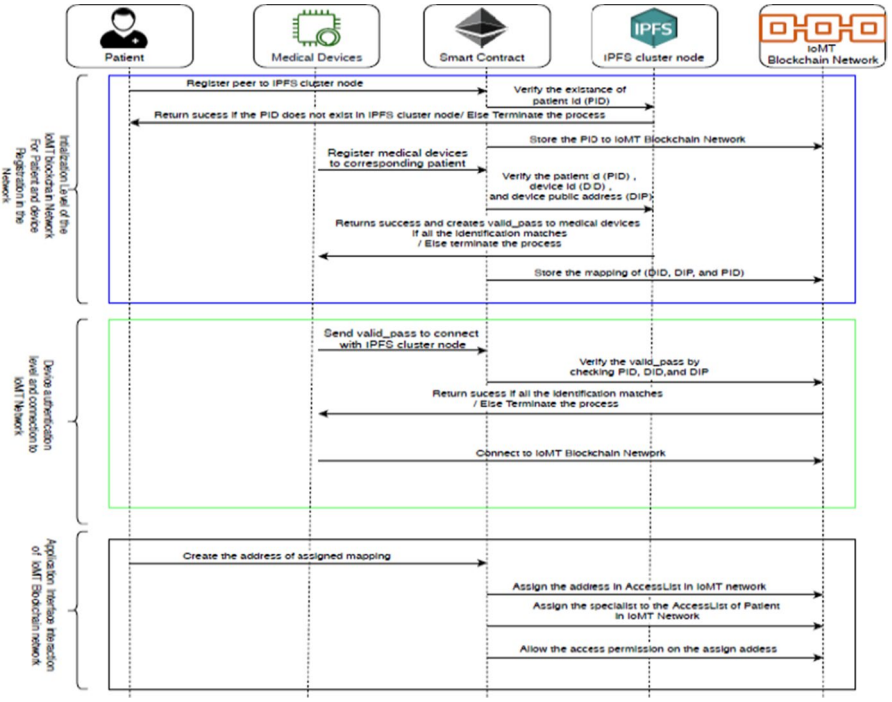

Fig. 4 Sequence diagram of proposed framework for IoMT enabled healthcare

Table 2 List of symbol and their associated meaning

List of symbol	Meaning of the symbol
IPFSC	Interplanetary File System cluster
$IPFSC_{IK}$	IPFS cluster private key
$IPFSC_{PK}$	IPFS cluster public key
PID	Patient Id
$IPFSC_{IK}(PID)$	Certificate of the patient's
DID	Device Id
DIP	Device public address stored in IoMT network
DID_{IK}	Device private key
DID_{PK}	Device public key stored on IPFS cluster node
Agents	Peers or nodes in blockchain network
(DID, DIP, PID)	Mapping of patient and devices or valid_pass or certificate to devices
DID_{IK} (DID, DIP, PID)	Registration_token of devices

4.4.1 Patient and their medical devices initialization level

At this level, the patients' and their respective medical devices are registered with the IoMT healthcare decentralized network. Registration is necessary to enable the re-authentication of the medical devices. It is further categorized into two different levels: patient and medical device registration level. These two levels are described below:

Patient registration level In this level, a patient's gets registered into the IoMT enabled blockchain network with a unique Patient ID (*PID*). After the successful registration process, the patient's details are stored into the IPFS cluster node, and a block is created for the same and is disseminated among all the agents (peers) into the IoMT blockchain network. In addition, the IPFS cluster node creates a new certificate for the newly registered patient within the IoMT network. Moreover, the steps of the registration process are pictorially illustrated in Fig. 5.

Algorithm 1: Algorithm for Patient Registration into Blockchain Network

```
Input: Blockchain
Output: Add Agents in Access List
//check the provided patient_id (PID) exist in the
  IPFSC or not//
if valid_Agent((PID), IPFSC)==true then
    //Provided (PID) exist in the IPFSC
    return error();
else
    //Register the PID into IPFSC and blockchain
      network
    Agent_registration((PID),IPFSC, blockchain)
end
```

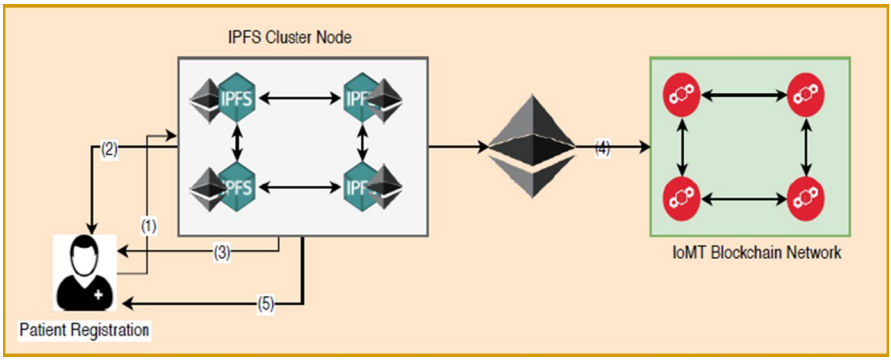

Fig. 5 Patient's registration in IoMT blockchain network

1. First, the smart contract creates a unique ID (*PID*) for the new patient, consisting of the name of the patient and the hash of the first 5 digits of the current timestamp. This obtained PID gets stored into the IPFS cluster node encrypted by their public key, this randomization of timestamp ensures the uniqueness of *PID*. Furthermore, the same smart contracts disseminate *PID* details into the IoMT healthcare network as a transaction T1.

$$PID = \text{PatientName} + \text{SHA (timestamp)}$$
$$T_1 = IPFSC_{PK}(PID)$$

   Here, $IPFSC_{PK}$ is the IPFS cluster node public key, which is applied to encrypt the *PID*.
2. The smart contract-enabled IPFS cluster nodes first check the transaction by verifying the given *PID*, if it exists in IPFS cluster node or not. This rule for the initialization level of patient registration is described in Algorithm 1.
3. If the *PID* exists in the IPFS cluster and IoMT healthcare blockchain network, then the transaction will not be permitted, and the error message will be sent during the process of registration.
4. If the *PID* does not exist in the IPFS cluster node, the smart contract creates a transaction and construct a block for the same. Furthermore, the created block will be disseminated among other agents (peers) in the IoMT blockchain network and the same will be stored in the IPFS cluster node as an encrypted format.
5. If the agents (peers) is registered successfully in the smart contract-enabled IPFS cluster node, then in reply to that it generates a certificate to the registered agents *PID* by using IPFSC_{IK}, private key and creates another transaction (T_2) into IoMT healthcare network.

$$T_2 = IPFSC_{IK}(PID).$$

   This certificate $\text{IPFSC}_{IK}(PID)$ is disseminated among all other peers in the IoMT blockchain network.

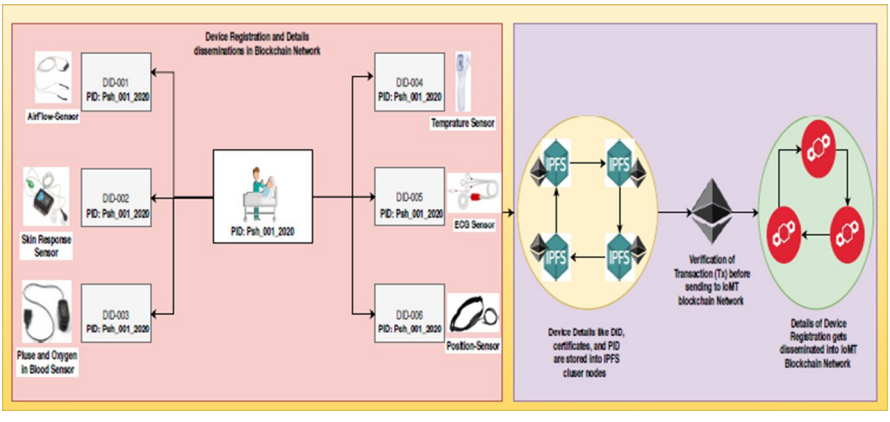

Fig. 6 Patient and their corresponding devices naming convention in IoMT blockchain network

Medical device registration Once the patient is registered successfully in the IoMT healthcare blockchain network, the next step is to register their corresponding devices into the IPFS cluster node which is shown in Fig. 6. In this level, a registration token is generated for each device. The registration token consists of device id (*DID*), device public address (*DIP*), and patient id (*PID*), and its mapping is denoted by $registration\_token = DID_{IK}(DID, DIP, PID)$, where DID_{IK} is private key of device.

The token is sent by the medical device along with the *PID* certificate to the IPFS cluster node, after that it is forwarded to the IoMT blockchain network. Then, the smart contract verifies the legitimacy of the PID certificate and the existence of the *PID* within the IoMT blockchain network. If the given certificate is authorized and the blockchain network preserve the *PID*, then smart contracts further verify the public key and the token which is shared on the blockchain network is valid or not. After the confirmations of successful matches, the smart contracts permit the transaction and it creates a block with the mapping of the device ID (*DID*), patient ID (*PID*), and device public address (*DIP*). The created block is then disseminated in the IoMT healthcare blockchain network. Finally, a valid pass certificate is created for the newly registered medical device. This certificate is disseminated among all the agents (peers) of the IoMT blockchain network. This certificate (valid pass) also gets stored into the IPFS cluster node for future authentication of the newly registered medical device. The valid pass is created with the above mapping device ID (*DID*), patient ID (*PID*), and device public address (*DIP*). The device registration is shown pictorially in Fig. 7.

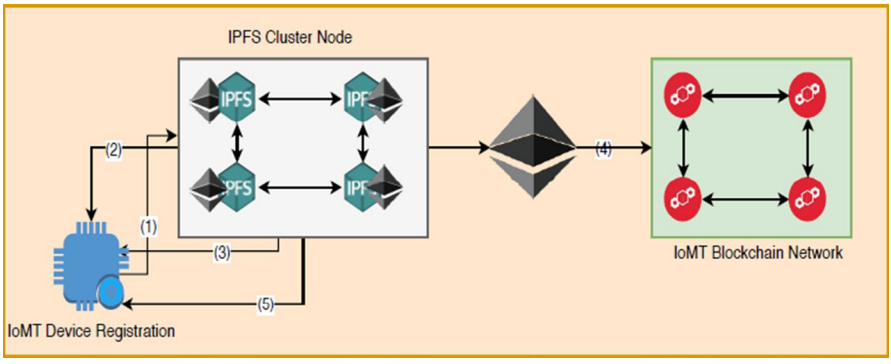

Fig. 7 Device registration in IoMT blockchain network

Algorithm 2: Algorithm for Device Registration into Blockchain Network

```
Input: PID
Output:
//check the provided patient_id (PID) exist in the
  IPFSC or not//
if valid_Agent((PID), IPFSC)==true then
    //check the provided device_id exist in the
      IPFSC or not//
    if valid_DID((DID), IPFSC)==true then
        //check the provided device public address
          exist in the IPFSC or not//
        if (device.pk == DIP) then
            map_patient(PID, DID, device.pk,
              blockchain)
else
    //if any of the above condition is not matched
      then return error//
    return error()
end
```

1. The device sends token together with the (*PID*) certificate to the IPFS cluster node through transaction T_3.

$$T_3 = IPFSC_{PK}(DID_{IK}(DID, DIP, PID), IPFSC_{IK}(PID)$$

$IPFSC_{PK}$ denotes the public key to the IPFS cluster node used for message encryption. By applying their private key $IPFSC_{IK}$, the cluster node checks the authenticity of the received message. The $IPFSC_{IK}$ (PID) certificate is also being verified by the smart contract, to complete the level of device registration.

$$IPFSC_{IK}(IPFSC_{PK}(DID_{IK}(DID, DIP, PID)), IPFSC_{IK}(PID))$$
$$= DID_{IK}(DID, DIP, PID), IPFSC_{IK}(PID)$$
$$IPFSC_{PK}(IPFSC_{IK}(PID)) = PID$$
$$DID_{PK}(DID_{IK}(DID, DIP, PID))$$
$$= DID, DIP, PID$$

2. The smart contract-enabled IPFS cluster node checks the existence of provided (*PID*) and (*DID*) off the chain and verifies the (*PID*), (*DID*). This rule for the device registration is described in Algorithm 2.
3. If the *PID* is not registered with the IPFS cluster node and the *DID* exists with the IPFS cluster node then the transaction cannot be processed further, and the registration process of the device will be simply terminated.
4. If the (*PID*) is registered in the IPFS cluster node, and *DID* is new for the mapping of *PID*, then in that case, smart contracts will verify the DID_{PK} and the token of

the device, if the match is found correct then the block is created with the mapping of (*PID*), (*DID*), (*DIP*) and disseminated in IoMT blockchain network.

5. Once the device is registered successfully into the IPFS cluster node. It generates a valid pass and authenticate the device and notify the device by creating transaction T4.

$$T_4 = DID_{PK}(IPFSC_{PK}(PID, DID, DIP))$$

When this transaction is received by the device the valid pass will be extracted from the transaction.

$$DID_{IK}(DID_{PK}(IPFSC_{IK}(PID, DID, DIP))) = IPFSC_{IK}(PID, DID, DIP)$$
$$valid\_pass = IPFSC_{IK}(PID, DID, DIP)$$

This valid pass is stored to authenticate the medical devices.

4.4.2 Device authentication level

Once the device and patient are registered into the IPFS cluster node, an authentication level is required for registered devices with specific patients [49]. The authentication process is performed by a deployed smart contract on the IPFS cluster node and the same is disseminated on IoMT blockchain network. The medical device sends the valid pass to the IPFS cluster node. The valid pass consists the mapping of (*PID*), (*DID*), (*DIP*). The valid_pass checks the following conditions for authenticating and allowing devices to communicate with the IoMT blockchain network, such as (1) (*PID*) existence in the blockchain, (2) (*DID*) existence in the blockchain, (3) the public address (*DIP*) used to validate the matches with stored public address (*DIP*), and (iv) a correct mapping existence between (*PID*), (*DID*), and (*DIP*). The authentication of the devices is shown pictorially in Fig. 8.

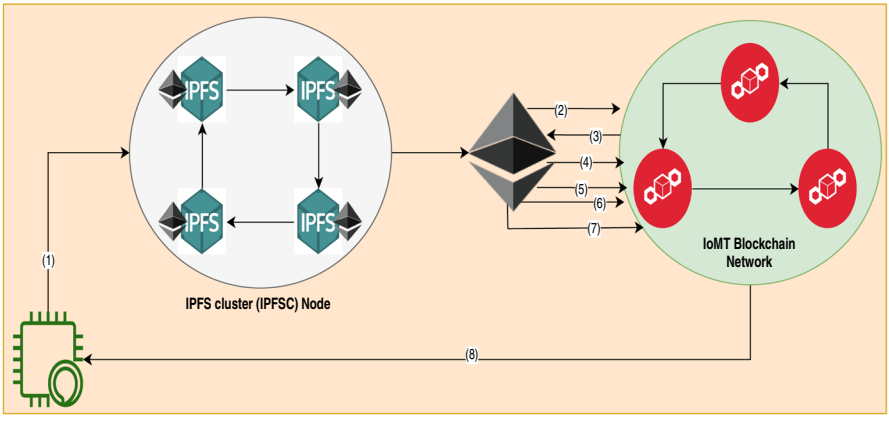

Fig. 8 Authentication of device on IPFS cluster by leveraging IoMT blockchain network

Algorithm 3: Algorithm for Device authentication into IoMT Blockchain Network

```
Input: PID,DID
Output:
//check the provided patient_id exist in the
  blockchain or not//
if valid_Agent((PID), blockchain)==true then
    //check the provided device_id exist in the
      blockchain or not//
    if valid_DID((DID), blockchain)==true then
        //check the provided device public address
          exist in the blockchain or not//
        if (device.pk == DIP) then
            //check the provided authentication
              mapping exist in the blockchain or
              not//
            if map_patient((PID), (DID), device.pk,
              blockchain) then
                //Device is Authenticated
                  Successfully//
else
    //if any of the above condition is not matched
      then return error//
    return error()
end
```

1. The medical device uses its private key to encrypts the valid pass and passes it to the IPFS cluster node by creating a transaction T5.

$$T5 = DID_{IK}(IPFSC_{IK}(PID, DID, DIP))$$

The deployed smart contracts on IPFS cluster node checks the legitimacy of the received transaction by using the device public key DID_{PK}.

$$DID_{PK}(DID_{IK}(IPFSC_{IK}(PID, DID, DIP))) = IPFSC_{IK}(PID, DID, DIP)$$

To check the extracted valid pass, IPFS cluster node uses its public key IPFSC_{PK}.

$$IPFSC_{PK}(IPFSC_{IK}(PID, DID, DIP)) = PID, DID, DIP$$

2. The smart contract matches the presence of *PID* in the IPFS cluster node and then disseminates it to the IoMT blockchain network. The device authentication rules are described in Algorithm 3.
3. If the presence of (*PID*) is not matched in IoMT blockchain network then the authentication process is terminated with an error, otherwise, the next stage of the process will continue.
4. The smart contract verifies the identity of the provided (*DID*) in the IPFS cluster and the IoMT network as well.

5. If the provided (*DID*) does not match in IPFS cluster node and the IoMT blockchain network, the process of medical device authentication will terminate with an error. Otherwise, it proceeds further for the next stage.
6. Now, the smart contract checks whether or not the given mapping (*PID, DID, DIP*) is correct.
7. If the provided mapping is not legitimate then the authentication of the device will be terminated, otherwise, it continues for the next stage process.
8. Finally, the device public key address (*DIP*) is matched with the stored key which is preserved at the time of registration in the IPFS clustered node. The same is checked with the IoMT network as well. If it is matched then the authentication process of the device will succeed, otherwise, authentication will terminate with an error.

5 Secure transmission of data in IoMT blockchain network

This section describes various security and privacy algorithms for the IoMT blockchain healthcare network to facilitate protected and authorized access to individual healthcare information.

5.1 Consensus deployment

The system constructs a transaction-based polynomial to implement conformance proof in the network. We assume that $T = \{T_1, T_2, T_, \ldots, T_m\}$ is the set of transactions. The hash of the transaction polynomial is constructed by computing $H_1(T_1), H_1(T_2), \ldots, H_1(T_m)$ and create a polynomial $f(q)$ of order m, such as $f(H_1(t_k)) = 0, k \in \{1, 2, 3, 4, \ldots, m\}$. Now, the polynomial can be formulated as

$$f(q) = (q - H_1(T_1))(q - H_1(T_2)) \ldots (q - H_1(T_m)) \tag{1}$$

The Eq.(1) can be rewritten as:

$$f(q) = q^m + a_{m-1}q^{m-1} + \cdots + a_1 q + a_0$$

where $[1, a_{m-1}, a_{m-2}, \ldots, a_0]$ are polynomial coefficient term. Then, the equation of $f(q) = 0$ can be transformed as:

$$q^m + a_{m-1}q^{m-1} + \cdots + a_1 q = -a_0 \tag{2}$$

Divide the Eq. (2) both side by -a_0 then it can be formulated as:

$$\frac{-1}{a_0}q^m + \frac{-a_{m-1}}{a_0}q^{m-1} + \cdots + \frac{-a_1}{a_0}q = 1 \tag{3}$$

Let $u_m = \frac{-1}{a_0}, u_{m-1} = \frac{-a_{m-1}}{a_0}, \ldots, u_1 = \frac{-a1}{-a0}$ constructs a new polynomial.

$$g(q) = u_m q^m + u_{m-1}q^{m-1} + \cdots + u_1 q \tag{4}$$

It can be described as $g(H_1(T_k)) = 1$, where $T_k \in T$. The vector U is defined as $U = \{u_1, u_2, \ldots, u_{m-1}, u_m\}$ and another vector h_i is defined as $h_i = [H_1(t_k), (H_1(t_k))^2, \ldots, (H_1(t_k))^{m-1}, (H_1(t_k))^m]$. Then, the inner product of both the vector 'u' and h_k is $u.h_k = 1$. The vector 'u' is a consensus vector and is checked each time when new index of block is created in the blockchain network. If more than $\frac{1}{3}$ of the agents (peers) verifies the new transaction (T_k) then it gets added to the new valid index block in the IoMT blockchain network.

5.2 Smart contracts algorithms for preserving privacy in IoMT blockchain network

The Algorithm 4 is responsible for adding agents (patients' and doctors') in IoMT enable healthcare network. The smart contracts verify the details of IPFS cluster node like valid patient's ID I(*PID*), device ID (*DID*), and device public address (*DIP*) along with their mapping.

Algorithm 4: Algorithm for adding Agents in IoMT Enabled Healthcare Blockchain Network

```
Input: string name, uint age, uint designation,
  string hash
Output: Add Agents in Access List
// check the sender of the message //
address addr=message.sender
//Define the Agent List//
address[] public patientList;
address[] public doctorList;
//Map the address of each agents using mapping
  functions//
mapping (address => patient) patientInfo
mapping (address => doctor) doctorInfo
//check the designation of agents and add into their
  respective List//
if (designation ==0) then
    patientInfo[addr].name = name;
      patientInfo[addr].age = age;
      patientInfo[addr].record = hash;
      patientList.push(addr)-1;
else if (designation ==1) then
    doctorInfo[addr].name = name;
      doctorInfo[addr].age = age;
      doctorList.push(addr)-1;
else
    Invalid addrsss of sender
    return;
end
```

In Algorithm 4, we have stored the mapping of patient's and doctor's using smart contract verification. In the application interface of the IoMT network, agents are divided into two different categories according to their designation. If the designation is equal to '*0*' then patient's detailed information is mapped with their name,

age, transaction hash, and their (PID, DID, DIP) mapping. If the designation is equal to '*1*' then the detail of the doctor will be recorded into the IoMT application interface. If the agents are not in any of the categories then the transaction will be discarded simply by the smart contracts.

Algorithm 5: Algorithm for Patient and Doctor Assignment

```
Input: Address of an Agent, uint creditPool
Output: Provide Access Permission
// check the sender of the message //
address addr=message.sender
//Assign the access List of Agents
address[] doctorAccessList;
address[] patientAccessList;
//Map the address of each agents using mapping
 functions//
mapping (address => patient) patientInfo
mapping (address => doctor) doctorInfo
if ((addr == doctorInfo[addr]) || (addr ==
 patientInfo[addr] )) then
    // check the balance of Sender Address//
    require(msg.value==2 ether)
    // increase the value of credit Pool by 2//
    creditPool+=2
    //Assinged a patient to a doctor//
    doctorInfo[addr].patientAccessList.push(msg.sender)
    //Assinged a doctor to a patient //
    patientInfo[msg.sender].doctorAccessList.push(addr);
else
    Invalid addrsss of sender
    return;
end
```

As shown in Algorithm 5, the two lists are initialized like doctorAccessList and patientAccessList. Both the lists are used for the assignment of a patient to a doctor and vise versa. For each assignment of a patient to a doctor, creditpool value gets increased by 2, and also the balance of the mapping address (PID, DID, and DIP) is checked. From the mapping of accessList, both patient and doctor's can verify to each other which is shown in Algorithm 6. The balance required in the patient accounts, to allocate the doctor.

Algorithm 6: Algorithm to view patient list and doctor list

```
Input: Address of an Agent, uint creditPool
Output: Provide Access Permission
// check the sender of the message //
address addr=message.sender
//Map the address of each agents using mapping
 functions//
mapping (address => patient) patientInfo
mapping (address => doctor) doctorInfo
if (addr == patientInfo[addr]) then
    // get the list of doctor assigned to a patient //
    return patientInfo[addr].doctorAccessList;
else if (addr == doctorInfo[addr]) then
    // get list of patient assigned to a doctor//
    return doctorInfo[addr].patientAccessList;
else
    Invalid addrsss of sender
    return;
end
```

As shown in Algorithm 6, the patient can see the assigned doctor list for their treatments. The assigned address is matched against the list of address recorded in Algorithm 4. If the address is matched then the patient can see the doctor list and vise versa, else the request will be discarded.

Algorithm 7: Algorithm to Revoke the Access of Patient and Doctor

```
Input: Address of an Agent, uint creditPool
Output: Provide Access Permission
// check the sender of the message //
address addr=message.sender
//Map the address of each agents using mapping
 functions//
mapping (address => patient) patientInfo
mapping (address => doctor) doctorInfo
if (addr == patientInfo[addr]) then
    // Revoke the access of Patient from the
      PatientAccessList //
    delete patientInfo[addr].patientAccessList
    msg.sender.transfer(2 ether);
    creditPool -= 2;
else if (addr == doctorInfo[addr]) then
    // Revoke the access of Doctor from the
      DoctorAccessList//
    delete doctorInfo[addr].doctorAccessList
    msg.sender.transfer(2 ether);
    creditPool -= 2;
else
    Invalid addrsss of sender
    return;
end
```

The Algorithm 7 shows the revoke access of patient and doctors. If the malicious activities are performed by any of the agents then their access permissions will be immediately denied and also the agents will be removed from the accessList and address mapping list. The underlying approach ensures security in the system. The revoke permissions are also applicable when agents (patients or doctors) are willing to leave the IoMT enabled blockchain healthcare network.

Algorithm 8: Algorithm to Access the individual Agents

```
Input: Address of an Agent, uint creditPool
Output: Provide Access Permission
// check the sender of the message //
address addr=message.sender
//Map the address of each agents using mapping
 functions//
mapping (address => patient) patientInfo
mapping (address => doctor) doctorInfo
if (addr == patientInfo[addr]) then
    //Access the list of patient in blockchain
      network//
    return patientList;
else if (addr == doctorInfo[addr]) then
    // Access the list of doctor in blockchain
      network//
    return doctorList;
else
    Invalid addrsss of sender
    return;
end
```

As shown in Algorithm 8, the details of doctors can be seen in the blockchain network. Similarly, the patient details can also be verified and seen in the blockchain network. The number of agents (doctors' or patients') and their details are shown in the application interface to ensure transparency in the system. Besides, the details are only shown to the registered agents in the IoMT network. The unauthorized agents are not permitted to access these details from the IoMT enabled healthcare blockchain network.

6 Implementation

This section provides an experimental study of the proposed privacy-enhancing technique using blockchain technology for the Internet of Medical Things (IoMT) enabled healthcare system. The experiment is carried out using the node js, solidity version 0.4.26 programming language, and remix IDE. The smart contracts are deployed on the IPFS cluster node, which directly interacts with the application interface and ensures device verification and their address storage.

Fig. 9 List of operations in IoMT blockchain healthcare Network

The performance of the proposed model is evaluated using ethereum ropsten network and metamask. Experiments are carried out on Tyrone PC run by Intel(R) Xeon(R) Silver 4114 CPU @ 2.20 GHz (2 processors), 128 GB RAM, and 2 TB hard disk.

The Fig. 9 shows list of operation viz., ***add_agent, permit_access, remove patient, revoke_access, doctorList, get_accessed_doctorlist_for_patient, get_accessed_patientlist_for_doctor, get_doctor, get_doctor_list, get_hash, get_patient, get_patient_doctor_name, get_patient_list, and patientList*** in IoMT blockchain healthcare network. The application interface is designed to interact with the healthcare network and get different services from the IoMT network. These services are allowed only for the registered agents in the system.

Adding agents into the IoMT system and providing permission to the agents are shown in Fig. 10. The details of the patient are added into the system, and permissions are granted to the patient address (*PID*). The patient's details are stored in the system for providing further access in the IoMT blockchain healthcare network. The ailment details of the patients are stored in hash, to maintain the scalability in the system and reduce the size of the blockchain network.

Fig. 10 Adding agents and granting permission in IoMT blockchain network

Fig. 11 Removing and revoking access permission of the agent from IoMT blockchain network

Fig. 12 List of patient allocated to the doctor and their details

Fig. 13 Details of patient using hash value in IoMT blockchain network

Fig. 14 Gas limits for access permissions in IoMT blockchain network

Figure 11 shows the two different types of access in the system like remove patient and revoke access of the agents. The remove patient's permissions are given to the doctor's, as the treatment of the patients is finished, the doctor can remove the patient's from their list. But the patient's exists in the IoMT blockchain healthcare after these operations. The revoke permissions are provided in the system to get back all the access control of specific agents. Once the revoke transaction gets executed for an agent, they are having no more access to the system.

Figure 12 shows the list of doctor and their details in the IoMT blockchain application interface. The get doctor list operation is executed to see the registered doctors' in the system. The individual doctor's details can be seen by their listed address in the IoMT application interface.

Figure 13 shows the patient-doctor mapping where a list of patients are allocated to an individual doctor. From the provided interface doctor can see the details of the

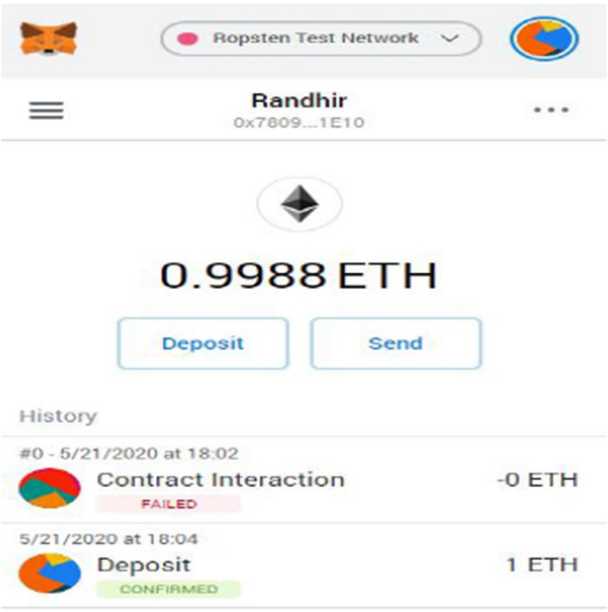

Fig. 15 Detailed history of transaction in IoMT blockchain network

Txn Hash	Block	Age	From		To	Value	[Txn Fee]
0x6041cba0f2aa396...	7727710	22 mins ago	0x64e46314dff7c98...	OUT	0x45b1f019fda513fd...	0 Ether	0.000032115
0xd04806682d2938...	7727681	30 mins ago	0x64e46314dff7c98...	OUT	Contract Creation	0 Ether	0.0004783527
0x76de7c0dd2370c...	7727675	33 mins ago	0x64e46314dff7c98...	OUT	0x0841b3e08436e7...	1 Ether	0.00021
0x6218ddbb3f0f42a...	7727350	2 hrs 1 min ago	0x64e46314dff7c98...	OUT	Contract Creation	0 Ether	0.0008680074
0xeafe363c0a03c03...	7727308	2 hrs 13 mins ago	0x64e46314dff7c98...	OUT	Contract Creation	0 Ether	0.0004783527
0xcc405f06d5c554f...	7727298	2 hrs 16 mins ago	0x81b7e08f65bdf56...	IN	0x64e46314dff7c98...	1 Ether	0.0000063
0x94e29a9e06d2c1...	7727298	2 hrs 16 mins ago	0x81b7e08f65bdf56...	IN	0x64e46314dff7c98...	1 Ether	0.0000063
0xc1453462ab063f0...	7727298	2 hrs 16 mins ago	0x81b7e08f65bdf56...	IN	0x64e46314dff7c98...	1 Ether	0.0000063
0x0ae637f79e57495...	7727297	2 hrs 17 mins ago	0x81b7e08f65bdf56...	IN	0x64e46314dff7c98...	1 Ether	0.0000063
0x4e6a9c37e1ca39...	7727297	2 hrs 17 mins ago	0x81b7e08f65bdf56...	IN	0x64e46314dff7c98...	1 Ether	0.0000063

Fig. 16 List of doctor and their details in the IoMT blockchain network

patient before starting the treatment. The doctors are allowed to remove the patient's from their list by a provided interface which is shown in Fig. 11.

The doctors are allowed to see the ailment details of an individual patient's from their address as shown in Fig. 14. The ailment details module is accessible to the

registered doctors in the system. The list of address of patients can be accessed by a doctor from the provided interface shown in Fig. 13.

Figure 15 shows the ether requirement while getting access permissions to the IoMT blockchain network. If the required amount of ether is available in the wallet then permission will be granted otherwise the requested transaction of permission grant will be failed.

Figure 16 shows a list of transaction in IoMT blockchain network. The results are shown in ethereum Ropsten network. The history shows the ether consumed in the network during the transaction process. Smart contract deployment results are shown as a transaction in the history list. This result ensures auditability of each transaction owing to the property of immutability in ethereum Ropsten network.

6.1 Evaluation of proposed framework with the attacks and security

This section presents the proposed framework conformity with the security requirements. These requirements include integrity [50], confidentiality [51], nonrepudiation [52], identification [53], mutual authentication [54], and authentication [55].

6.1.1 Integrity of transaction

To manage the data integrity in the IoMT network, each transaction is signed using the private key of the devices and mapped with *DID, DIP, PID* on the ethereum network. The process of encryption is performed using ECDSA algorithm which ethereum supports. The transactions contain the patient's data, a hash of corresponding data and the sender signature [52]. The receiving nodes verify it by checking the address in the accessList and the generated hash of the data with their mapping.

6.1.2 Identification of peers and devices

To achieve the security of the proposed framework, every device should be registered and must have a device Id (*DID*). Similarly, each patient's must have their unique patient Id (*PID*). The mappings of the device and the patients are securely stored in the network. Thus, it becomes easy to recognize the devices of patients [52].

6.1.3 Verification of non-repudiation

Each transaction in the network is signed by their device private key associated mapping verification. Therefore, the sender can't deny the performed transaction in the IoMT network [51].

6.1.4 Mutual authentication of Devices

For mutual authentication during information sharing in the network, the proposed framework generates valid pass to each medical device those are the parts of the network. This enhances the trust among other peers while communicating with each other. This valid pass is created by a private key of the IPFS cluster node ($IPFSC_{IK}$). If any existing nodes (peers) wants to create a fake valid pass, it requires IPFS cluster node private key ($IPFSC_{IK}$) which is secured and difficult to retrieve [56].

6.1.5 Authentication of peers

For authentication of the medical device in the network, each device has to register in the IPFS cluster node. If the device is already registered in the network then their public address and mappings are checked against authentication. The smart contracts verify the legitimacy of the existing devices then will permit the devices to create transactions in the IoMT network.

6.1.6 Prevention of spoof attack

To generate the spoofing attack, the attacker must acquire a medical device id, patient id, and medical device private key. If somehow the attacker gets the patient id (*PID*) and device id (*DID*) then still the private key of the device (DID_{IK}) will be unknown to the attacker to spoof the IoMT network [51].

6.1.7 Prevention of Sybil attack

In the case of the Sybil attack, the attacker creates fake identities in the system to send a false transaction in the system. In the IoMT network, medical devices are not permitted to associate with multiple *PID* thus, each medical devices can associate itself with only one *PID* that is already registered into the IPFS cluster node. The transactions created by the medical devices are signed with the device private key (DID_{PK}) [57]. Thus, creating fake identities and injecting false transactions into the IoMT network is possibly difficult and reduced to zero.

6.1.8 Prevention of replay attack

In the IoMT network, all the created transactions are assigned with a transaction id and timestamp. Thus, the replay attack will be rejected from the IoMT network owing to the already stored transaction id [58].

6.1.9 Prevention of substitution attack

In the IoMT network, all the transactions are mapped with the patient Id (PID), device id (DID), and device public address (DIP). Thus, if any malicious nodes attack the network and want to change the message, the mapping will be checked

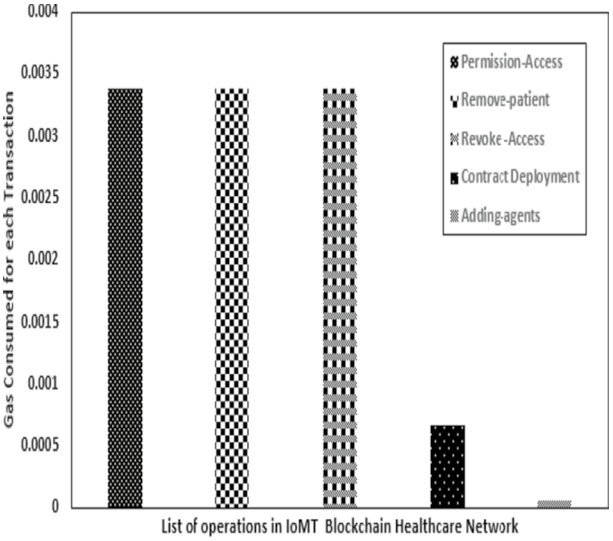

Fig. 17 Gas consumed for different operations of IoMT blockchain healthcare network

during the device authentication process and it will be simply discarded or rejected [59].

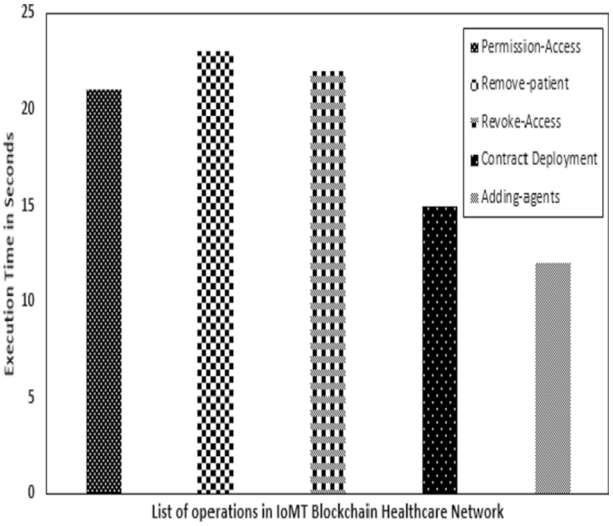

Fig. 18 Execution time in seconds for different operations of IoMT blockchain healthcare network

7 Result analysis

In this section, we first execute the primary operations of the IoMT blockchain healthcare network. The result of these primary operations is discussed in terms of execution time and gas consumption. We evaluated our model in terms of execution time and gas consumption with 25 peers in the ethereum Ropsten network. Figure 17 shows the result of gas consumption for different operations in the IoMT network. The result shows the cost of processing of different operations, for example, permission-access, remove-patient, revoke-access, contract deployment, and adding-agents in terms of gas consumption. The access permissions like permission-access, remove-patient, revoke-access consume high gas consumption rather than smart contract deployment and adding agents to the IoMT enabled blockchain healthcare network.

Figure 18 shows the execution time for the individual operations in the IoMT network. The maximum time is taken in the network to remove patient from the doctor list due to the mapping process of patient-doctor allocation and search of the patient's details in the accessList. The time required to grant and revoke the permissions takes almost similar time in the IoMT network due to address-based permissions. Moreover, the smart contracts deployment and the adding agent take less time than other operations in the IoMT network.

To preserve privacy in the IoMT network each peer must have to register against the malicious behavior. The registration time for a varying number of peers is shown in Fig. 19. It can be observed that registration time increases as the number of peers increases in the IoMT network.

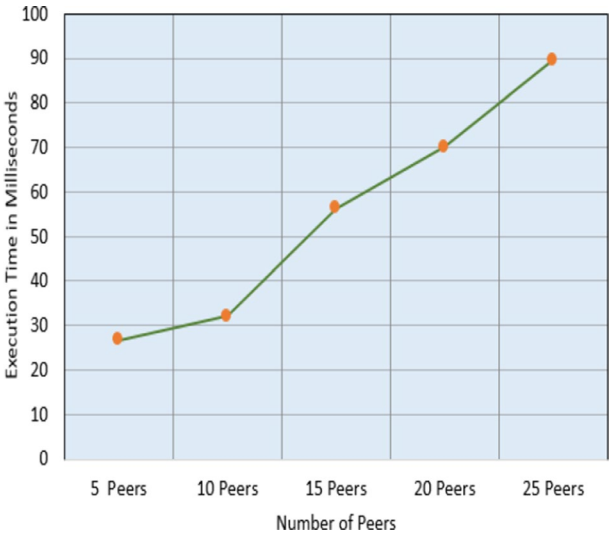

Fig. 19 Execution time analysis for registration of varying number of pees on the IoMT blockchain network

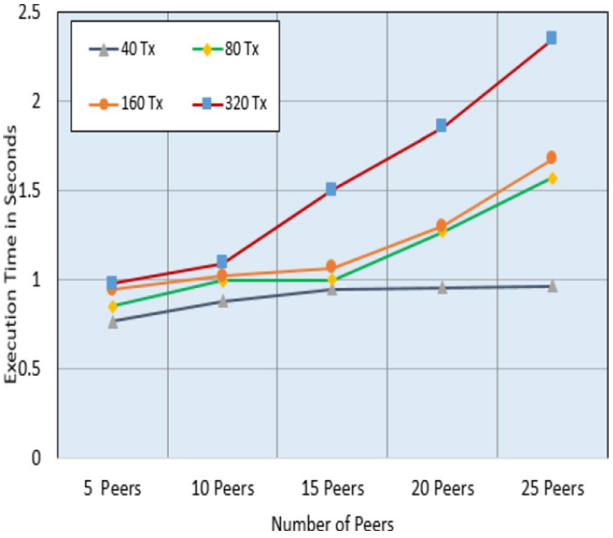

Fig. 20 Execution time analysis for upload of varying sizes of transaction (Tx) on IPFS secured storage layer

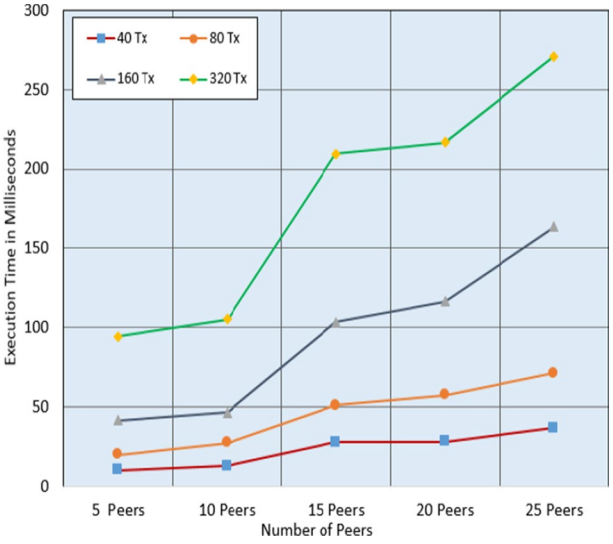

Fig. 21 Execution time analysis of non-repudiation for varying number of Tx with varying number of peers

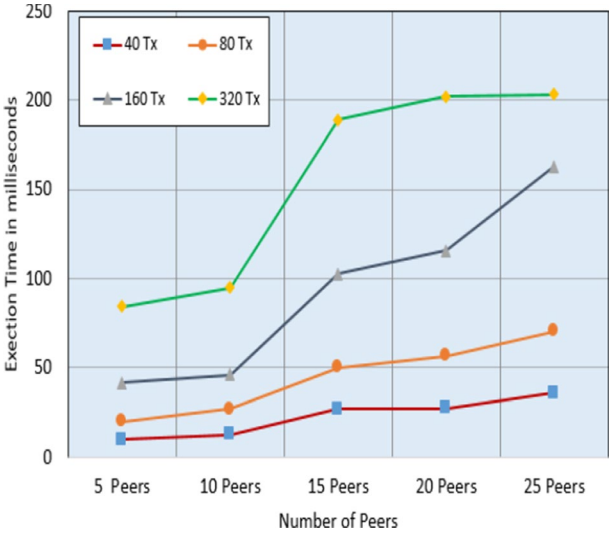

Fig. 22 Execution time analysis of block mining for varying number of transactions and peers

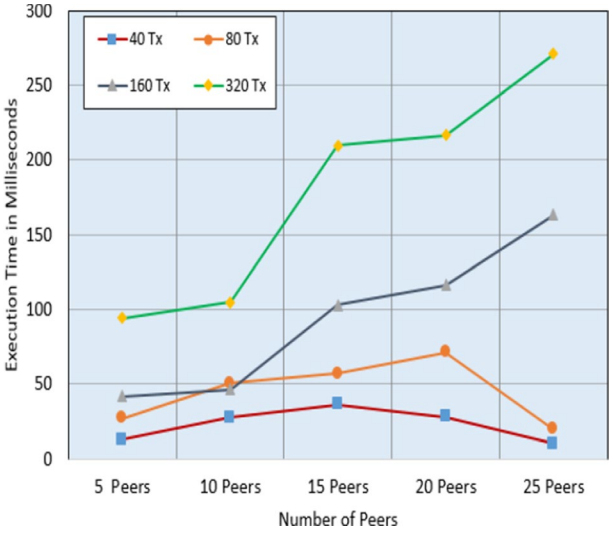

Fig. 23 Execution time analysis of block creation for varying number of transactions and peers

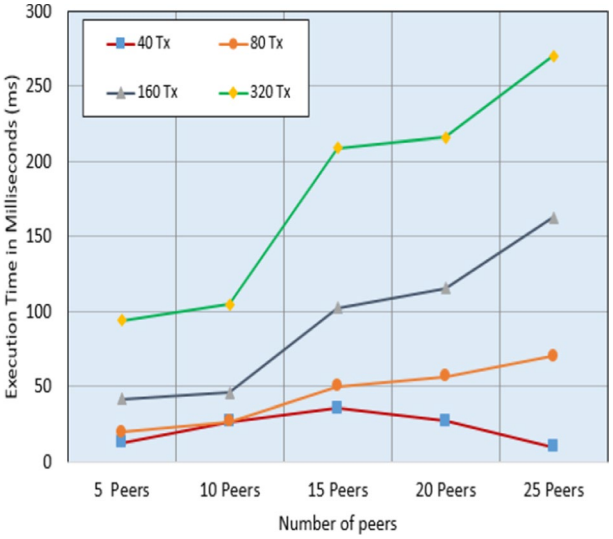

Fig. 24 Execution time analysis of block access for varying number of transactions and peers

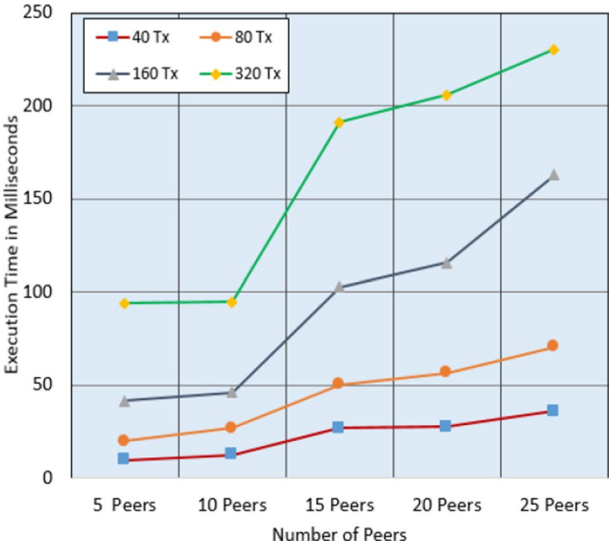

Fig. 25 Execution time analysis of contract deployment for varying number of transactions and peers

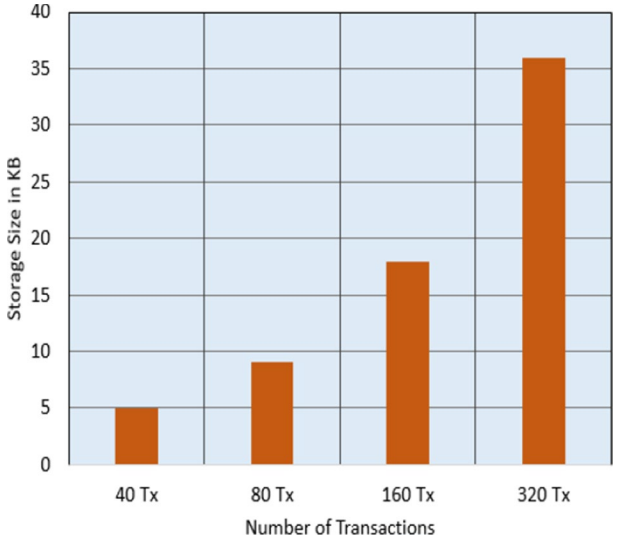

Fig. 26 Execution time analysis of off-chain storage for varying number of transactions and peers

Table 3 Comparison of proposed model with existing IoMT-based healthcare system

Authors	Year	Scalability	Security	Privacy	Off-chain	Decentralized
Alsubaei et al. [8]	2018	×	✓	×	×	×
He et al. [21]	2018	×	×	✓	×	×
Qureshi et al. [45]	2018	×	✓	✓	×	×
Laplante et al. [16]	2018	×	✓	×	×	×
Tewari et al. [22]	2019	×	✓	✓	×	×
Sun et al. [10]	2019	×	✓	✓	×	×
Haoyu et al. [44]	2019	×	✓	×	×	×
Han et al. [46]	2019	×	✓	✓	✓	×
Al-Turjman et al. [17]	2019	×	✓	✓	×	×
Sun et al. [43]	2019	×	✓	✓	×	×
Kotronis et al. [2]	2019	×	✓	×	×	×
Hathaliya et al. [18]	2020	×	✓	✓	×	×
Yaacoub et al. [23]	2020	×	✓	×	×	×
Farouk et al. [12]	2020	×	✓	✓	✓	×
Proposed model	2020	✓	✓	✓	✓	✓

Figure 20 shows the upload time on IPFS secure storage layer for a varying number of transactions and varying number of peers. It can be seen that the upload time increases as the number of transactions increases.

Figure 21 shows the signing time of a varying number of transactions with a varying number of peers to ensure the non-repudiation in the IoMT network. It can be seen that as the number of peers increases the signing time increases, respectively.

Figure 22 shows the mining time for a varying number of transactions validation. The mining process ensures consistency in the network. It can be observed that, mining time increases as the number of transactions increases along with a varying number of peers. The block creation and access time are shown in Figs. 23 and 24. The block creation time for 40Tx, 80Tx, and 160 Tx is almost similar for 10 peers. The block access time for 40Tx and 80Tx is almost similar to up to 15 peers. However, the block creation and access time increase as the number of transactions and peers increases (more than 15 peers) in the IoMT network.

Figure 25 shows the contract deployment time in IoMT network for varying size of transactions. The deployment time increases as the number of transactions increases.

Figure 26 shows the actual storage of the size of the transaction in KB for a varying number of transactions in the IoMT network. The storage size is computed with IPFS secure and distributed storage layer. The off-chain operations are performed to make the framework more scalable. It can be observed that storage size increases as the number of transactions increases.

7.1 Advantages and challenges of proposed framework

There are many advantages of the proposed framework. First, the blockchain-based infrastructure ensures decentralization in the system. Second, the registration-based security model is presented for the patient and their medical devices. Third, the access control is designed and implemented using consortium blockchain to ensure privacy in the IoMT network. However, few challenges have been identified such that building a distributed cluster using IPFS and also maintaining more devices in the system takes more computational time.

8 Comparison analysis of the proposed model

This Section describes the comparison of the proposed model with existing work in IoMT enabled healthcare system. Table 3 shows a comparative analysis of the proposed model with different parameters. Most of the existing work is designed and implemented based on the centralized infrastructure where security and privacy issues arise. The proposed model improves the existing challenges and enhances the working of the IoMT healthcare network. Our model is designed and implemented based on distributed off-chain storage which is highly secure and preserves privacy. The proposed framework uses an IPFS cluster which makes IoMT healthcare systems more scalable and facilitates secure access to patient data.

9 Conclusion

This paper emphasizes privacy, security, and storage management in IoMT infrastructure and how blockchain can provide a distributed platform to address these issues. To address the issues of the current IoMT system, this paper proposes towards design and implementation of security and privacy of IoMT by leveraging blockchain and IPFS technology. For the security in the IoMT network, the framework is divided into two different levels such as initialization level and authentication level. In the initialization level patient and their specific devices gets registered into the smart contract enabled-IPFS cluster node and their patient Id (*PID*), device Id (*DID*) are disseminated into the blockchain network. The authentication level includes the mapping of patient Id (*PID*), device Id (*DID*) and device public address (*DIP*) of the device and disseminated into IoMT blockchain network. Also, the IPFS cluster node provides a distributed off-chain data storage layer that fulfills the demands of secure storage management. To preserve privacy in the IoMT network, patients and their specific device details are sent as a transaction to the blockchain network after successful registration and authentication. Moreover, the framework does not rely on a third-party (as done in cloud-based systems) and provides fair service to authorized agents (peers). Furthermore, the proposed framework is efficient in terms of the security and privacy of device-generated medical data. In the future, we plan to extend this implemented work in the real-time application for the IoMT healthcare system by adding a large number of agents (peers) and their specific medical devices with various statistical tests.

References

1. Statista Research Department, "Internet of things—number of connected devices worldwide 2015–2025," Nov 27, 2016, [Online; accessed 11-May-2020]. https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/
2. Kotronis C, Routis I, Politi E, Nikolaidou M, Dimitrakopoulos G, Anagnostopoulos D, Amira A, Bensaali F, Djelouat H (2019) Evaluating Internet of Medical Things (IOMT)-based systems from a human-centric perspective. Internet of Things 8:100125
3. Digiteum, "Internet of medical things and medical software development," 2020, [Online; accessed 5-June-2020]. https://www.digiteum.com/internet-medical-things-medical-software-development
4. Patel N (2017) Internet of things in healthcare: applications, benefits, and challenges," Internet: https://www.peerbits.com/blog/internet-of-things-healthcare-applications-benefits-andchallenges.html. Accessed 21 March 2019
5. deloitte, Medtech and the internet of medical things. 2018, [Online accessed 09-May-2020]. [Online]. https://www2.deloitte.com/global/en/pages/life-sciences-and-healthcare/articles/medtech-internet-of-medical-things.html
6. Hofdijk J, Séroussi B, Lovis C, Sieverink F, Ehrler F, Ugon A (2016) Transforming healthcare with the internet of things. In: Proceedings of the EFMI Special Topic Conference 2016
7. Rodrigues JJ, Segundo DR, Junqueira HA, Sabino MH, Prince RM, Al-Muhtadi J, De Albuquerque VHC (2018) Enabling technologies for the internet of health things. IEEE Access 6:13129–13141
8. Alsubaei F, Abuhussein A, Shiva S (2017) Security and privacy in the internet of medical things: taxonomy and risk assessment. In: 2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops). IEEE, pp 112–120

9. Khalid U, Asim M, Baker T, Hung PC, Tariq MA, Rafferty L (2020) A decentralized lightweight blockchain-based authentication mechanism for IoT systems. Cluster Comput 1–21
10. Sun W, Cai Z, Li Y, Liu F, Fang S, Wang G (2018) Security and privacy in the medical internet of things: a review. Secur Commun Netw 2018
11. Fan S, Song L, Sang C (2019) Research on privacy protection in IoT system based on blockchain. In: International Conference on Smart Blockchain. Springer, pp. 1–10
12. Farouk A, Alahmadi A, Ghose S, Mashatan A (2020) Blockchain platform for industrial healthcare: vision and future opportunities. Comput Commun
13. Aileni RM, Suciu G (2020) IoMT: a blockchain perspective. In: Decentralised internet of things. Springer, Berlin, pp 199–215
14. Banerjee M, Lee J, Choo K-KR (2018) A blockchain future for internet of things security: a position paper. Digital Commun Netw 4(3):149–160
15. Aloqaily M, Al Ridhawi I, Salameh HB, Jararweh Y (2019) Data and service management in densely crowded environments: challenges, opportunities, and recent developments. IEEE Commun Mag 57(4):81–87
16. Laplante PA, Kassab M, Laplante NL, Voas JM (2017) Building caring healthcare systems in the internet of things. IEEE Syst J 12(3):3030–3037
17. Al-Turjman F, Nawaz MH, Ulusar UD (2019) Intelligence in the internet of medical things era: a systematic review of current and future trends. Comput Commun
18. Hathaliya JJ, Tanwar S (2020) An exhaustive survey on security and privacy issues in healthcare 4.0. Comput Commun 153:311–335
19. Mahmoud R, Yousuf T, Aloul F, Zualkernan I (2015) Internet of things (IoT) security: current status, challenges and prospective measures. In: 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST). IEEE 2015, pp 336–341
20. Aman MN, Chua KC, Sikdar B (2017) Mutual authentication in IoT systems using physical unclonable functions. IEEE Internet of Things J 4(5):1327–1340
21. He D, Ye R, Chan S, Guizani M, Xu Y (2018) Privacy in the internet of things for smart healthcare. IEEE Commun Mag 56(4):38–44
22. Tewari A, Gupta B (2018) Security, privacy and trust of different layers in Internet-of-Things (IoTs) framework. Future Gener Comput Syst 108:909–920
23. Yaacoub J-PA, Noura M, Noura HN, Salman O, Yaacoub E, Couturier R, Chehab A (2020) Securing internet of medical things systems: limitations, issues and recommendations. Future Gene Comput Syst 105:581–606
24. Abie H, Balasingham I (2012) Risk-based adaptive security for smart IoT in ehealth. In: Proceedings of the 7th International Conference on Body area Networks, pp 269–275
25. Savola RM, Savolainen P, Evesti A, Abie H, Sihvonen M (2015) Risk-driven security metrics development for an e-health IoT application. In: Information security for South Africa (ISSA). IEEE 2015, pp 1–6
26. Russell B, Garlati C, Lingenfelter D (2015) Security guidance for early adopters of the internet of things (IoT). White paper, Cloud Security Alliance
27. OWASP T, list 2013: https://www.owasp.org/index.php. Top_10_2013-Top_10, 10
28. Alsubaei F, Abuhussein A, Shandilya V, Shiva S (2019) Iomt-saf: Internet of medical things security assessment framework. Internet of Things 8:100123
29. Islam SR, Kwak D, Kabir MH, Hossain M, Kwak K-S (2015) The internet of things for health care: a comprehensive survey. IEEE Access 3:678–708
30. Blowers M, Iribarne J, Colbert E, Kott A (2016) The future internet of things and security of its control systems. arXiv preprint arXiv:1610.01953
31. Mohsin M, Sardar MU, Hasan O, Anwar Z (2017) Iotriskanalyzer: a probabilistic model checking based framework for formal risk analytics of the internet of things. IEEE Access 5:5494–5505
32. Park KC, Shin D-H (2017) Security assessment framework for IoT service. Telecommun Syst 64(1):193–209
33. Perera C, McCormick C, Bandara AK, Price BA, Nuseibeh B (2016) Privacy-by-design framework for assessing internet of things applications and platforms. In: Proceedings of the 6th International Conference on the Internet of Things, pp 83–92
34. McMahon E, Williams R, El M, Samtani S, Patton M, Chen H (2017) Assessing medical device vulnerabilities on the internet of things. In: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI). IEEE, pp 176–178

35. Zhang B, Zou Z, Liu M (2011) Evaluation on security system of internet of things based on fuzzy-AHP method. in 2011 International Conference on E-Business and E-Government (ICEE). IEEE, pp 1–5
36. Darwish S, Nouretdinov I, Wolthusen SD (2017) Towards composable threat assessment for medical IoT (MIOT). Proc Comput Sci 113:627–632
37. Alsubaei F, Abuhussein A, Shiva S (2018) A framework for ranking IOMT solutions based on measuring security and privacy. In: Proceedings of the Future Technologies Conference. Springer, Berlin, pp 205–224
38. Zhou J, Cao Z, Dong X, Lin X (2015) Ppdm: a privacy-preserving protocol for cloud-assisted e-healthcare systems. IEEE J Sel Top Signal Process 9(7):1332–1344
39. Ziglari H, Negini A (2017) Evaluating cloud deployment models based on security in EHR system. In: 2017 International Conference on Engineering and Technology (ICET). IEEE, pp 1–6
40. Sanz-Requena R, Mañas-García A, Cabrera-Ayala JL, García-Martí G (2015) A cloud-based radiological portal for the patients: It contributing to position the patient as the central axis of the 21st century healthcare cycles. In: IEEE/ACM 1st international workshop on TEchnical and LEgal aspects of data pRivacy and SEcurity. IEEE 2015, pp 54–57
41. Deshmukh P (2017) Design of cloud security in the EHR for Indian Healthcare Services. J King Saud Univ-Comput Inf Sci 29(3):281–287
42. Liu W, Liu H, Wan Y, Kong H, Ning H (2016) The yoking-proof-based authentication protocol for cloud-assisted wearable devices. Personal Ubiquitous Comput 20(3):469–479
43. Sun Y, Lo FP-W, Lo B (2019) Security and privacy for the internet of medical things enabled healthcare systems: a survey. IEEE Access 7:183339–183355
44. Haoyu L, Jianxing L, Arunkumar N, Hussein AF, Jaber MM (2019) An IOMT cloud-based real time sleep apnea detection scheme by using the SPO2 estimation supported by heart rate variability. Future Gener Comput Syst 98:69–77
45. Qureshi F, Krishnan S (2018) Wearable hardware design for the internet of medical things (IOMT). Sensors 18(11):3812
46. Han T, Zhang L, Pirbhulal S, Wu W, de Albuquerque VHC (2019) A novel cluster head selection technique for edge-computing based IOMT systems. Comput Netw 158:114–122
47. Kumar R, Tripathi R (2020) Secure healthcare framework using blockchain and public key cryptography. In: Blockchain cybersecurity, trust and privacy. Springer, Berlin, pp 185–202
48. Goyal TK, Sahula V (2016) Lightweight security algorithm for low power IoT devices. In: 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, pp 1725–1729
49. Chakravorty R (2006) A programmable service architecture for mobile medical care. In: Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06). IEEE
50. Barua M, Liang X, Lu R, Shen X (2011) ESPAC: Enabling security and patient-centric access control for Ehealth in cloud computing. Int J Secur Netw 6(2–3):67–76
51. Sultan A, Mushtaq MA, Abubakar M (2019) IoT security issues via blockchain: a review paper. In: Proceedings of the 2019 International Conference on Blockchain Technology, pp 60–65
52. Fotiou N, Polyzos GC (2016) Decentralized name-based security for content distribution using blockchains. In: 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, pp 415–420
53. Reddy AG, Suresh D, Phaneendra K, Shin JS, Odelu V (2018) Provably secure pseudo-identity based device authentication for smart cities environment. Sustain Cities Soc 41:878–885
54. Lee KC, Lee H-H (2004) Network-based fire-detection system via controller area network for smart home automation. IEEE Trans Consum Electron 50(4):1093–1100
55. Hammi MT, Hammi B, Bellot P, Serhrouchni A (2018) Bubbles of trust: a decentralized blockchain-based authentication system for IoT. Comput Secur 78:126–142
56. Al-Turjman F (2019) Security in IoT-enabled Spaces. CRC Press, Boca Raton
57. Hassija V, Chamola V, Saxena V, Jain D, Goyal P, Sikdar B (2019) A survey on IoT security: application areas, security threats, and solution architectures. IEEE Access 7:82721–82743

58. Zhang J, Wang Z, Yang Z, Zhang Q (2017) Proximity based IoT device authentication. In: IEEE INFOCOM 2017-IEEE Conference on Computer Communications. IEEE, pp 1–9
59. Wu M, Wang K, Cai X, Guo S, Guo M, Rong C (2019) A comprehensive survey of blockchain: from theory to IoT applications and beyond. IEEE Internet Things J 6(5):8114–8154